혼인 건수 ARMA 분석

과　　목	**시계열 자료분석**
학　　과	**응용통계학과**
학　　번	**201552043**
이　　름	**전태양**

1. 분석 개요

예로부터 평생을 함께할 안식처이자 울타리 같은 가족을 이루고 살기 위해서 혼인은 반드시 거쳐야 하는 필수 사회 규범이었다.
과거의 혼인은 자식으로서의 의무이자 시간이 지나면 당연히 해야 하는 과정으로 생각했지만 급속한 산업화와 도시화를 지나 개인의 욕구와 선택의 의지가 강해지면서 현대인들의 혼인에 대한 인식은 조건에 따라, 개인의 선택 기준에 따라 변화하고 있다.


“결혼 굳이 할 필요 있나”…달라진 결혼관

크게보기

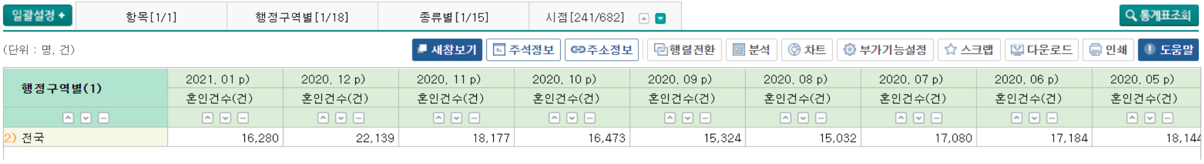
<현대인들의 변화한 결혼관>

이렇듯 본 보고서에서는 필수에서 선택으로 변화한 결혼관[1]이 혼인 수에도 영향이 있는지에 대하여 최근 20년간의 혼인 건수를 활용해 변화된 현대인들의 혼인 추세를 파악, 이후의 혼인 건수를 예측해보고자 한다.

[1] 사전적 의미: 결혼에 대하여 가지는 일정한 견해

2. 분석

2-1. 데이터 파악


일괄설정 + | 항목[1/1] | 행정구역별[1/18] | 종류별[1/15] | 시점[241/682] | 통계표조회

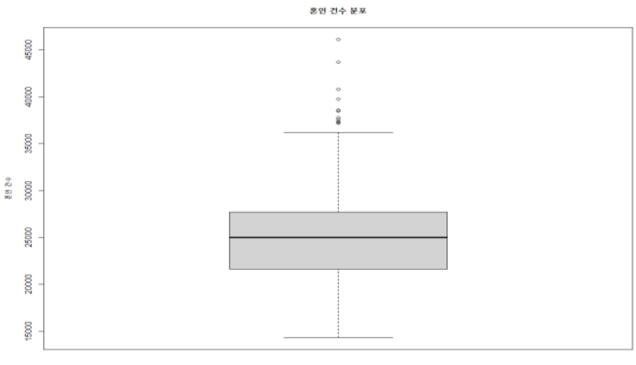
(단위 : 명, 건)

새창보기 | 주석정보 | 주소정보 | 행렬전환 | 분석 | 차트 | 부가기능설정 | 스크랩 | 다운로드 | 인쇄 | 도움말

행정구역별(1)	2021. 01 p)	2020. 12 p)	2020. 11 p)	2020. 10 p)	2020. 09 p)	2020. 08 p)	2020. 07 p)	2020. 06 p)	2020. 05 p)
	혼인건수(건)	혼인건수(건)	혼인건수(건)	혼인건수(건)	혼인건수(건)	혼인건수(건)	혼인건수(건)	혼인건수(건)	혼인건수(건)
2) 전국	16,280	22,139	18,177	16,473	15,324	15,032	17,080	17,184	18,144

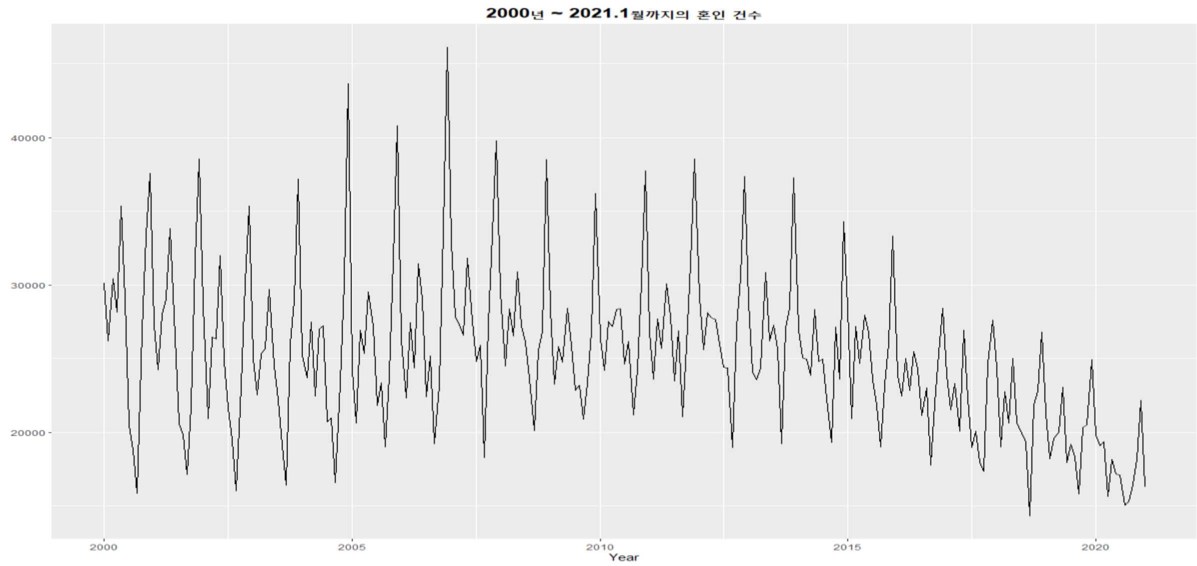
통계청 월.분기.연간 인구동향을 통해 2000 년 1 월부터 2021 년 1 월까지, 총 253 개월의 전국 혼인 건수 데이터를 활용했다.

<그림 1 - 연도별 혼인 건수>



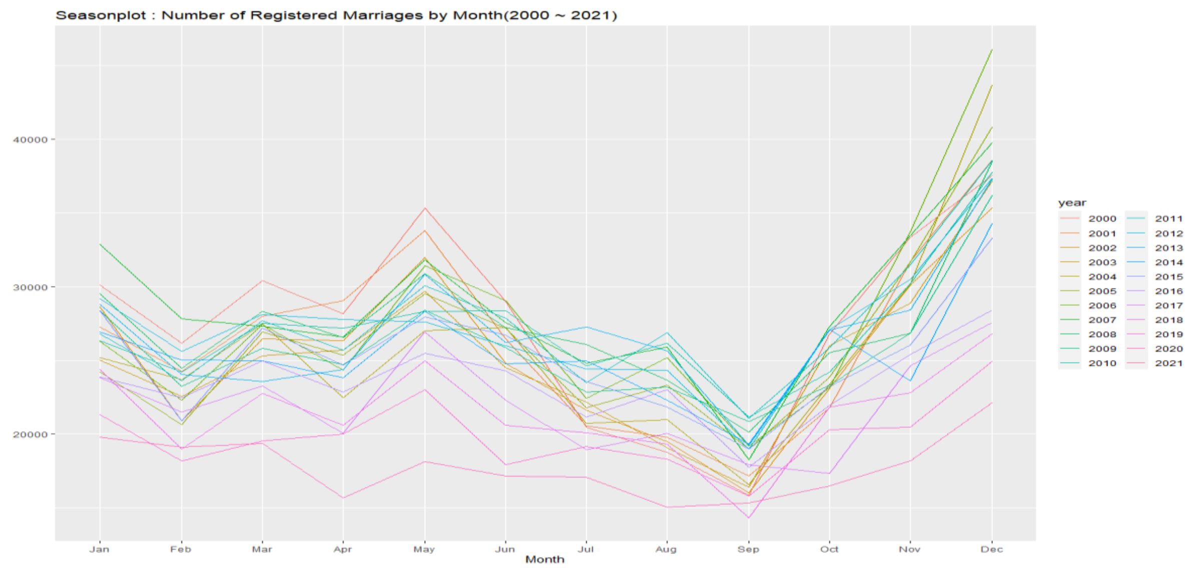
<그림 2 - 혼인 건수 분포>

먼저 월별 데이터를 연도별로 묶어 20 년 간 혼인 건수와 분포를 살펴본 결과, 2007 년에 343,559 건으로 혼인 건수가 가장 많았으며, 2015 년을 기점으로 꾸준히 줄어들어 2020 년에 213,502 건까지 줄어들었다. 20 년 간 평균 혼인 건수는 25,211 건이며, 최소 혼인 건수는 14,344 건, 최대 혼인 건수는 46,099 건으로 나타났다.



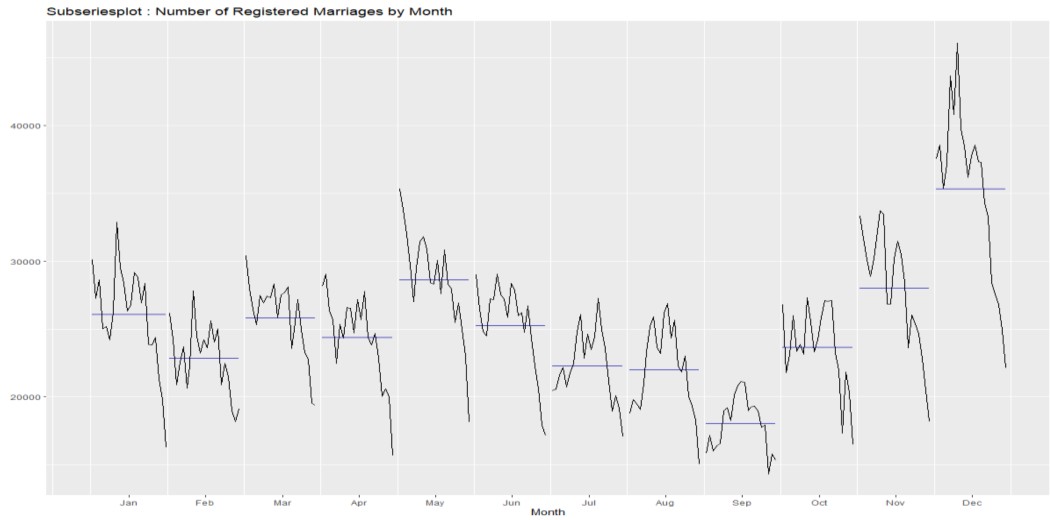
<그림 3 - 혼인 건수 추이>

데이터 추이를 살펴보면 전체적으로 시간이 지나면서 변동 폭이 줄어들고 혼인 건수가 감소하는 추세를 보인다.



<그림 4 - 혼인 건수 계절성 그래프>

계절성 그래프를 보면 대체적으로 9월의 혼인 수가 가장 적은 패턴을 보이며 5월과 12월일 때 혼인 수가 높은 패턴을 보인다.



<그림 5 - 혼인 건수 계절성 부시계열 그래프>

계절성 부시계열 그래프로 보면 9월의 수평선이 가장 낮으므로 평균 혼인 수가 가장 낮음을 의미하며 12월의 평균 혼인수가 다른 월에 비해 상당히 높게 나타났다.

데이터 탐색 결과 2015년을 기점으로 하향 추세를 보이며, 12월에 혼인 건수 평균이 높고, 9월에 혼인 건수 평균이 낮은 것으로 보아 조금의 계절성이 나타나는 것으로 보인다. 그렇기에 정상성[2]을 만족하지 못한다.

2-2. train 데이터와 test 데이터 분리

모델 학습에 사용할 train 데이터는 2000년 1월부터 2018년 12월까지의 혼인 건수 데이터를 사용하였으며, 학습한 모델의 정확도를 판단할 test 데이터는 마지막 2년(2019.01 ~ 2021.01)까지의 혼인 건수 데이터를 사용하여 분리했다.

Train 데이터: 총 228개월의 혼인 건수
Test 데이터: 총 25개월의 혼인 건수

[2] 정상성을 만족하려면 시계열의 평균이 일정하거나 분산이 시점에 의존하지 말아야한다.

2-3. 다중회귀모형 적합 및 결과

회귀모형은 혼인 건수를 반응 변수(Y_t)로 설정하고, 시계열 데이터의 추세를 나타내는 T_t와 계절 변동을 나타내는 S_t를 설명변수로 활용한 $Y_t = T_t + S_t + \varepsilon_t$ 식을 기반으로 진행한다.

이때, 회귀모형의 오차 ε_t는 관측값에서 계절 변동과 계절 변동을 뺀 상태로 비정상 요소가 모두 제거된 상태로 가정한다. 따라서 오차는 정상성을 만족해야 하며 자기상관이 없는 백색 잡음이어야 한다.

시계열 데이터에 대한 회귀를 사용하기에 앞서 반응 변수를 설명하기 위한 설명 변수 T_t와 S_t를 생성했다.

 # 추세 변수 생성

추세 변수T_t는 매년 1 월을 기준으로 12 월까지 $1 \div 12 = 0.833$의 등 간격으로 생성했다.

월	1	2	3	…	11	12
변수 값	2000.000	2000.083	2000.167	…	2000.833	2000.917

<표 1 - 추세 변수>

 # 계절성 변수 생성

 # 1. Seasonal dummy: 월별 평균의 차이가 이전의 계절 그래프에서 파악되었기에 해당 월에 속할 경우 1 아닐 경우 0 인 더미 변수를 생성하여, 계절성 설명 변수로 활용하였다.

Month	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov
Jan	1	0	0	0	0	0	0	0	0	0	0
Feb	0	1	0	0	0	0	0	0	0	0	0
Mar	0	0	1	0	0	0	0	0	0	0	0
⋮											
⋮											

<표 2 - Seasonal dummy>

회귀모형 적합

선형 추세와 월별 더미 변수를 고려하여 만든 회귀 모델식은 다음과 같다.

$$y_t = 404656.88 + -187.9t + -3854.82SeasonFeb + \cdots\cdots + 1935.64SeasonNov + 9587.88SeasonDec + \varepsilon_t$$

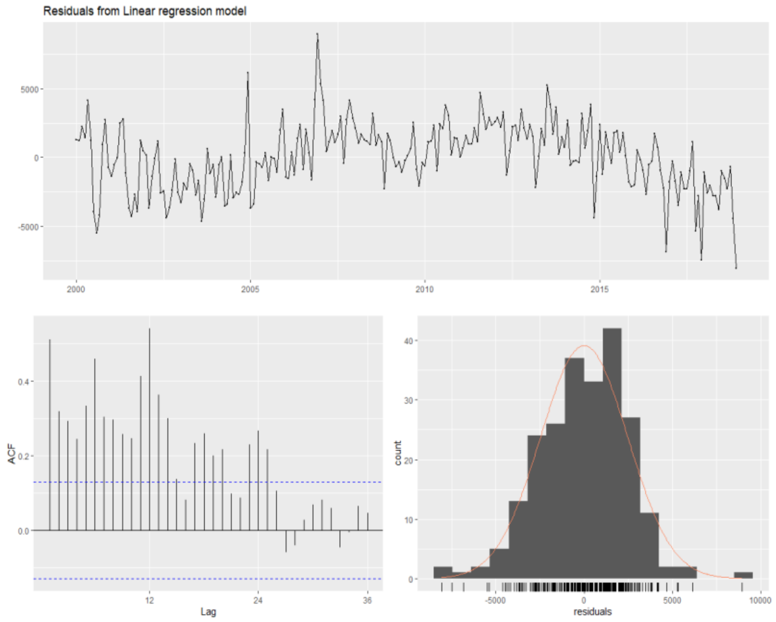
	Estimate	Pr(>|t|)	R^2	Pr(>|F|)
(Intercept)	404656.88	4.09e-10	0.7531	< 2.2e-16
Trend	-187.9	4.51e-09		
SeasonFeb	-3854.82	5.20e-06		
SeasonMar	-622.16	0.451394		
SeasonApr	-2041.34	0.014081		
SeasonMay	2400	0.003992		
SeasonJun	-972.34	0.2397		
SeasonJul	-4322.63	3.80e-07		
SeasonAug	-4471.55	1.58e-07		
SeasonSep	-8703.68	< 2e-16		
SeasonOct	-2801.34	0.000815		
SeasonNov	1935.64	0.019876		
SeasonDec	9587.88	< 2e-16		

<표 3 - 다중회귀모형 적합 결과>

추정된 다중회귀모형 결정계수 값이 0.7531 로 설명 변수들의 설명력이 약 75.3%로 나타났으며, F-값이 0.05 보다 작기 때문에 "설명 변수의 계수가 모두 0 이다."라는 귀무가설을 기각한다.

3 월과 6 월을 제외한 모든 월이 유의수준 5%에서 "1 월 대비 혼인 건수는 차이가 없다."라는 귀무가설을 기각했다. 월별 추세는 평균적으로 -187.9 건으로 감소하는 추세이며, 5, 11, 12 월을 제외하곤 1 월보다 평균 혼인 건수가 적게 나타났다.

 # 오차 가정 만족 여부 확인



<그림 6 – 다중회귀모형의 잔차>

Breusch-Godfrey test	
Model df	**P-Value**
17	1.456e-14

<표 4 – 잔차 독립성 가정 결과>

앞서 만든 회귀모형을 가지고 예측이 가능한지 판단하기 위해 오차 가정을 만족하는지 여부를 확인해본 결과, P-Value 값이 유의수준 5%에서 너무 낮아 "잔차가 독립이다"라는 가정을 만족하지 못했고, ACF 도 신뢰 한계선에서 튀어나와 있는 지점이 많이 보였다. 따라서 앞서 본 다중회귀모형은 오차 가정을 만족하지 못해 예측 모델로 활용하기 어렵다고 판단했다.

2-4. ARMA 오차 회귀 모형 적용 및 결과

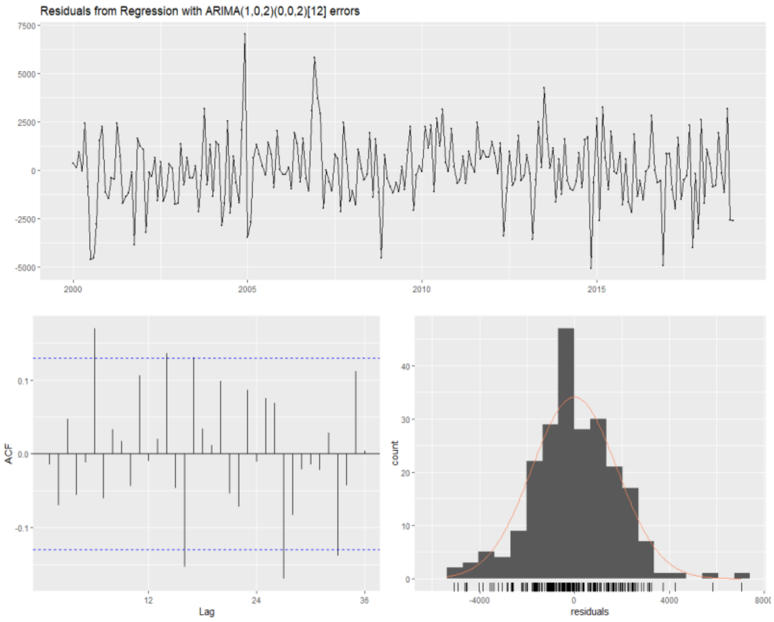
Auto.arima()를 사용하여 앞서 본 다중회귀 변수를 그대로 활용하면서, 다중회귀오차에 ARIMA 모형 오차를 반영한 모형을 만들었다.

그 결과, *ARIMA(1,0,2)(0,0,2)[12]*모형이 적용되었으며, 이전의 유의하지 않게 나왔던 3 월과 6 월의 값도 유의수준 5%에서 유의하게 나타났다.

	Ar1	**Ma1**	**Ma2**	**Sma1**	**Sma2**	**intercept**	**Time**	**Mar**	**Jun**	**AICc**
	0.9690	-0.6788	-0.1702	0.4233	0.2706	644357.5	-302.57-	-10125.6	-10510.9	4109.06
S.e	0.0234	0.0761	0.0729	0.0672	0.0743	267441.9	133.1074	935.4689	940.2974	

<표 5 - ARMA 오차 회귀 모형 적합 결과>

오차 가정 만족 여부 확인



<그림 7 – ARIMA(1,0,2)(0,0,2)[12]의 잔차>

Ljung-Box test	
Model df	**P-Value**
18	3.281e-06

<표 6 – 잔차 독립성 가정 결과>

ARIMA(1,0,2)(0,0,2)[12] 모형 진단 결과, P-Value 값이 유의수준 5%에서 “잔차가 독립이다.” 라는 귀무가설을 기각했으며, ACF 에서도 신뢰 한계선을 벗어나는 부분이 있어 모형의 잔차는 백색잡음이라고 말하기 어렵다. 그러나 예측오차와 신뢰구간에 영향을 주더라도 예측값에 대한 문제는 아니기에 이후의 분석을 진행했다.

 # 2. Fourier Series: 계절성 변수는 앞서 사용한 Seasonal dummy 대신 sine 변수와 cosine 변수의 선형 결합인 Fourier Series 변수로도 나타낼 수 있다. Fourier Series 는 최적 주기 K 를 기반으로 데이터에 계절 시점이 많은 경우 Seasonal dummy 보다 더 미세하게 계절성 패턴을 조절할 수 있다는 장점이 있다. 최적 주기는 ARMA 오차 모형의 AICc 가 최솟값을 가질때의 주기를 적용하며 K 는 데이터의 계절 주기(m)/2를 통해 구할 수 있다.

K	**1**	**2**	**3**	**4**	**5**	**6**
AICc	4162.819	4129.659	4129.698	4146.980	4120.936	4120.025

<표 7 - K값에 따른 AICc>

1 부터 데이터의 계절주기 12/2 까지의 AICc 를 계산한 결과 K = 6 일 때, 모형의 AICc 가 가장 작은 값을 가지는 것으로 나타났다. 계절성 주기의 최댓값을 사용했기 때문에 이는 이전에 구했던 Seasonal dummy 의 변수 개수 11 개와 같은 것을 알 수 있다.

첫 번째(K = 1) 주기 항	S1-12, C1-12 = $\sin\left(\frac{2\pi t}{12}\right), \cos(\frac{2\pi t}{12})$
두 번째(K = 2) 주기 항	S2-12, C2-12 =$\sin\left(\frac{2\pi 2t}{12}\right), \cos(\frac{2\pi 2t}{12})$
⋮	⋮
다섯 번째(K = 5) 주기 항	S5-12, C5-12 =$\sin\left(\frac{2\pi 5t}{12}\right), \cos(\frac{2\pi 5t}{12})$
여섯 번째(K = 6) 주기 항	$k = \left[\frac{m}{2}\right] = 0$ *일 때* $\sin\left(\frac{2\pi kt}{m}\right) = 0 , \cos(\frac{2\pi 5t}{12})$

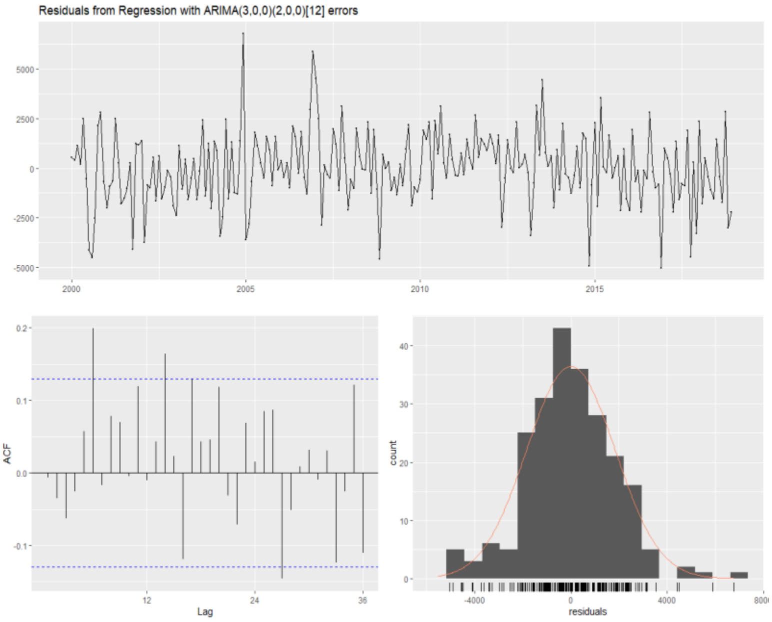
<표 8 - 1 - 주기별 sin, cos 변수>

	S1-12	**C1-12**	**S2-12**	**C2-12**	**S3-12**	**C3-12**	**S4-12**	**C4-12**	**S5-12**	**C5-12**	**C6-12**
	0.5	0.866	0.866	0.5	1	0	0.866	-0.5	0.5	-0.866	-1
	0.866	0.5	0.866	-0.5	0	-1	-0.866	-0.5	-0.866	0.5	1
⋮											

<표 8 - 2 - 주기별 sin, cos 변수>

그러나 Fourier 를 활용한 arma 오차 회귀 모형 적합 결과는, Seasonal dummy 를 적용했을 때 *ARIMA(1,0,2)(0,0,2)[12]* 나왔던 것과 달리 *ARIMA(3,0,0)(2,0,0)[12]* 모형이 적합되었다.

오차 가정 만족 여부 확인



<그림 8 – ARMA(3,0,0)(2,0,0)[12]의 잔차>

Ljung-Box test	
Model df	**P-Value**
18	4.679e-07

<표 9 – 잔차 독립성 가정 결과>

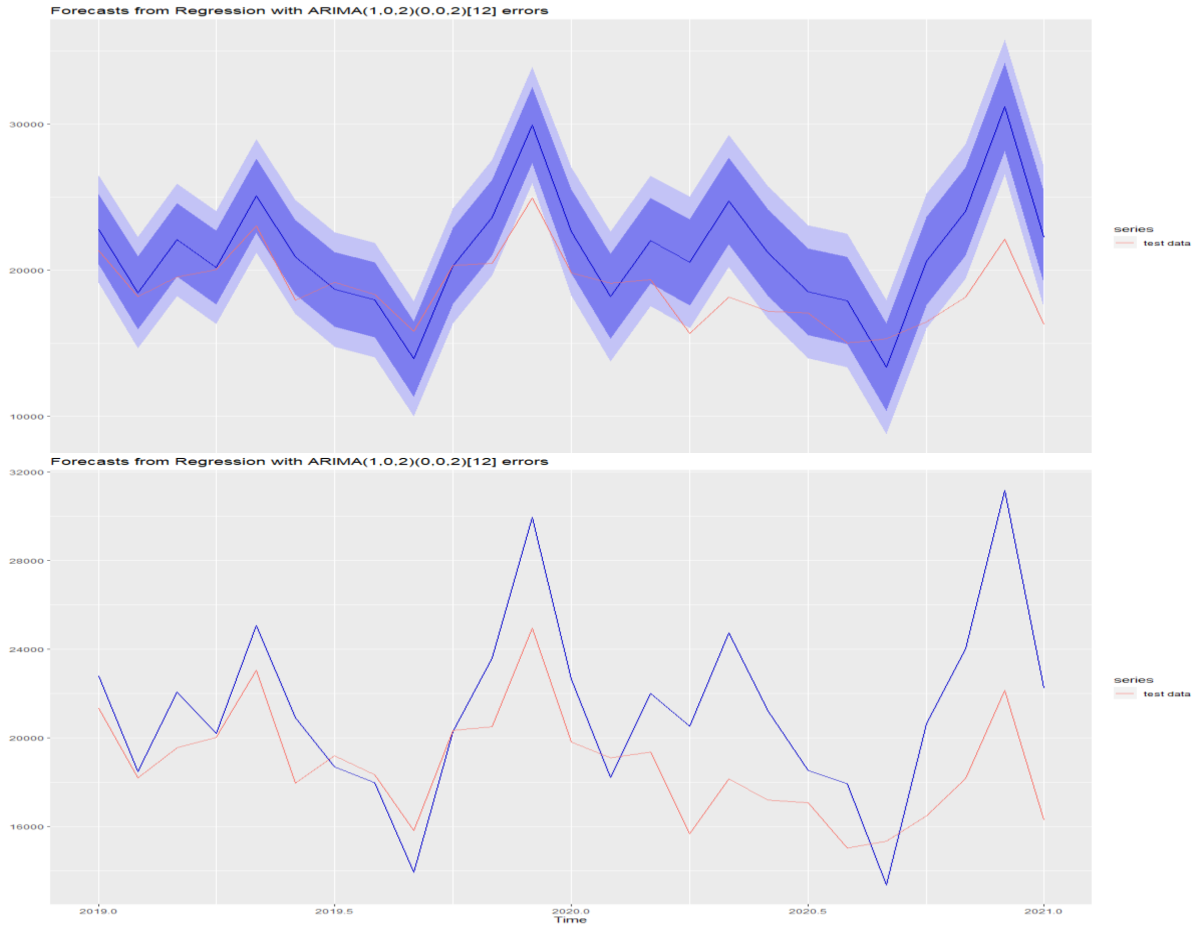
ARIMA(3,0,0)(2,0,0)[12] 모형 역시 유의수준 5%에서 잔차의 독립성 가정을 만족하지 못했다. 잔차 그래프를 보면 잔차의 분산 변동 폭이 커지는 구간이 보이며, ACF 그래프에서 신뢰 한계선을 벗어나는 구간이 나타나 잔차가 백색 잡음이라고 보기 어렵다.

Model		**RMSE**	**MASE**	**AICc**
Regression with ARIMA(1,0,2)(0,0,2)[12]	**Train Data**	1792.477	0.7707539	4109.06
	Test Data	3698.524	1.6557962	
Regression with ARIMA(3,0,0)(2,0,0)[12]	**Train Data**	1836.402	0.7984612	4120.025
	Test Data	4113.452	1.8734028	

<표 10 – ARMA 모형별 예측 성능>

모형별 성능 확인 결과, Seasonal dummy 를 사용한 *ARIMA(1,0,2)(0,0,2)[12] 모형이 AICc* 도 더 낮았으며 Train Data 와 Test Data 의 RMSE 와 MASE 가 더 낮게 나와 모두 성능이 더 좋게 나타났다.

따라서, 최종 모형으로 *ARIMA(1,0,2)(0,0,2)[12]* 모형을 채택하였다.



<그림 9 – ARIMA(1,0,2)(0,0,2)[12] 모형의 혼인 건수 예측>

그래프로 살펴보면 예측값과 실제 데이터 값의 차이가 있으며, 신뢰구간에서 벗어난 부분도 보인다.

3. ETS, ARIMA, ARMA 모형 비교

Model		$ETS(M, Ad, M)$	$ARIMA(0,1,1)(0,1,1)[12]$	$ARIMA(1,0,2)(0,0,2)[12]$
RMSE	**Train**	1839.457	1853.143	1792.477
	Test	2817.157	1875.007	3698.524
MASE	**Train**	0.7919666	0.7833229	0.7707539
	Test	1.2592082	0.8933152	1.6557962
백색잡음		X	X	X

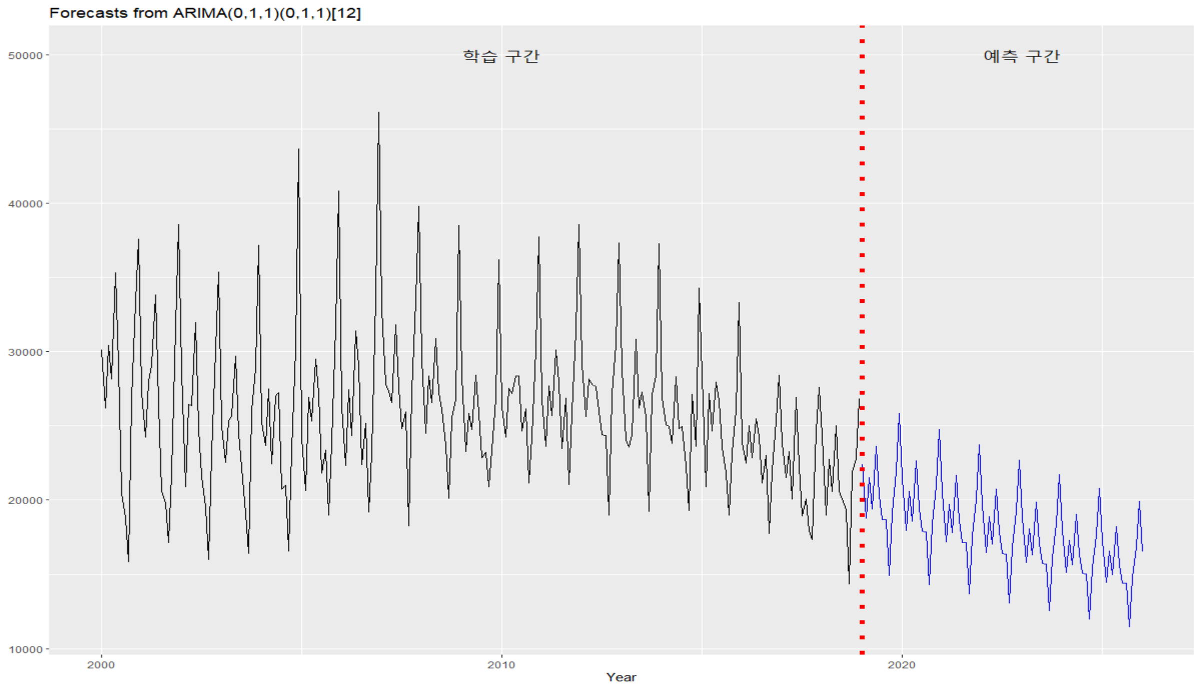
<표 11 - 모형별 예측 성능>

3 가지 모형 비교 결과 ARIMA 모형의 성능이 가장 좋게 나타났다.

<그림 10 – 모형별 혼인 건수 예측>

4. 결론

본 보고서는 과거 20 년 간 혼인 건수를 기반으로 미래 혼인 건수를 예측하는 것을 목적으로 진행했으며 예측 오차가 가장 작았던 ARIMA(0,1,1)(0,1,1)[12] 모형을 최종 모형으로 선정했다.



<그림 11 – 2018년 기준 혼인 건수 예측>

그 결과 최종모형을 기반으로 향후 5 년간 미래 혼인 건수는 5 월과 12 월에 증가하고 9 월에 감소하는 추세를 반복하며 서서히 줄어들 것이라고 예측하였다.

본 보고서에서 ARIMA 모형을 기반으로 진행한 분석은 단지 과거의 혼인 건수만을 이용하여 패턴을 분석한 것이다. 따라서 사회 환경, 주거 환경과 같은 외부 변수에 의해 혼인 건수는 또다시 변할 가능성이 크다.

추후에 이러한 외부 변수를 활용하여 추가적으로 분석한다면 더욱 더 유의미한 결과를 도출해 현재 혼인 건수의 감소에 영향을 미치는 가장 큰 요인을 찾아낼 수 있어 앞으로의 결혼, 출산 문제를 해결하는데 활용될 수 있을 것으로 판단된다.

5. 부록

5-1. 코드

```
# 패키지 호출
library(forecast)
library(readxl)
library(tidyverse)
library(fpp2)
library(Rmisc)

# data 불러오기
data <- read_excel("C:/Users/admin/Desktop/program/혼인건수(2000_2021).xlsx")

# 데이터 정제
data <- as.numeric(as.vector(as.matrix(data[-1,-1])))

# data 를 시계열 객체로 변경
ts_df<- ts(data,start = 2000,frequency = 12)

# 결측값
sum(is.na(ts_df))

# 데이터 개수
length(ts_df)

# 데이터 요약
summary(ts_df)

# 연간 데이터로 변환
df <- data.frame(Y=as.matrix(ts_df), year = floor(time(ts_df)))
gp_df <- df %>% group_by(year) %>%
  summarise(sum_Y = sum(Y))
gp_df <- gp_df[-22,]

# 연간 그래프
gp_df %>% ggplot() +
  geom_line(aes(x = factor(year)[1:21], y = sum_Y[1:21], group = 1)) +
  ggtitle('연도별 혼인 건수') + labs(x = "연도", y = "혼인 건수") +
  theme(plot.title = element_text(hjust = 0.5, size = 15, face = 'bold'))

# boxplot
boxplot(ts_df, ylab = "혼인 건수", main = "혼인 건수 분포")

# 혼인 건수 추이
autoplot(ts_df) +
  ylab(NULL) +
  ggtitle("2000 년 ~ 2021.1 월까지의 혼인 건수")+
```

```r
  theme(plot.title = element_text(hjust = 0.5, size = 15, face = 'bold')) +
  labs(x = "Year")

# subseriesplot
ggsubseriesplot(ts_df) +
  labs(title = "Subseriesplot : Number of Registered Marriages by Month", y = NULL)

# ggseasonplot
ggseasonplot(ts_df) +
  labs(title = "Seasonplot : Number of Registered Marriages by Month(2000 ~ 2021)")

# train_test_split
train_df <- window(ts_df,end = c(2018,12))
test_df <- window(ts_df,start = c(2019,1), end = c(2021,1))
length(train_df)
length(test_df)

#
ggtsdisplay(train_df)

# 추세 변수 생성
Time <- time(train_df)
time(train_df)[1:12]

# 계절 변수 생성
season <- seasonaldummy(train_df)
season
view(seasonaldummy(train_df)[1:12,])

#  다중회귀모형
fit1 <- tslm(train_df ~ Time + season)
summary(fit1)

# 다중회귀모형 적합값 확인
fit1$fitted

# 오차 가정 만족 여부 확인
checkresiduals(fit1)

# 적합 결과 비교 그래프
data.frame(data = train_df, fitted = fit1$fitted) %>%
  ggplot(aes(x = data, y = fitted)) +
  geom_point(aes(color = factor(cycle(data))), size = 2) +
  geom_abline(intercept = 0, slope = 1) +
  labs(color = "Month", x= "Observed data", y = "fitted data")

# 회귀모형에 의한 예측
fc1 <- forecast(fit1, newdata = data.frame(Time = time(test_df), seasonaldummy(test_df)))
```

```r
# ARMA 오차 회귀 모형 적합
fit2 <- auto.arima(train_df, xreg = cbind(Time,   season),
                stepwise = FALSE, approximation = FALSE)

fit2

# 오차 가정 만족 여부 확인
checkresiduals(fit2)

# 예측값
fc2 <- forecast(fit2,
                       xreg = cbind(time(test_df), seasonaldummy(test_df)))
fc2

# 최적 주기 AICc 계산
Time <- time(train_df)
res <- vector("numeric",6)
for (i in seq(res)){
   xreg <- cbind(Time, fourier(train_df,K=i))
   fit <- auto.arima(train_df,xreg=xreg)
   res[i] <- fit$aicc
}
res
(k_min <- which.min(res))
res

# 푸리에 급수
fourier(train_df, K = 6)

# 다중회귀모형 적합
fit3 <- tslm(train_df ~ time(train_df) + fourier(train_df,K = 6))
summary(fit3)

# 푸리에 오차 arma 모형 적합
Time <- time(train_df)
Fourier <- fourier(train_df, K = 6)
fit4 <- auto.arima(train_df, xreg = cbind(Time, Fourier))
summary(fit4)

# 오차 가정
checkresiduals(fit4)

# 모형 예측
new_g3 <- cbind(Time = time(test_df),
                       Fourier = fourier(test_df, K = 6))

# 예측값
```

```
fc4 <- forecast(fit4 , xreg = new_g3)
fc4

# 성능 평가
accuracy(fc2, test_df)
accuracy(fc4, test_df)


# AICc
fit2$aicc
fit4$aicc

### fit2 채택

# autoplot
plot_fc2 <- autoplot(fc2, include = 0)+
  autolayer(test_df, series = 'test data') +
  labs(y = NULL)

plot_fc2_1 <- autoplot(fc2, include = 0, PI = FALSE) +
  autolayer(test_df, series = 'test data') +
  labs(y = NULL)


# ARIMA
fit_arima <- Arima(train_df, order = c(0,1,1),seasonal = c(0,1,1),lambda=0)
fit_arima


fc_arima <- forecast(fit_arima, h = length(test_df))
fc_arima

accuracy(forecast(fc_arima),test_df)

plot_arima <- autoplot(fc_arima, include = 0)+
  autolayer(test_df, series = 'test data') +
  labs(y = NULL)

plot_arima2 <- autoplot(fc_arima, include = 0, PI = FALSE) +
  autolayer(test_df, series = 'test data') +
  labs(y = NULL)


### ETS ###
fit1 <- ets(train_df)
# 모델 정보 요약
summary(fit1)
autoplot(fit1)
```

```
# 모형의 가정 만족 여부 확인
checkresiduals(fit1)
# 예측 진행
fc1 <- forecast(fit1, h = length(test_df))
fc1

accuracy(fc1, test_df)

plot_fc1 <- autoplot(fc1, include = 0)+
  autolayer(test_df, series = 'test data') +
  labs(y = NULL)

plot_fc1_1 <- autoplot(fc1, include = 0, PI = FALSE) +
  autolayer(test_df, series = 'test data') +
  labs(y = NULL)

# subplot
multiplot(plot_fc1, plot_arima, plot_fc2)
multiplot(plot_fc1_1, plot_arima2, plot_fc2_1)

# 5 년 예측 그래프
fc_arima5 <- forecast(fit_arima, h = length(test_df) + 60)
autoplot(fc_arima5 , alpha = 0.3, PI = FALSE) +
  geom_vline(xintercept=2019,linetype = 'dotted',color = "red", size=2) +
  annotate(geom="text", x=2023, y=50000,
           label="예측 구간", size = 6) +
  annotate(geom="text", x=2010, y=50000,
           label="학습 구간", size = 6) +
  labs(x = 'Year', y = NULL)
```

5-2. 출처

데이터: 통계청 월.분기.연간인구동향(출생, 사망, 혼인, 이혼)

https://kosis.kr/statisticsList/statisticsListIndex.do?menuId=M_01_01&vwcd=MT_ZTITLE&parmTabId=M_01_01&outLink=Y&entrType=

기사: https://www.gokmu.com/mobile/article.html?no=14992