서울시 아파트 실거래 가격 지수 예측

고영희

1. 시계열의 경향 분석

1) 이동평균 기간 구하기

경기변동 사이클을 예측하는 모델 중 "키친 사이클 (Kitchin Cycles)"을 사용해 이동 평균 기간을 구한다. 키친 사이클은 물가나 금리 등을 기준으로 움직이는 소순환 사이클을 의미하며, 3년~4년을 주기로 갖는다. 따라서 아파트 실거래 가격 지수의 인덱스는 월별 데이터 이기 때문에 이동평균 후보를 36개월 ~ 48개월로 지정한다. 이동평균 후보 중, 단순 이동 평균의 예측 값과 실제값의 차이(잔차) 평균을 의미하는 MSE가 가장 작은 값을 이동평균으로 지정한다.

```
> m.vec<-36:48
> sma.mse.vec<-c()

> for (m in m.vec){
      sma.m <- c(NA,SMA(df.ts,n=m))
      end<-length(sma.m)
      sma.mse<-mean((df.ts-sma.m[-end])^2,na.rm=TRUE)
      sma.mse.vec<-c(sma.mse.vec,sma.mse)
   }

> m.vec[which.min(sma.mse.vec)]
```

Output]

```
[1] 36
```

✓ 이동 평균은 MSE가 가장 작은 36개월로 정한다.

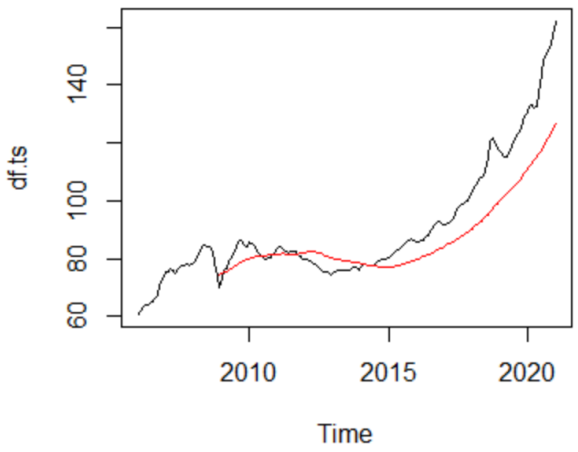
2) 단순 이동평균 (simple moving average, SMA)

앞서 정한 36개월의 이동평균 구간(m) 내 표본 평균을 구해 데이터의 추세를 파악한다.

$$SMA_t = \frac{x_t + x_{t-1} + \cdots + x_{t-m+1}}{m} \quad , m \le t \le n$$

```
> plot(df.ts)

> sma.vec2<-SMA(df.ts,n=36)

> lines(sma.vec2,col='red')
```

Output]



- ✓ 이동평균 기간을 36개월이라는 큰 값으로 했기 때문에 평활이 많이 되었다는 것을 알 수 있다.
- ✓ 단순이동평균은 추세를 잘 나타내지만, 과거의 데이터만을 이용해 현재시점의 값을 나타내기 때문에 데이터의 처음 35개의 값은 결측치로 나타낸다.
- ✓ 따라서 데이터의 추세가 35개월만큼 뒤로 밀리는 모습을 볼 수 있다.
- ✓ 이러한 점을 보완하기 위해 중심화 이동평균을 통해 추세를 확인한다.

3) 중심화 이동평균 (MA)

이동평균 구간내 데이터를 사용하되, 추정하고자 하는 시점을 기준으로 과거의 데이터만이 아니라 과거와 미래 데이터를 적절히 합하여 사용한다. 여기서 이동평균 기간(m)은 36이므로 짝수일 때의 중심화 이동평균을 사용한다. $(m = 2k)$

$$MA_t = \frac{{MA_t}^P + {MA_t}^F}{2} \ , t = k+1, \dots n-k$$

$${MA_t}^P = \frac{x_{t-k} + \cdots + x_{t+k-1}}{m}$$

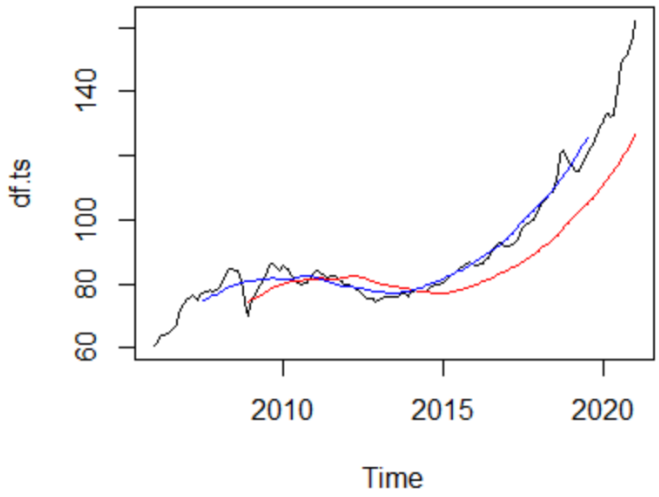
$${MA_t}^F = \frac{x_{t-k+1} + \cdots + x_{t+k}}{m}$$

```
> ma.cen<-ma(df.ts,order=36, centre=TRUE)
> head(ma.cen,20)
> tail(ma.cen,20)

> lines(ma.cen,col='blue')
```

Output]

```
      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
2006  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
2007  NA   NA   NA   NA   NA   NA   74.65139 75.02222
```

```
      Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
2019                        124.0722 125.6792  NA   NA   NA   NA   NA
2020  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
2021  NA
```



- ✓ 단순 이동 평균은 처음 35개의 데이터가 결측치인 반면, 중심화 이동평균은 처음 18개의 데이터가 결측치이고, 마지막 18개의 데이터가 결측치로 할당된다.
- ✓ 따라서 단순이동 평균과 중심화 이동평균의 결측치 개수는 35개로 같지만 결측치의 위치는 다르게 해 추세에 대한 시차를 없앤다.

4) 국소회귀 (Locally Weighted Regression and Smoothing, lowess)

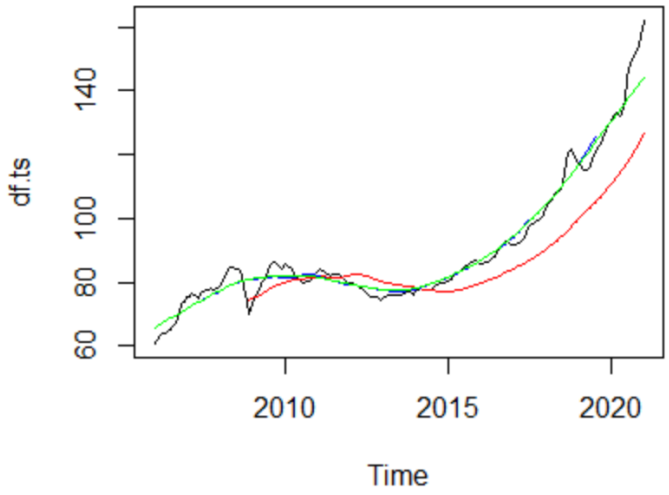
단순 이동 평균과 마찬가지로 일부 데이터를 사용하되, 일부 데이터의 평균을 구하는 것이 아니라 가중치가 존재하는 회귀분석(WLS)를 사용해 회귀식을 도출한다. 단, 일부 데이터를 사용할 때, 단순 이동평균과 다르게 이동 평균 기간을 정하는 것이 아니라 $size = 2k+1 \approx nf$, $0 < f < 1$에서 f의 값을 지정한다.

$\max_{b_0,b_1} \sum_{j=-k}^{k} w_j \left(x_{t+j} - b_0 - b_1(t+j)\right)^2$일 때 $\widehat{\beta_{0t}} = b_0$, $\widehat{\beta_{1t}} = b_1$

$\widehat{x_t} = \widehat{\beta_{0t}} + \widehat{\beta_{1t}} * t$ 적합값을 사용한다.

```
> low<-lowess(df.ts,f=1/3)

> lines(low,col='green')
```

Output]



✓ 국소회귀를 시행했을 때의 추세선은 중심화 이동평균의 추세선과 비슷하다는 것을 알 수 있다.

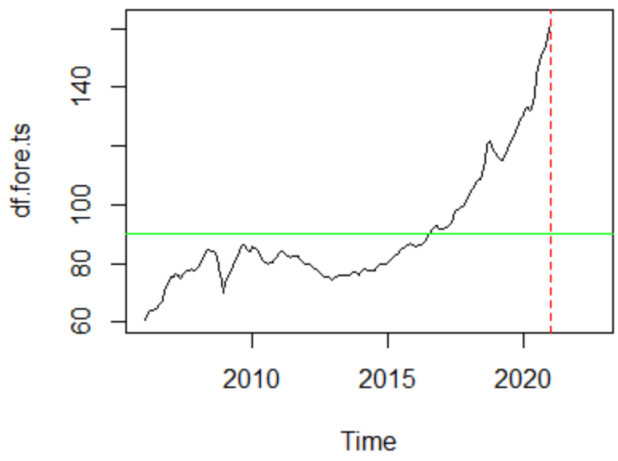
2. 미래 시점 값 예측

1) 데이터 프레임 변경 및 시각화

2006년 1월부터 2020년 1월까지의 기존 데이터 프레임값 이후를 20개월 이후의 시점까지 예측하기 위해 2020년 1월 이후의 20개월 값을 결측치로 만들어 벡터로 합친다. 합친 벡터를 시계열 데이터로 변환하고 시각화한다. 이때 가장 마지막 데이터가 있는 2021년을 기준으로 수직선을 그리고, 수직선 이후의 값을 예측하는 것이 목표이다. 또한, 데이터의 평균값을 초록색 수평선으로 나타낸다.

```
> df.fore<-c(df.ts,rep(NA,20))

> df.fore.ts<-ts(df.fore,start=c(2006,1),frequency = 12)

> plot(df.fore.ts)

> abline(v=2021,col='red',lty=2)

> abline(h=mean(df.ts),col='green')
```

Output]



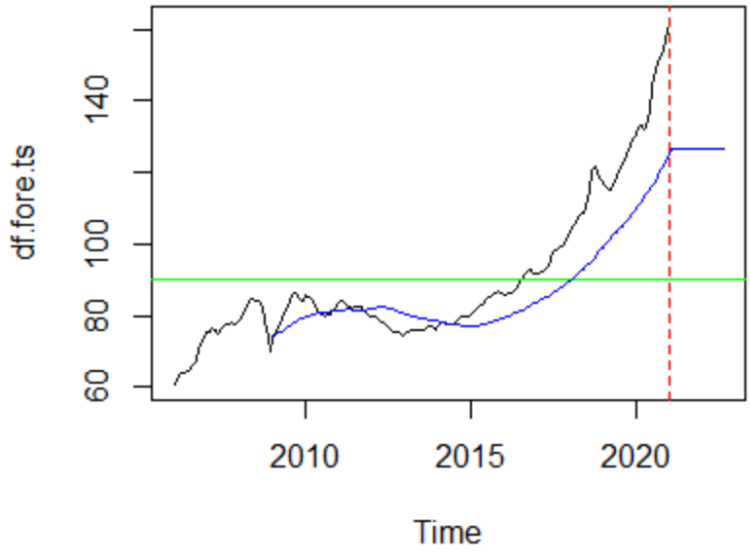
✓ 데이터의 평균값은 시계열 데이터의 추세를 나타내지 않고, 가장 최근 데이터와는 많은 차이가 있다는 것을 알 수 있다.

2) 단순이동평균(SMA)을 통한 예측

이동평균을 시행할 때, 앞선 분석과 마찬가지로 이동평균 기간은 36개월로 한다. 이동평균에서의 미래 예측값은 국소적으로 고정된 평균 모형을 사용하기 때문에 가장 마지막에 구한 단순 이동 평균값이 예측값이 된다. 따라서 가장 마지막 값을 19번 반복하고, 가장 처음 값에는 NA 값을 넣어 한 시차씩 늦춰 예측값과 시점을 일치시킨다.

```
> sma.36<-SMA(df.ts,n=36)

> n<-length(df.ts)

> sma.fore.ts<-ts(c(NA,sma.36,rep(sma.36[n],19)),start=c(2006,1),frequency = 12)

> lines(sma.fore.ts,col='blue')
```

Output]



- ✓ 단순 이동평균을 이용한 예측값은 데이터가 점점 증가하는 추세를 무시하고 미래의 값을 고정된 상수로만 예측한다.
- ✓ 가장 최근의 데이터인 2021 년 1 월 데이터와의 차이도 많이 나기 때문에 좋은 예측값이라 말할 수 없다.

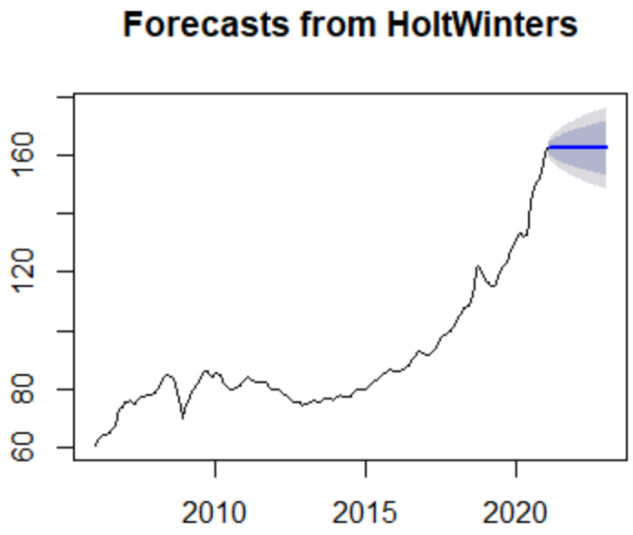
3) 단순지수평활(Simple Exponential Smoothing)을 통한 예측

미래에 영향을 많이 준 최근 데이터는 큰 가중치를 주고, 영향을 적게 준 과거 데이터는 작은 가중치를 주어 최근 흐름을 반영한다. 또한 모든 데이터를 사용해 미래의 값을 예측한다.

$$F_n\,(l)\;\approx\;\sum_{t=1}^{n} w(1-w)^{n-t}\,x_t$$

```
> es.auto<- HoltWinters(df.ts,beta=FALSE,gamma=FALSE)

> lines(ts(c(NA,es.auto$fitted),start=c(2006,1),frequency = 12),col='purple')

> plot(forecast(es.auto))
```

Output]



- ✓ 단순 이동평균과 마찬가지로 데이터가 증가하는 추세를 무시하고 미래의 값을 고정된 상수로 예측한다.
- ✓ 하지만 단순 이동평균과 다른 점은 가장 최근의 데이터를 많이 반영하기 때문에 좀 더 좋은 예측값이라 할 수 있다.

4) 단순이동평균(SMA)와 이중이동평균(DMA)을 통한 예측

데이터가 시간의 흐름에 따라 증가하는 추세가 있기 때문에 단순이동 평균과 이중이동 평균을 이용해 선형 이동 평균법을 사용한다. 이는 국소적으로 선형추세 모형을 가정하기 때문에 미래의 값을 상수항으로 가정한 단순 이동 평균과 단순 지수 평활법보다는 더 좋은 예측을 할 것이라고 예상할 수 있다.

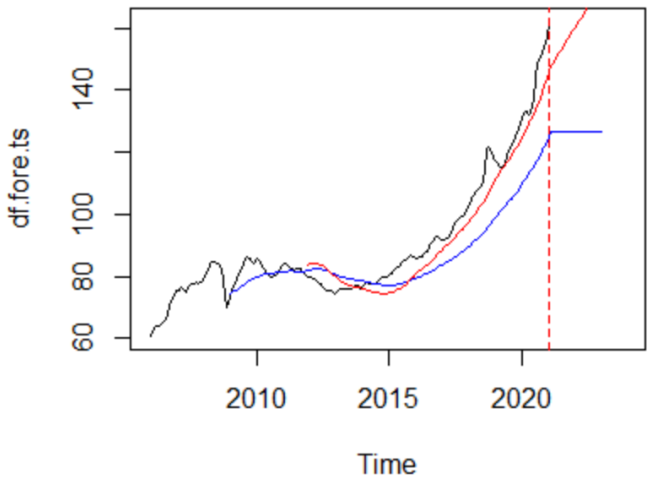
$$F_n(l) = (2SMA_n - DMA_n) + \frac{2}{m-1}(SMA_n - DMA_n) * l$$

$$SMA_t = \frac{x_t + x_{t-1} + \cdots + x_{t-m+1}}{m}, m \le t \le n$$

$$DMA_t = \frac{SMA_t + SMA_{t-1} + \cdots + SMA_{t-m+1}}{m}, 2m-1 \le t \le n$$

```
> dma.36<-SMA(sma.36,n=36)

> ma.trend<-c(NA,(2*sma.36-dma.36)+2/35*(sma.36-dma.36),
                (2*sma.36[n]-dma.36[n])+2/35*(sma.36[n]-dma.36[n])*(2:23))

> lines(ts(ma.trend,start=c(2006,1),frequency = 12),col='red')
```

Output]



✓ 파란선의 단순이동 평균 예측값보다 단순이동 평균과 이중 이동 평균을 통한 예측값이 가장 최근의 데이터와 더 가깝고 전반적인 데이터의 증가 추세를 잘 반영하고 있기 때문에 더 좋은 예측이라고 할 수 있다.