Smart Technology - AIR

Forecasting

Forecasts

- (과거와 현재의 데이터를 바탕으로) 관심있는 변수의 미래 값을 나타내는 것
  - 수요 예측
  - 인구 예측

- Forecasts의 두가지 활용
  - Plan the system
  - Plan the use of the system

- Forecasts의 특성
  - Assumes that the causal system existed in the past will continue to exist in the future
  - Forecasts rarely perfect because of randomness
  - Forecasts for groups of items tend to be more accurate than forecasts for individual items
  - Forecast accuracy decreases as the time period covered (time horizon) by the forecast increase

Methods

- 예측의 방법들
  - Qualitative: rely on human judgment (little historical data / experts with market intelligence)
  - Time series: use historical demand (stable demand pattern)
  - Causal: find relationship between demand and other factors (price and demand)
  - Simulation: imitate customers choices

  → combined forecast is more effective than using any one method alone

- 전통적인 예측 방법과 지도학습과의 차이
  - Regression 방법은 Forecast에 많이 활용됨
  - 시계열 데이터를 다룸 (예, 판매량, 공급량, 인구, GDP)
  - Trend, Cycle등의 변화의 특성을 활용
  - 미래 데이터는 과거데이터와 다른 경우가 많음

시계열 (Time-Series) 예측 기법

- Time-series: a time-oriented sequence of observations taken at regular interval (e.g. daily, weekly, monthly)

  Observed demand = systematic component + random component

- Systematic component – expected value of demand
  - Level: current deseasonalized demand
  - Trend: growth or decline in demand
  - Seasonality: predictable seasonal fluctuation

- Random component – part of forecast that deviates from systematic component

- Forecast error – difference between forecast and actual demand

Forecasting Level

- Naive method: a forecast for any period that equals the pervious period's actual value

$$F_t = A_{t-1}$$

where

F_t	=	forecast for next period, t
A_{t-1}	=	actual value for current period, $t-1$

- Modification
  - Naive method with seasonality: forecast for this season is equal to the actual value of last season

$$F_t = A_{t-n}$$

where

n	=	number of period in one seasonal cycle

  - Naive method with trend: forecast is equal to the last value plus the difference between the last two values

$$F_t = A_{t-1} + (A_{t-1} - A_{t-2})$$

Forecasting Level

- N-period Moving average: the average demand over the last *N* period

$$F_t = \frac{A_{t-1} + A_{t-2} + \ldots + A_{t-n}}{n} = \frac{\sum_{i=1}^{n} A_{t-i}}{n}$$

where

n = number of period in the moving average

- Compute a three-period moving average forecast given demand for shopping carts for the last five period

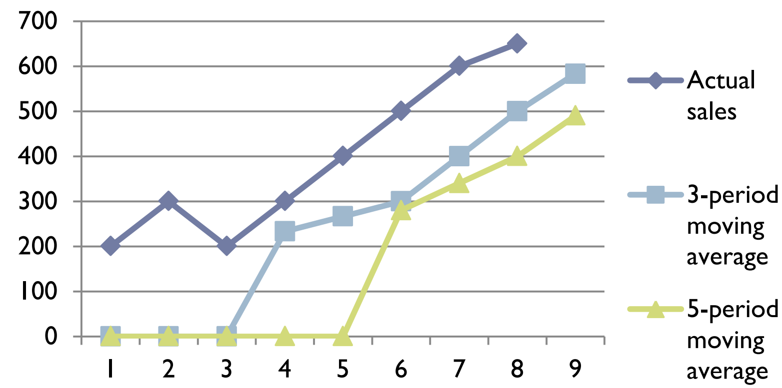
Period	Demand
1	42
2	40
3	43
4	40
5	41

If actual demand in period 6 is 38,

Forecasting Level

- <u>N-period Moving average</u>
  - Averaging technique smooth fluctuations in a time series
  - The forecasts are training behind the actual data (lagging the data)
  - Easy to use

Month	Actual sales	3-period moving average	5-period moving average
January	200		
February	300		
March	200		
April	300	233.3	
May	400	266.7	
June	500	300.0	280
July	600	400.0	340
August	650	500.0	400
September		**583.3**	**490**



- If responsiveness is important, a moving average with relatively few data points should be uses
- Moving averages based on more data points will be more stable

Forecasting Level

- <u>Weighted moving average</u>: Assign more weight to the more recent values in a time series

$$F_t = w_{t-1}A_{t-1} + w_{t-2}A_{t-2} + \ldots + w_{t-n}A_{t-n} = \sum_{i=1}^{n} w_{t-i}A_{t-i}$$

where

w_t = weight placed on the actual value in period t

- Compute a weighted moving average forecast using a weight of 0.4 for the most recent period, 0.3, 0.2 and 0.1

Period	Demand
1	42
2	40
3	43
4	40
5	41

If actual demand in period 6 is 39,

Forecasting Level

- Simple exponential smoothing: A weighted moving average method based on previous forecast plus a percentage of the forecast error

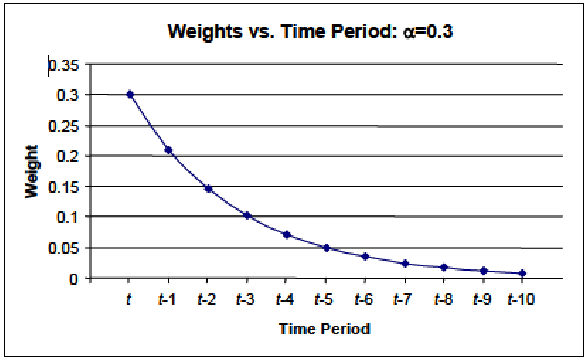
$$
\begin{aligned}
F_t &= F_{t-1} + \alpha\left(A_{t-1} - F_{t-1}\right) \\
&= \left(1-\alpha\right)F_{t-1} + \alpha A_{t-1}
\end{aligned}
$$

where

α = smoothing coefficient $(0 \le \alpha \le 1)$

- Since $F_t = (1-\alpha)F_{t-1} + \alpha A_{t-1}$ and $F_{t-1} = (1-\alpha)F_{t-2} + \alpha A_{t-2}$

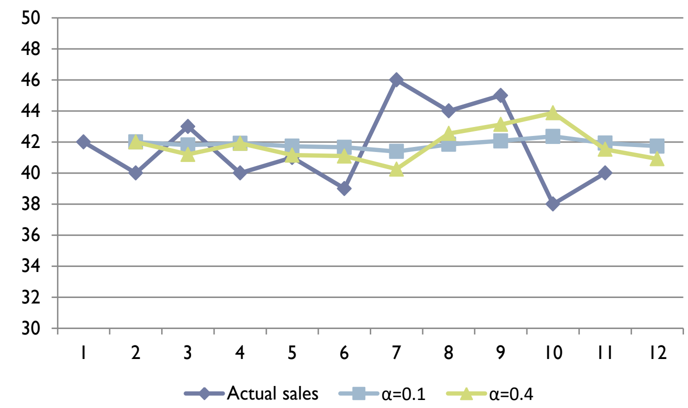
$$
\begin{aligned}
F_t &= \alpha A_{t-1} + (1-\alpha)F_{t-1} \\
&= \alpha A_{t-1} + (1-\alpha)\left(\alpha A_{t-2} + (1-\alpha)F_{t-2}\right) \\
&= \alpha A_{t-1} + \alpha(1-\alpha)A_{t-2} + (1-\alpha)^2 F_{t-2} \\
&= \alpha A_{t-1} + \alpha(1-\alpha)A_{t-2} + (1-\alpha)^2\left(\alpha A_{t-3} + (1-\alpha)F_{t-3}\right) \\
&= \alpha A_{t-1} + \alpha(1-\alpha)A_{t-2} + \alpha(1-\alpha)^2 A_{t-3} + \alpha(1-\alpha)^3 A_{t-4} + \ldots
\end{aligned}
$$



Forecasting Level

- <u>Simple exponential smoothing</u>:
  - Premise: The most recent observations might have the highest predictive value
  - A number of different approaches can be used to obtain a **starting forecast**

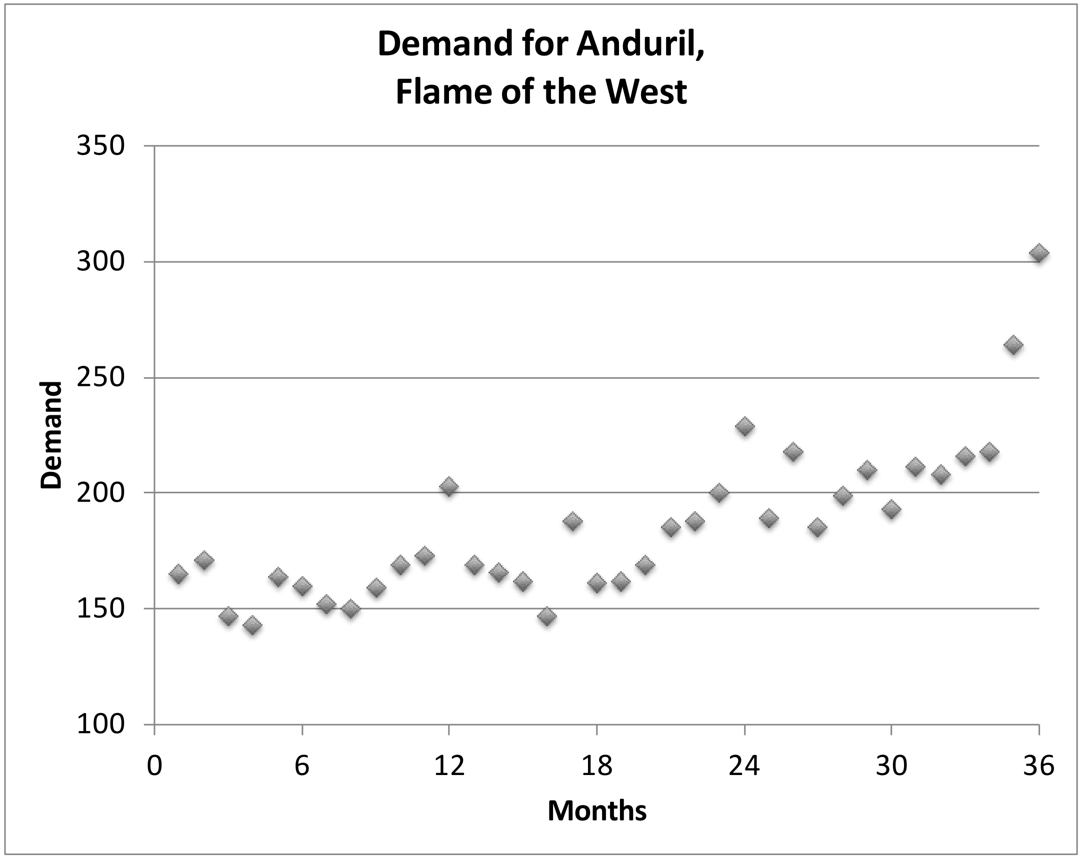
Period	Actual sales	α=0.1		α=0.4	
		Forecast	Error	Forecast	Error
1	42				
2	40	42.00	-2.00	42.00	-2.00
3	43	41.80	1.20	41.20	1.80
4	40	41.92	-1.92	41.92	-1.92
5	41	41.73	-0.73	41.15	-0.15
6	39	41.66	-2.66	41.09	-2.09
7	46	41.39	4.61	40.25	5.75
8	44	41.85	2.15	42.55	1.45
9	45	42.07	2.93	43.13	1.87
10	38	42.36	-4.36	43.88	-5.88
11	40	41.92	-1.92	41.53	-1.53
12		41.73		40.92	



- Lower value of α are used when the underlying average tend to be stable
- Higher value of α are used when the underlying average is susceptible to change

Time Series Data

- 과거 3년간의 수요 데이터

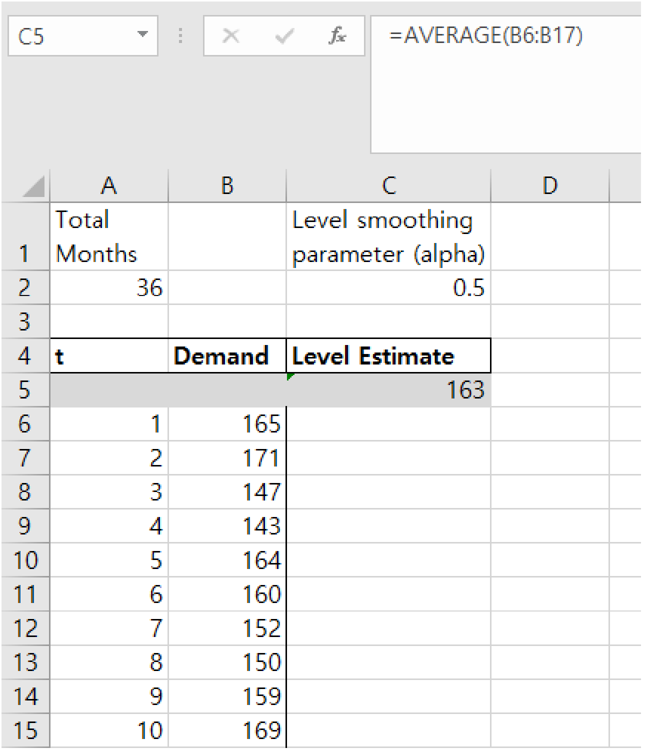


t	Demand
1	165
2	171
3	147
4	143
5	164
6	160
7	152
8	150
9	159
10	169
11	173
12	203
13	169
14	166
15	162
16	147
17	188
18	161
19	162
20	169
21	185
22	188
23	200
24	229
25	189
26	218
27	185
28	199
29	210
30	193
31	211
32	208
33	216
34	218
35	264
36	304

Simple Exponential Smoothing

- Simple Exponential Smoothing (Tab: SES)
- A1, A2, C1, C2 작성
- Timeseries Data 복사 A4부터 붙여넣기

- 5번 Row (Initial starting point) 삽입
- C4 작성
- Initial level은 첫 12개월의 평균값
- C5=AVERAGE(B6:B17)



	A	B	C	D
1	Total Months		Level smoothing parameter (alpha)	
2	36		0.5	
3				
4	**t**	**Demand**	**Level Estimate**	
5			163	
6	1	165		
7	2	171		
8	3	147		
9	4	143		
10	5	164		
11	6	160		
12	7	152		
13	8	150		
14	9	159		
15	10	169		

Simple Exponential Smoothing

- D4, E4 작성

- 1시점의 수요예측 (= level0)
- D6=C5
- 에러값 계산 $\left(A_{t-1} - F_{t-1}\right)$
- E6=B6-D6
- Level 예측 $F_t = F_{t-1} + \alpha\left(A_{t-1} - F_{t-1}\right)$
- C6=C5+C$2*E6

C6 =C5+C$2*E6

	A	B	C	D	E
1	Total Months		Level smoothing parameter (alpha)		
2	36		0.5		
3					
4	**t**	**Demand**	**Level Estimate**	**One-step Forecast**	**Forecast Error**
5			163		
6	1	165	164	163	2
7	2	171			
8	3	147			
9	4	143			
10	5	164			
11	6	160			
12	7	152			
13	8	150			
14	9	159			
15	10	169			

Simple Exponential Smoothing

- C6:E6을 36month까지 적용

- Column A에 37~48 작성
- 12개월 간 수요를 Level 36으로 작성
- B42=C$41
- B42를 B53까지 적용

B42 =C$41

	A	B	C	D	E
1	Total Months		Level smoothing parameter (alpha)		
2	36		0.5		
3					
4	**t**	**Demand**	**Level Estimate**	**One-step Forecast**	**Forecast Error**
34	29	210	203.9681271	197.9362542	12.06374585
35	30	193	198.4840635	203.9681271	-10.96812708
36	31	211	204.7420318	198.4840635	12.51593646
37	32	208	206.3710159	204.7420318	3.257968231
38	33	216	211.1855079	206.3710159	9.628984116
39	34	218	214.592754	211.1855079	6.814492058
40	35	264	239.296377	214.592754	49.40724603
41	36	304	271.6481885	239.296377	64.70362301
42	37	271.6482			
43	38	271.6482			
44	39	271.6482			
45	40	271.6482			
46	41	271.6482			
47	42	271.6482			
48	43	271.6482			
49	44	271.6482			
50	45	271.6482			
51	46	271.6482			
52	47	271.6482			
53	48	271.6482			

Simple Exponential Smoothing

- 예측 에러의 확인
- Sum of Squared Error (SSE)
- E1, F1, F4 작성
- F6=E6^2, F41까지 적용
- E2=SUM(F6:F41)

- Standard Error (Standard deviation)
- F2=SQRT(E2/(36-1))

- 68-95-99.7 rule
- P(-20.94 ≤ E ≤ 20.94) = 0.68
- P(-2σ ≤ E ≤ 2σ) = 0.95
- P(-3σ ≤ E ≤ 3σ) = 0.997

F2 =SQRT(E2/(36-1))

	A	B	C	D	E	F
1	Total Months		Level smoothing parameter (alpha)		SSE	Standard Error
2	36		0.50		15346.86	20.94
3						
4	**t**	**Demand**	**Level Estimate**	**One-step Forecast**	**Forecast Error**	**Squared Error**
5			163			
6	1	165	164	163	2	4
7	2	171	167.5	164	7	49
8	3	147	157.25	167.5	-20.5	420.25
9	4	143	150.125	157.25	-14.25	203.0625
10	5	164	157.0625	150.125	13.875	192.515625
11	6	160	158.53125	157.0625	2.9375	8.62890625
12	7	152	155.265625	158.53125	-6.53125	42.65722656
13	8	150	152.6328125	155.265625	-5.265625	27.72680664
14	9	159	155.8164063	152.6328125	6.3671875	40.54107666
15	10	169	162.4082031	155.8164063	13.18359375	173.8071442
16	11	173	167.7041016	162.4082031	10.59179688	112.186161
17	12	203	185.3520508	167.7041016	35.29589844	1245.800447
18	13	169	177.1760254	185.3520508	-16.35205078	267.3895648
19	14	166	171.5880127	177.1760254	-11.17602539	124.9035435
20	15	162	166.7940063	171.5880127	-9.588012695	91.92998745
21	16	147	156.8970032	166.7940063	-19.79400635	391.8026873
22	17	188	172.4485016	156.8970032	31.10299683	967.3964116
23	18	161	166.7242508	172.4485016	-11.44850159	131.0681886
24	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Simple Exponential Smoothing

- 최적의 smoothing parameter 찾기
- 목적함수

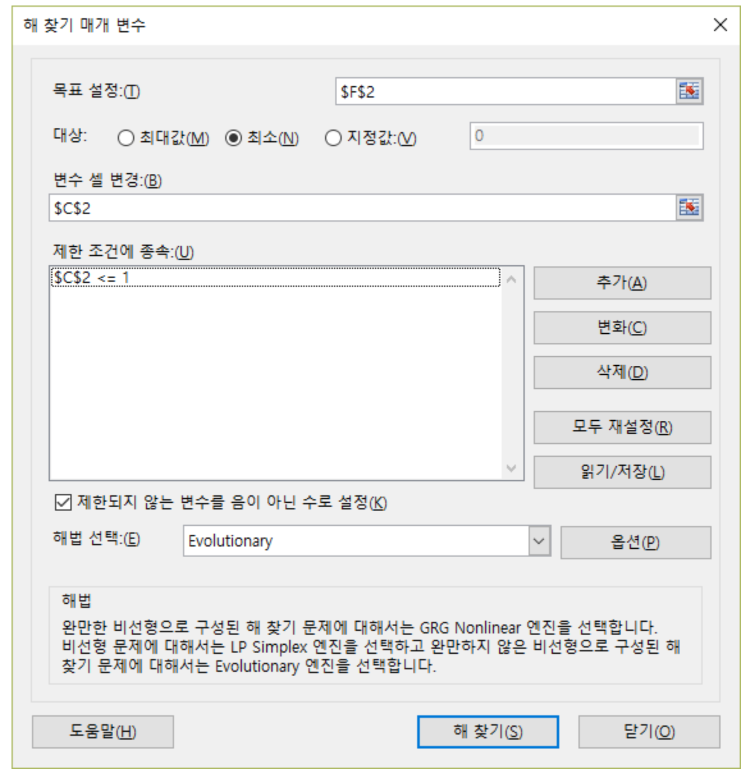
  Min F2=SQRT(E2/(36-1))

- 의사결정변수

  C2

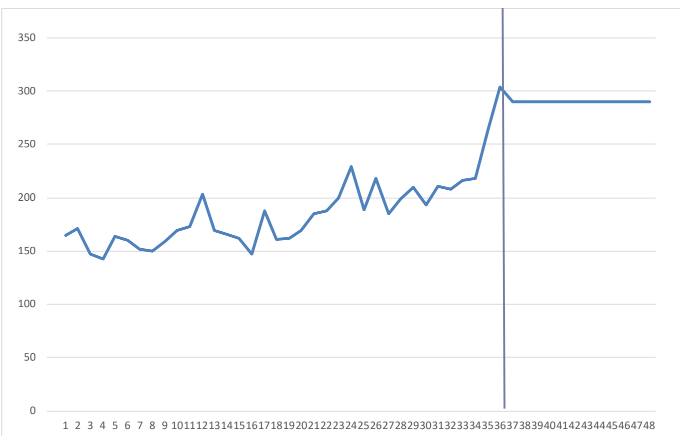
- 제약식

  0≤ C2 ≤1



Simple Exponential Smoothing

- 최적해 확인
  - Standard Error가 20.94 에서 20.39로 감소
  - 최적의 smoothing parameter : 0.73



F2 =SQRT(E2/(36-1))

	A	B	C	D	E	F
1	Total Months		Level smoothing parameter (alpha)		SSE	Standard Error
2	36		0.73		14555.77	20.39
3						
4	**t**	**Demand**	**Level Estimate**	**One-step Forecast**	**Forecast Error**	**Squared Error**
5			163			
6	1	165	164.4641792	163	2	4
7	2	171	169.2489855	164.4641792	6.535820847	42.71695414
8	3	147	152.9607351	169.2489855	-22.24898547	495.0173544
9	4	143	145.6685848	152.9607351	-9.960735116	99.21624406
10	5	164	159.0888228	145.6685848	18.33141524	336.0407846
11	6	160	159.7558861	159.0888228	0.911177217	0.830243921
12	7	152	154.0778827	159.7558861	-7.755886126	60.1537696
13	8	150	151.0925073	154.0778827	-4.077882736	16.6291276
14	9	159	156.8815003	151.0925073	7.90749271	62.52844096
15	10	169	165.7533276	156.8815003	12.11849972	146.8580355
16	11	173	171.0585409	165.7533276	7.24667239	52.51426073
17	12	203	194.4425502	171.0585409	31.94145907	1020.256807
18	13	169	175.8163244	194.4425502	-25.44255018	647.3233596
19	14	166	168.6298956	175.8163244	-9.81632439	96.36022452
20	15	162	163.7762181	168.6298956	-6.629895623	43.95551597
21	16	147	151.4945237	163.7762181	-16.77621814	281.4414952
22	17	188	178.2198024	151.4945237	36.50547629	1332.6498
23	18	161	165.6133645	178.2198024	-17.21980239	296.5215944
24	19	162	162.968058	165.6133645	-3.613364549	13.05640336
25	20	169	167.3839799	162.968058	6.031941974	36.38432398

Trend의 확인

- 선형회귀 (Linear Regression) 함수
  - =LINEST(B2:B37,A2:A37,TRUE,TRUE)
  - 통계 관련 데이터를 Array형태의 값을 return하며, Index 함수를 이용해 자세한 결과를 살펴볼 수 있음
  - Index 함수를 이용하지 않는 경우, 기본적으로 Slope를 리턴함

- 선형회귀 (Linear Regression) 함수의 통계값
  - Slope 계산
  - B39=INDEX(LINEST(B2:B37,A2:A37,TRUE,TRUE),1,1)

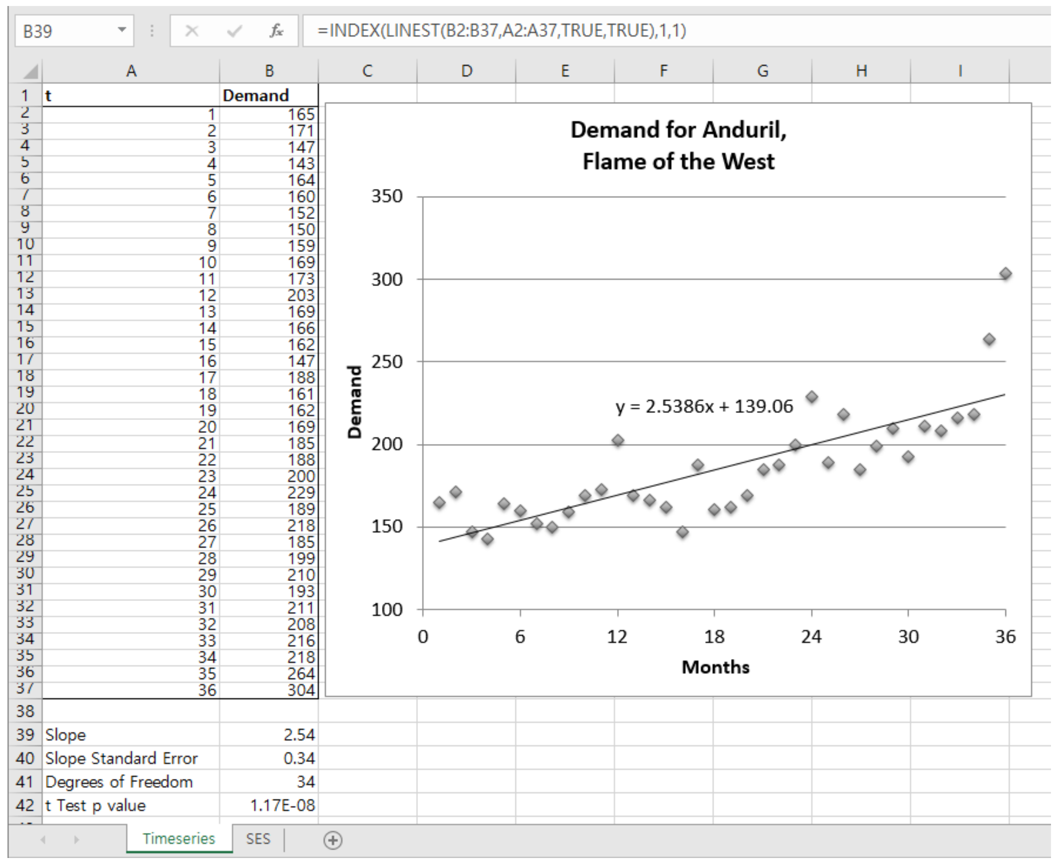
  - Slope Standard Error
  - B40=INDEX(LINEST(B2:B37,A2:A37,TRUE,TRUE),2,1)

  - Degree of Freedom 계산
  - B41=INDEX(LINEST(B2:B37,A2:A37,TRUE,TRUE),4,2)

  - T-test p-value 계산
  - B42=TDIST(ABS(B39/B40),B41,2)

Trend의 확인

- 선형회귀 (Linear Regression) 함수 결과
  - 그래프의 추세선 이용 가능함
  - 매달 2.54의 수요 증가
  - P-value가 0에 가까움
    - Slope가 0인 상황에서
    이와 같은 데이터가
    발생할 확률은 거의 없음
    - 통계적으로 Trend가 있음



	A	B
1	t	Demand
2	1	165
3	2	171
4	3	147
5	4	143
6	5	164
7	6	160
8	7	152
9	8	150
10	9	159
11	10	169
12	11	173
13	12	203
14	13	169
15	14	166
16	15	162
17	16	147
18	17	188
19	18	161
20	19	162
21	20	169
22	21	185
23	22	188
24	23	200
25	24	229
26	25	189
27	26	218
28	27	185
29	28	199
30	29	210
31	30	193
32	31	211
33	32	208
34	33	216
35	34	218
36	35	264
37	36	304
38		
39	Slope	2.54
40	Slope Standard Error	0.34
41	Degrees of Freedom	34
42	t Test p value	1.17E-08

Holt's Trend-corrected Exponential Smoothing

- Seasonality는 존재하지 않고, Level과 Trend가 존재하는 경우

Systematic component of demand = level + trend

- 초기값의 예측을 위해서 linear regression이 이용될 수 있음

$$D_t = at + b \; (T_0 = a, \; L_0 = b)$$

- 예측치 update 과정

$$L_1 = L_0 + T_0 + \alpha\,(D_1 - (L_0 + T_0)) = \alpha\,D_1 + (1 - \alpha)(L_0 + T_0)$$

$$T_1 = T_0 + \alpha\gamma\,(D_1 - (L_0 + T_0)) = T_0 + \gamma\,(L_1 - (L_0 + T_0))$$

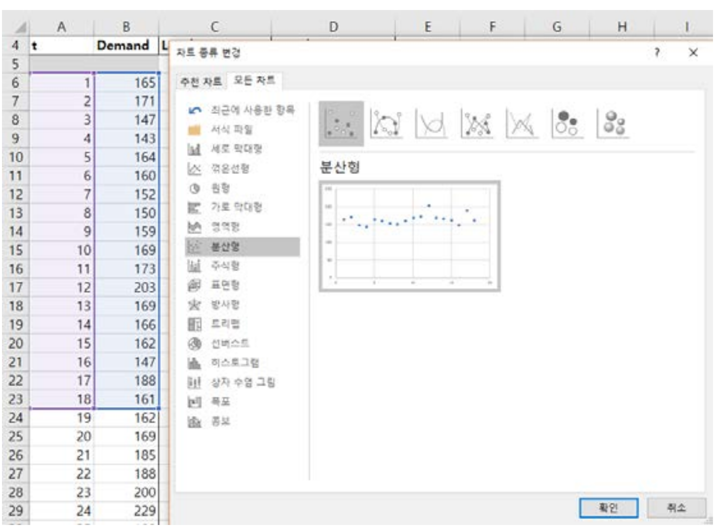
$$= \gamma\,(L_1 - L_0) + (1 - \gamma)(T_0)$$

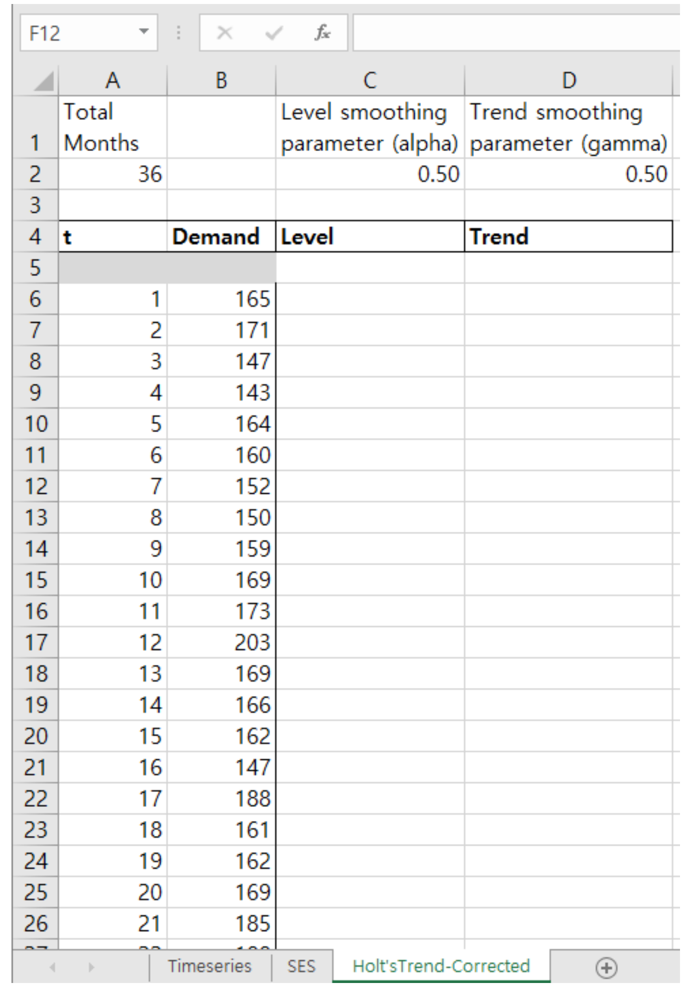
$$F_2 = L_1 + T_1 \quad (F_{t+1} = L_t + T_t)$$

Holt’s Trend-corrected Exponential Smoothing

- Holt’s model (Tab: Holt'sTrend-Corrected)
- A1, A2, C1, C2, D1, D2 작성
- SES Demand 데이터 복사 A4부터 붙여넣기
- C4, D4 작성

- 초기 level, trend 계산: 첫 18개월 데이터
  - 삽입>추천차트(모든차트)>분산형





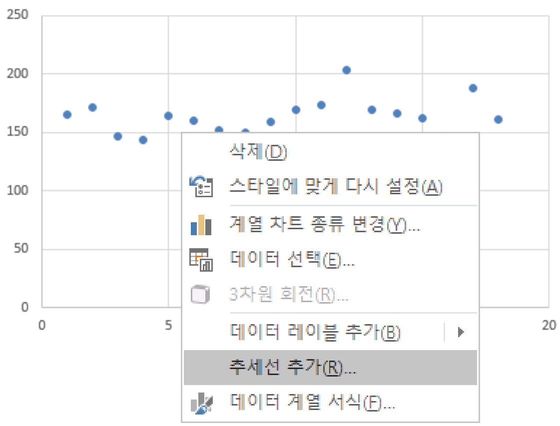
F12

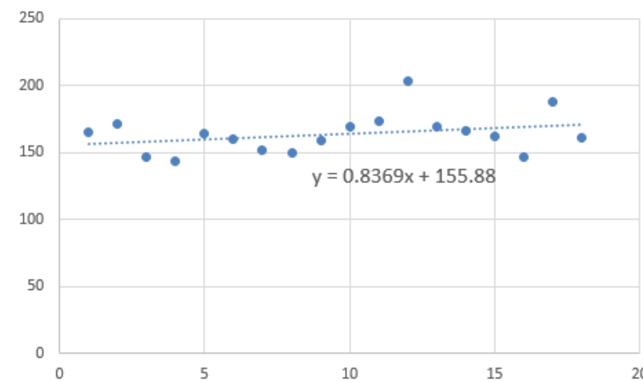
	A	B	C	D
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)
2	36		0.50	0.50
3				
4	**t**	**Demand**	**Level**	**Trend**
5				
6	1	165		
7	2	171		
8	3	147		
9	4	143		
10	5	164		
11	6	160		
12	7	152		
13	8	150		
14	9	159		
15	10	169		
16	11	173		
17	12	203		
18	13	169		
19	14	166		
20	15	162		
21	16	147		
22	17	188		
23	18	161		
24	19	162		
25	20	169		
26	21	185		

Timeseries | SES | Holt'sTrend-Corrected

Holt's Trend-corrected Exponential Smoothing

- 초기 level, trend 계산: 첫 18개월 데이터
  - 추세선 추가, 수식을 차트에 표시
  - $Y=0.8369x + 155.88$ ($L_0 = 155.88$, $T_0 = 0.8369$)





Holt's Trend-corrected Exponential Smoothing

- 초기 예측값
  - E4, F4 작성
  - 1시점 예측 값 작성 ($F_1 = L_0 + T_0$)
  - E6 =C5+D5
- Forecast Error 계산
  - F6=B6-E6 (D$_1$-($L_0 + T_0$))
- Level, Trend 업데이트
  - $L_1 = L_0 + T_0 + \alpha$ (D$_1$-($L_0 + T_0$))
  - C6=C5+D5+C$2*F6
  - $T_1 = T_0 + \alpha\gamma$ (D$_1$-($L_0 + T_0$))
  - D6=D5+D$2*C$2*F6
- t = 36까지 적용

C6 | =C5+D5+C$2*F6

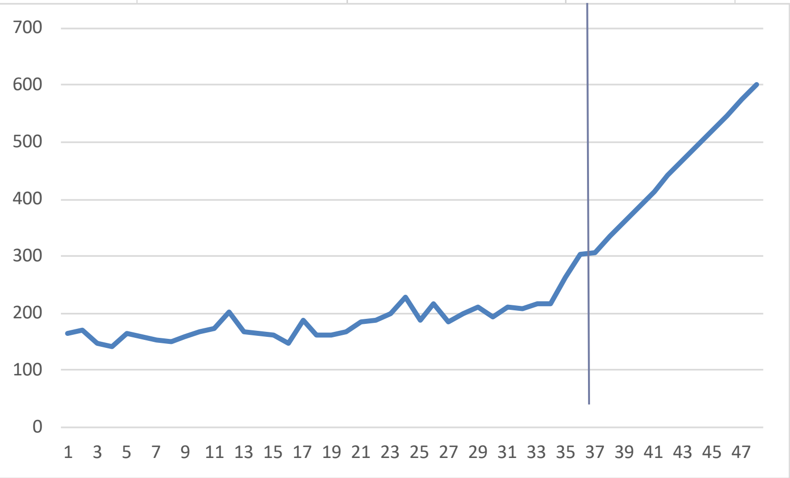
	A	B	C	D	E	F
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)		
2	36		0.50	0.50		
3						
4	**t**	**Demand**	**Level**	**Trend**	**One-step Forecast**	**Forecast Error**
5			155.88	0.8369		
6	1	165	160.85845	2.907675	156.7169	8.2831
7	2	171	167.3830625	4.71614375	163.766125	7.233875
8	3	147	159.5496031	-1.558657813	172.0992063	-25.09920625
9	4	143	150.4954727	-5.306394141	157.9909453	-14.99094531
10	5	164	154.5945393	-0.60366377	145.1890785	18.81092148
11	6	160	156.9954377	0.898617358	153.9908755	6.009124512
12	7	152	154.9470276	-0.574896417	157.8940551	-5.894055103
13	8	150	152.1860656	-1.667929201	154.3721311	-4.372131134
14	9	159	154.7590682	0.452536708	150.5181364	8.481863634
15	10	169	162.1058024	3.899635485	155.2116049	13.78839511
16	11	173	169.502719	5.648276002	166.0054379	6.99456207
17	12	203	189.0754975	12.61052726	175.150995	27.84900503
18	13	169	185.3430124	4.439021074	201.6860247	-32.68602474
19	14	166	177.8910167	-1.506487287	189.7820334	-23.78203345
20	15	162	169.1922647	-5.102619646	176.3845294	-14.38452944
21	16	147	155.5448225	-9.375030914	164.0896451	-17.08964507
22	17	188	167.0848958	1.08252118	146.1697916	41.83020838
23	18	161	164.5837085	-0.709333067	168.167417	-7.167416991
24	19	162	162.9371877	-1.177926924	163.8743754	-1.874375428
25	20	169	165.3796304	0.632257878	161.7592608	7.24073921

Holt's Trend-corrected Exponential Smoothing

- 미래 데이터 예측 (t=37~48)
  - $D_t = L_{36} + T_{36} * t$
  - 37 시점의 수요예측 값
  - B42=C$41+(A42-A$41)*D$41
  - B43~B53에 적용

B42 =C$41+(A42-A$41)*D$41

	A	B	C	D	E	F
31	26	218	216.5306944	5.159860734	215.0613888	2.938611247
32	27	185	203.3452776	-4.012778044	221.6905551	-36.69055511
33	28	199	199.1662498	-4.095902922	199.3324995	-0.332499511
34	29	210	202.5351734	-0.36348963	195.0703468	14.92965317
35	30	193	197.5858419	-2.656410577	202.1716838	-9.171683787
36	31	211	202.9647157	1.361231594	194.9294313	16.07056868
37	32	208	206.1629736	2.279744781	204.3259473	3.674052748
38	33	216	212.2213592	4.169065179	208.4427184	7.557281593
39	34	218	217.1952122	4.571459083	216.3904244	1.609575617
40	35	264	242.8833356	15.12979126	221.7666713	42.23332873
41	36	304	281.0065635	26.62650954	258.0131269	45.9868731
42	37	307.6331				
43	38	334.2596				
44	39	360.8861				
45	40	387.5126				
46	41	414.1391				
47	42	440.7656				
48	43	467.3921				
49	44	494.0186				
50	45	520.6451				
51	46	547.2717				
52	47	573.8982				
53	48	600.5247				



Holt's Trend-corrected Exponential Smoothing

- 예측 에러의 확인
- Sum of Squared Error (SSE)
- F1, G1, G4 작성
- G6=F6^2, G41까지 적용
- F2=SUM(G6:G41)

- Standard Error (Standard deviation)
- G2=SQRT(F2/(36-2))

I5

	A	B	C	D	E	F	G
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)		SSE	Standard Error
2	36		0.50	0.50		15315.32	21.22
3							
4	**t**	**Demand**	**Level**	**Trend**	**One-step Forecast**	**Forecast Error**	**Squared Error**
5			155.88	0.8369			
6	1	165	160.85845	2.907675	156.7169	8.2831	68.60974561
7	2	171	167.3830625	4.71614375	163.766125	7.233875	52.32894752
8	3	147	159.5496031	-1.558657813	172.0992063	-25.09920625	629.9701544
9	4	143	150.4954727	-5.306394141	157.9909453	-14.99094531	224.7284414
10	5	164	154.5945393	-0.60366377	145.1890785	18.81092148	353.8507671
11	6	160	156.9954377	0.898617358	153.9908755	6.009124512	36.1095774
12	7	152	154.9470276	-0.574896417	157.8940551	-5.894055103	34.73988555
13	8	150	152.1860656	-1.667929201	154.3721311	-4.372131134	19.11553065
14	9	159	154.7590682	0.452536708	150.5181364	8.481863634	71.9420107
15	10	169	162.1058024	3.899635485	155.2116049	13.78839511	190.1198397
16	11	173	169.502719	5.648276002	166.0054379	6.99456207	48.92389855
17	12	203	189.0754975	12.61052726	175.150995	27.84900503	775.5670813
18	13	169	185.3430124	4.439021074	201.6860247	-32.68602474	1068.376214
19	14	166	177.8910167	-1.506487287	189.7820334	-23.78203345	565.5851149
20	15	162	169.1922647	-5.102619646	176.3845294	-14.38452944	206.9146871
21	16	147	155.5448225	-9.375030914	164.0896451	-17.08964507	292.0559687
22	17	188	167.0848958	1.08252118	146.1697916	41.83020838	1749.766333
23	18	161	164.5837085	-0.709333067	168.167417	-7.167416991	51.37186633
24	19	162	162.9371877	-1.177926924	163.8743754	-1.874375428	3.513283246
25	20	169	165.3796304	0.632257878	161.7592608	7.24073921	52.42830431
26	21	185	175.5059441	5.37928581	166.0118883	18.98811173	360.548387

Holt's Trend-corrected Exponential Smoothing

- 최적의 smoothing parameter 찾기
- 목적함수

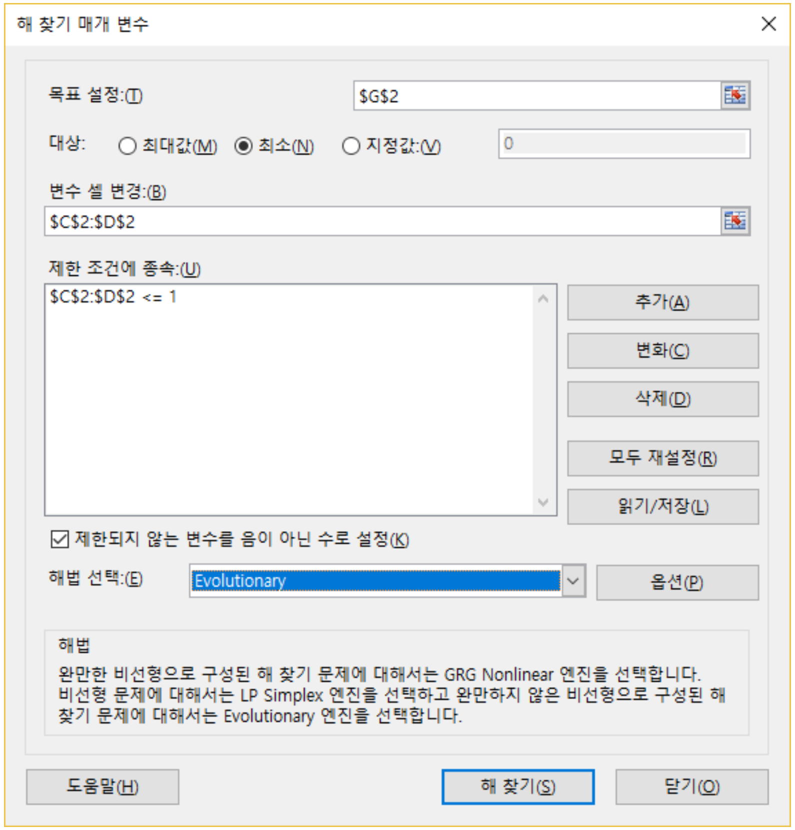
  Min G2=SQRT(F2/(36-2))

- 의사결정변수

  C2, D2

- 제약식

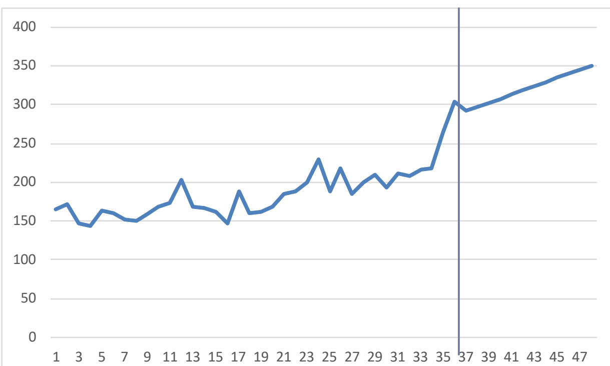
  $0 \le C2, D2 \le 1$



Simple Exponential Smoothing

- 최적해 확인
  - Standard Error가 21.22에서 20.36으로 감소
  - 최적의 smoothing parameter

    : 0.66, 0.05



G2 =SQRT(F2/(36-2))

	A	B	C	D	E	F	G
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)		SSE	Standard Error
2	36		0.66	0.05		14097.28	20.36
3							
4	**t**	**Demand**	**Level**	**Trend**	**One-step Forecast**	**Forecast Error**	**Squared Error**
5			155.88	0.8369			
6	1	165	162.1762916	1.126887489	156.7169	8.2831	68.60974561
7	2	171	168.3761541	1.396349618	163.3031791	7.696820916	59.24105221
8	3	147	154.7631455	0.599094827	169.7725037	-22.77250373	518.586926
9	4	143	147.2142871	0.166298529	155.3622403	-12.36224029	152.8249849
10	5	164	158.3344424	0.748136515	147.3805857	16.61941433	276.2049328
11	6	160	159.6872512	0.780255001	159.0825789	0.917421057	0.841661396
12	7	152	154.8865725	0.483811537	160.4675062	-8.467506205	71.69866134
13	8	150	151.8307637	0.295796884	155.370384	-5.370384008	28.84102439
14	9	159	156.6568448	0.536432811	152.1265605	6.873439458	47.24416998
15	10	169	164.9750889	0.949780684	157.1932776	11.80672238	139.3986934
16	11	173	170.5880884	1.197477718	165.9248696	7.075130426	50.05747054
17	12	203	192.3590009	2.29028057	171.7855661	31.21443391	974.3408846
18	13	169	177.7438389	1.392311123	194.6492815	-25.6492815	657.8856416
19	14	166	170.4781129	0.932420605	179.13615	-13.13614999	172.5584366
20	15	162	165.2080504	0.602962192	171.4105335	-9.410533526	88.55814124
21	16	147	153.4126733	-0.055602634	165.8110126	-18.81101263	353.8541963
22	17	188	176.190227	1.15723025	153.3570707	34.6429293	1200.132551
23	18	161	166.5728474	0.584913355	177.3474573	-16.34745726	267.239359
24	19	162	163.7582804	0.404342543	167.1577608	-5.157760775	26.60249621
25	20	169	167.3509384	0.573696869	164.162623	4.837377049	23.40021672

Seasonality의 확인

- Error term에 대한 확인이 필요 (Holt's Autocorrelation)
  - Error가 random하게 발생하면 괜찮음
  - 반복되는 패턴이 있다면, Seasonality도 고려해야 함

  - Holt'sTrend-Corrected에서 t 와 Forecast Error (값) 복사
  - 평균 Error 계산
  - B38=AVERAGE(B2:B37)

B38 =AVERAGE(B2:B37)

	A	B	C	D	E
1	**t**	**Forecast Error**			
29	28	2.398381705			
30	29	10.33952364			
31	30	-15.31532337			
32	31	11.47512284			
33	32	-0.793754017			
34	33	6.051575393			
35	34	2.173285023			
36	35	44.77509032			
37	36	51.73048861			
38		3.576404935			
39					

Seasonality의 확인

- 평균으로부터의 차이 계산
  - C1작성
  - 평균 Error로 부터의 차이 계산
  - C2=B2-B$38
  - t=36 까지 적용

  - Sum of squared deviation

$$\sum_i (x_i - \bar{x})^2$$

  - C38=SUMPRODUCT($C2:$C37,C2:C37)

C38 =SUMPRODUCT($C2:$C37,C2:C

	A	B	C	D
1	t	Forecast Error	Deviations from mean	
23	22	7.649493578	4.073088643	
24	23	13.16840931	9.59200438	
25	24	31.58878753	28.0123826	
26	25	-32.23761519	-35.81402013	
27	26	16.13259105	12.55618611	
28	27	-29.94280243	-33.51920736	
29	28	2.398381705	-1.17802323	
30	29	10.33952364	6.763118704	
31	30	-15.31532337	-18.89172831	
32	31	11.47512284	7.898717909	
33	32	-0.793754017	-4.370158951	
34	33	6.051575393	2.475170458	
35	34	2.173285023	-1.403119912	
36	35	44.77509032	41.19868538	
37	36	51.73048861	48.15408368	
38		3.576404935	13636.81662	
39				
40				

Seasonality의 확인

- Lagged Error Deviation Matrix계산
  - 1 month lag: 1달 차이로 Error 반복
  - D1 작성
  - D2는 빈칸
  - D3 = C2
  - D37까지 적용

D3 =C2

	A	B	C	D
1	t	Forecast Error	Deviations from mean	1
2	1	8.2831	4.706695065	
3	2	7.696820916	4.120415981	4.706695
4	3	-22.77250373	-26.34890866	4.120416
5	4	-12.36224029	-15.93864522	-26.3489
6	5	16.61941433	13.0430094	-15.9386
7	6	0.917421057	-2.658983878	13.04301
8	7	-8.467506205	-12.04391114	-2.65898
9	8	-5.370384008	-8.946788943	-12.0439
10	9	6.873439458	3.297034523	-8.94679

D37 =C36

	A	B	C	D
1	t	Forecast Error	Deviations from mean	1
32	31	11.47512284	7.898717909	-18.8917
33	32	-0.793754017	-4.370158951	7.898718
34	33	6.051575393	2.475170458	-4.37016
35	34	2.173285023	-1.403119912	2.47517
36	35	44.77509032	41.19868538	-1.40312
37	36	51.73048861	48.15408368	41.19869
38		3.576404935	13636.81662	

Seasonality의 확인

- Lagged Error Deviation Matrix계산
  - D1:D37 선택 후, O column까지 Drag… 자동으로 생성

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	t	Forecast Error	Deviations from mea	1	2	3	4	5	6	7	8	9	10	11	12
2	1	8.2831	4.706695065												
3	2	7.696820916	4.120415981	4.706695	0	0	0	0	0	0	0	0	0	0	0
4	3	-22.77250373	-26.34890866	4.120416	4.706695	0	0	0	0	0	0	0	0	0	0
5	4	-12.36224029	-15.93864522	-26.3489	4.120416	4.706695	0	0	0	0	0	0	0	0	0
6	5	16.61941433	13.0430094	-15.9386	-26.3489	4.120416	4.706695	0	0	0	0	0	0	0	0
7	6	0.917421057	-2.658983878	13.04301	-15.9386	-26.3489	4.120416	4.706695	0	0	0	0	0	0	0
8	7	-8.467506205	-12.04391114	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695	0	0	0	0	0	0
9	8	-5.370384008	-8.946788943	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695	0	0	0	0	0
10	9	6.873439458	3.297034523	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695	0	0	0	0
11	10	11.80672238	8.230317446	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695	0	0	0
12	11	7.075130426	3.498725491	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695	0	0
13	12	31.21443391	27.63802898	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695	0
14	13	-25.6492815	-29.22568644	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416	4.706695
15	14	-13.13614999	-16.71255493	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489	4.120416
16	15	-9.410533526	-12.98693846	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386	-26.3489
17	16	-18.81101263	-22.38741757	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301	-15.9386
18	17	34.6429293	31.06652437	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898	13.04301
19	18	-16.34745726	-19.9238622	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439	-2.65898
20	19	-5.157760775	-8.73416571	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679	-12.0439
21	20	4.837377049	1.260972114	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035	-8.94679
22	21	17.07536474	13.49895981	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317	3.297035
23	22	7.649493578	4.073088643	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725	8.230317
24	23	13.16840931	9.59200438	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803	3.498725
25	24	31.58878753	28.0123826	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257	27.63803
26	25	-32.23761519	-35.81402013	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257
27	26	16.13259105	12.55618611	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126
28	27	-29.94280243	-33.51920736	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869
29	28	2.398381705	-1.17802323	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874
30	29	10.33952364	6.763118704	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652

Seasonality의 확인

- Lagged Column를 대상으로 sumproduct 계산
  - 만일 Column C와 특정 Column이 동기화 된다면
    (둘다 평균 이상, 둘다 평균이하) Sumproduct의 값은 큰 양수가 될 것임
    (하나는 평균이상, 다른 하나는 평균이하) Sumproduct의 값은 큰 음수가 될 것임
  - C38 선택 후, O column까지 Drag… 자동으로 생성

O38 =SUMPRODUCT($C2:$C37,O2:O37)

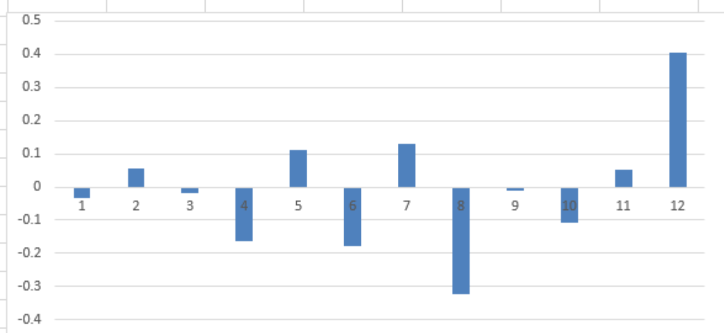
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	t	Forecast Error	Deviations from mean	1	2	3	4	5	6	7	8	9	10	11	12
26	25	-32.23761519	-35.81402013	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126	-29.2257
27	26	16.13259105	12.55618611	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869	-16.7126
28	27	-29.94280243	-33.51920736	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874	-12.9869
29	28	2.398381705	-1.17802323	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652	-22.3874
30	29	10.33952364	6.763118704	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239	31.06652
31	30	-15.31532337	-18.89172831	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417	-19.9239
32	31	11.47512284	7.898717909	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972	-8.73417
33	32	-0.793754017	-4.370158951	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896	1.260972
34	33	6.051575393	2.475170458	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089	13.49896
35	34	2.173285023	-1.403119912	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089
36	35	44.77509032	41.19868538	-1.40312	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004
37	36	51.73048861	48.15408368	41.19869	-1.40312	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238
38		3.576404935	13636.81662	-474.07	753.3627	-229.307	-2223.64	1544.284	-2442.53	1790.525	-4382.61	-167.7	-1460.42	733.0618	5512.801

Seasonality의 확인

- Autocorrelation의 계산
  - C40 작성
  - 1 month lag의 Autocorrelation
  - D40=D38/$C38
  - C40 선택 후, O column까지 Drag… 자동으로 생성

D40 =D38/$C38

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	t	**Forecast Error**	**Deviations from mean**	1	2	3	4	5	6	7	8	9	10	11	12
35	34	2.173285023	-1.403119912	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089
36	35	44.77509032	41.19868538	-1.40312	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004
37	36	51.73048861	48.15408368	41.19869	-1.40312	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238
38		3.576404935	13636.81662	-474.07	753.3627	-229.307	-2223.64	1544.284	-2442.53	1790.525	-4382.61	-167.7	-1460.42	733.0618	5512.801
39															
40			Autocorrelation	-0.03476	0.055245	-0.01682	-0.16306	0.113244	-0.17911	0.131301	-0.32138	-0.0123	-0.10709	0.053756	0.404259



Seasonality의 확인

- Autocorrelation의 계산
  - 일반적으로 $2/\sqrt{n}$ 이내의 값은 문제 없다고 판단함 (n: 데이터의 수)
  - $\frac{2}{\sqrt{36}} = 0.333$
  - C41, C42작성
  - D41=2/SQRT(36)
  - D42=-2/SQRT(36)

D41 =2/SQRT(36)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	t	Forecast Error	Deviations from mean	1	2	3	4	5	6	7	8	9	10	11	12
35	34	2.173285023	-1.403119912	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004	4.073089
36	35	44.77509032	41.19868538	-1.40312	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238	9.592004
37	36	51.73048861	48.15408368	41.19869	-1.40312	2.47517	-4.37016	7.898718	-18.8917	6.763119	-1.17802	-33.5192	12.55619	-35.814	28.01238
38		3.576404935	13636.81662	-474.07	753.3627	-229.307	-2223.64	1544.284	-2442.53	1790.525	-4382.61	-167.7	-1460.42	733.0618	5512.801
39															
40			Autocorrelation	-0.03476	0.055245	-0.01682	-0.16306	0.113244	-0.17911	0.131301	-0.32138	-0.0123	-0.10709	0.053756	0.404259
41			Upper cutoff	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333
42			Lower cutoff	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333
43															

12개월 단위의 Seasonality 존재

Multiplicative Holt-Winters Exponential Smoothing

- Level, Trend, Seasonality가 모두 존재하는 경우

Systematic component = (level + trend) x seasonal factor

- L_t , T_t , S_{t+1-p} , S_{t+2-p} … S_t를 이용해 미래 수요를 예측 (p: Periodicity)

$$F_{t+1} = (L_t + T_t)S_{t+1-p} \quad \dots \quad F_{t+p} = (L_t + pT_t)S_t$$

- 추정치 update 과정

$$L_1 = L_0 + T_0 + \alpha\,(D_1-(L_0 + T_0)*S_{1-p})/\,S_{1-p} = \alpha\,(D_1\,/\,S_{1-p}) + (1-\alpha)(L_0 + T_0)$$

$$T_1 = T_0 + \alpha\gamma\,(D_1-(L_0 + T_0)*S_{1-p})/\,S_{1-p} = T_0 + \gamma\,(L_1 - (L_0 + T_0))$$

$$= \gamma\,(L_1 - L_0) + (1-\gamma)(T_0)$$

$$S_1 = S_{1-p} + \delta(1-\alpha)(D_1-(L_0 + T_0)*S_{1-p})/\,(L_0 + T_0)$$

$$\approx \delta(D_1\,/L_1) + (1-\delta)S_{1-p}$$

Multiplicative Holt-Winters Exponential Smoothing

- Level과 Trend의 초기값 추정하기
  - Periodicity: the number of period after which the seasonal cycle repeats
    → p = 12 (the pattern repeats every year)
  - Deseasonalized demand: the demand in the absence of seasonal fluctuation

P=5 (홀수)

1 2 3 4 5

↑ (mid-point)

(p+1)/2 = 6/2 = 3

$$\bar{D}_3 = \sum_{i=1}^{5} D_i / 5$$

$$\bar{D}_t = \sum_{i=t-[(p-1)/2]}^{t+[(p-1)/2]} D_i / p$$

P=4 (짝수)

1 2 3 4

↑ (mid-point) $\bar{d}^1$

(p+1)/2 = 5/2 = 2.5

2 3 4 5

↑ (mid-point) $\bar{d}^2$

1+ (p+1)/2 = 1+ 5/2 = 3.5

$$\bar{D}_3 = \left(\bar{d}^1 + \bar{d}^2\right)/2 = \left(\sum_{i=1}^{4} D_i / 4 + \sum_{i=2}^{5} D_i / 4\right)/2$$

$$= \left(D_1 + D_5 + 2\sum_{i=2}^{4} D_i\right)/\left(2*4\right)$$

$$\bar{D}_t = \left(D_{t-(p/2)} + D_{t+(p/2)} + 2\sum_{i=t-(p/2)+1}^{t+(p/2)-1} D_i\right)/\left(2p\right)$$

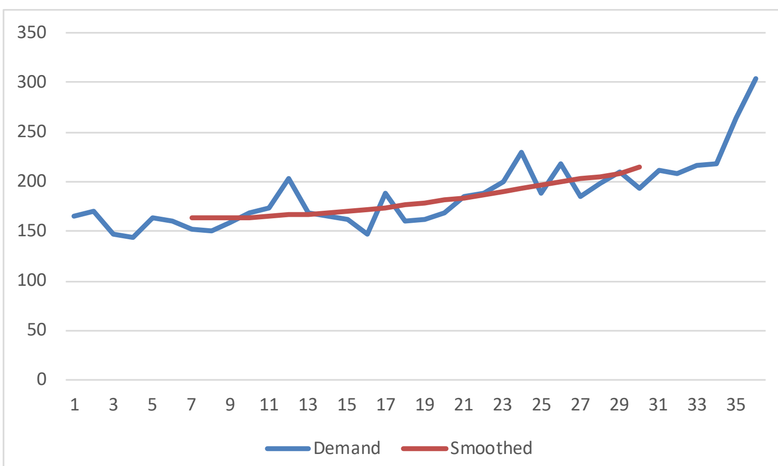
Multiplicative Holt-Winters Exponential Smoothing

- Level과 Trend의 초기값 추정하기(HoltWintersInitial)
- 과거 36개월의 수요 데이터 복사 (A, B column)
- C1 작성

- Smoothed Demand 계산
- 기간이 짝수(p=12)이므로…
- C8=(AVERAGE(B3:B14)+AVERAGE(B2:B13))/2

C8 =(AVER

	A	B	C
1	t	Demand	Smoothed
2	1	165	
3	2	171	
4	3	147	
5	4	143	
6	5	164	
7	6	160	
8	7	152	163.17
9	8	150	163.13
10	9	159	163.54
11	10	169	164.33
…			
29	28	199	205.08
30	29	210	209.00
31	30	193	214.79
32	31	211	
33	32	208	
34	33	216	
35	34	218	
36	35	264	
37	36	304	



Multiplicative Holt-Winters Exponential Smoothing

- Seasonal Factor 계산
- D1, E1, F1 작성

- 각 시점의 Seasonal Factor
  - D8=B8/C8

- 초기 Seasonal Factor
  - 1월: E2 =AVERAGE(D14,D26)

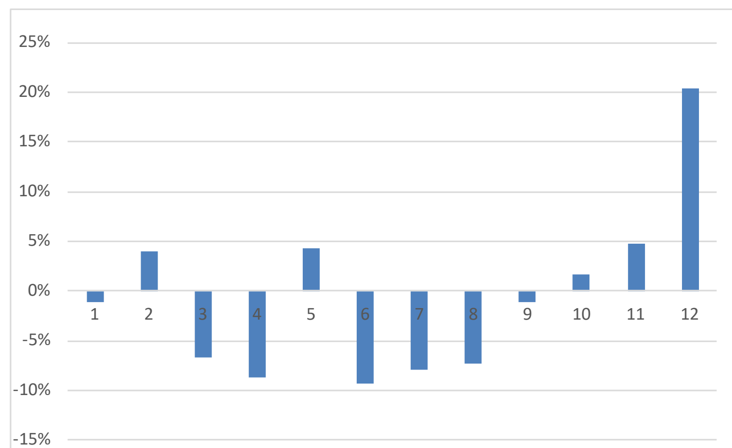
    …
  - 6월: E7 =AVERAGE(D19,D31)
  - 7월: E8 =AVERAGE(D8,D20)

    …
  - 12월: E13 =AVERAGE(D13,D25)

- % Skew (%로 표기)
  - F2 =E2-1

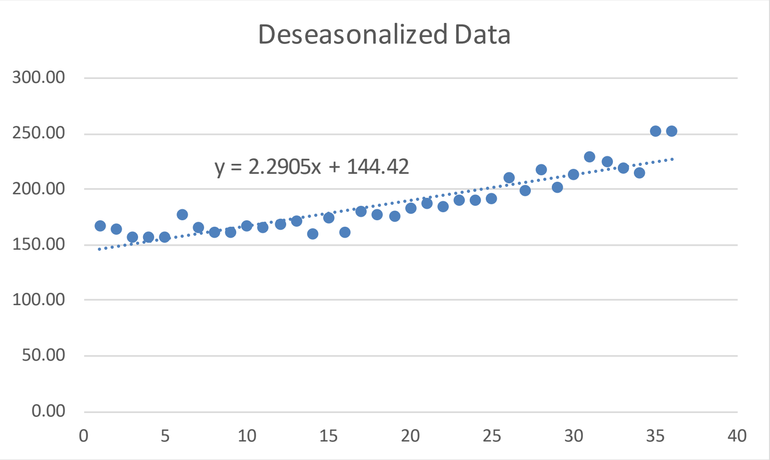
E2 =AVERAGE(D14,D26)

	A	B	C	D	E	F
1	t	Demand	Smoothed	Seasonal Factor Estimate	Initial Seasonal Factors	% Skew
2	1	165			0.99	-1%
3	2	171			1.04	4%
4	3	147			0.93	-7%
5	4	143			0.91	-9%
6	5	164			1.04	4%
7	6	160			0.91	-9%
8	7	152	163.17	0.93	0.92	-8%
9	8	150	163.13	0.92	0.93	-7%
10	9	159	163.54	0.97	0.99	-1%
11	10	169	164.33	1.03	1.02	2%
12	11	173	165.50	1.05	1.05	5%
13	12	203	166.54	1.22	1.20	20%
14	13	169	167.00	1.01		



Multiplicative Holt-Winters Exponential Smoothing

- Initial Seasonal Factor의 적용
  - G1,H1작성
  - E2:E13을 G2:G37에 값으로 복사
  - H2=B2/G2
  - H37까지 적용
  - 그래프와 추세선 확인



H2 =B2/G2

	A	B	C	D	E	F	G	H
1	t	Demand	Smoothed	Seasonal Factor Estimate	Initial Seasonal Factors	% Skew	Initial Seasonal Factors x3	Deseasonalized Data
2	1	165			0.99	-1%	0.99	166.96
3	2	171			1.04	4%	1.04	164.51
4	3	147			0.93	-7%	0.93	157.57
5	4	143			0.91	-9%	0.91	156.70
6	5	164			1.04	4%	1.04	157.24
7	6	160			0.91	-9%	0.91	176.51
8	7	152	163.17	0.93	0.92	-8%	0.92	165.07
9	8	150	163.13	0.92	0.93	-7%	0.93	161.88
10	9	159	163.54	0.97	0.99	-1%	0.99	160.85
11	10	169	164.33	1.03	1.02	2%	1.02	166.31
12	11	173	165.50	1.05	1.05	5%	1.05	165.07
13	12	203	166.54	1.22	1.20	20%	1.20	168.60
14	13	169	167.00	1.01			0.99	171.01
15	14	166	168.21	0.99			1.04	159.70
16	15	162	170.08	0.95			0.93	173.65
17	16	147	171.96	0.85			0.91	161.08

Multiplicative Holt-Winters Exponential Smoothing

- 추정치 업데이트 (HoltWintersSeasonal)를 위한 초기 데이터 정리
  - Holt'sTrend-Corrected에서 A, B Column 복사
  - C1, D1, E1 작성
  - C2, D2, E2 작성 (0.5)
  - Row5 ~ 15 삽입: 시점 -11~0까지 작성
  - Initial Level 작성: C16
  - Initial Trend 작성: D16
  - Initial Seasonality 작성: E5~E16

    : HoltWintersInitial의 E column

E16 | 1.2040049076184

	A	B	C	D	E
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)	Seasonal smoothing parameter (delta)
2	36		0.5	0.5	0.5
3					
4	**t**	**Demand**	**Level**	**Trend**	**Seasonal Adjustment**
5	-11				0.988233399
6	-10				1.039459514
7	-9				0.932933292
8	-8				0.912597756
9	-7				1.043010605
10	-6				0.906442452
11	-5				0.920837589
12	-4				0.926620944
13	-3				0.988490753
14	-2				1.016201453
15	-1				1.048052656
16	0		144.42	2.2905	1.204004908
17	1	165			
18	2	171			
19	3	147			
20	4	143			
21	5	164			

Multiplicative Holt-Winters Exponential Smoothing

- 추정치 업데이트
  - F4, G4 작성
  - t=1의 수요예측
  - F17=(C16+D16)*E5
  - t=1의 수요예측에러
  - G17=B17-F17

G17 =B17-F17

	A	B	C	D	E	F	G
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)	Seasonal smoothing parameter (delta)		
2	36		0.5	0.5	0.5		
3							
4	**t**	**Demand**	**Level**	**Trend**	**Seasonal Adjustment**	**One-step Forecast**	**Forecast Error**
5	-11				0.988233399		
6	-10				1.039459514		
7	-9				0.932933292		
8	-8				0.912597756		
9	-7				1.043010605		
10	-6				0.906442452		
11	-5				0.920837589		
12	-4				0.926620944		
13	-3				0.988490753		
14	-2				1.016201453		
15	-1				1.048052656		
16	0		144.42	2.2905	1.204004908		
17	1	165				144.9842	20.01578
18	2	171					
19	3	147					
20	4	143					
21	5	164					
22	6	160					

Multiplicative Holt-Winters Exponential Smoothing

- 추정치 업데이트
  - Level 업데이트
  - $\alpha (D_1 / S_{1-p}) + (1-\alpha)(L_0 + T_0)$
  - `C17=$C$2*B17/E5+(1-$C$2)*(C16+D16)`
  - Trend 업데이트
  - $\gamma (L_1 - L_0) + (1-\gamma)(T_0)$
  - `D17=$D$2*(C17-C16)+(1-$D$2)*D16`
  - Seasonality 업데이트
  - $\delta(D_1 / L_1) + (1-\delta) S_{1-p}$
  - `E17=$E$2*(B17/C17)+(1-$E$2)*E5`
  - t=36까지 적용

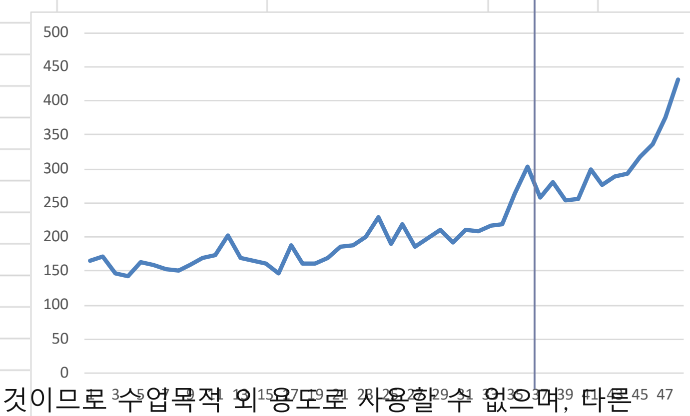
	A	B	C	D	E	F	G
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)	Seasonal smoothing parameter (delta)		
2	36		0.5	0.5	0.5		
3							
4	**t**	**Demand**	**Level**	**Trend**	**Seasonal Adjustment**	**One-step Forecast**	**Forecast Error**
5	-11				0.988233399		
6	-10				1.039459514		
7	-9				0.932933292		
8	-8				0.912597756		
9	-7				1.043010605		
10	-6				0.906442452		
11	-5				0.920837589		
12	-4				0.926620944		
13	-3				0.988490753		
14	-2				1.016201453		
15	-1				1.048052656		
16	0		144.42	2.2905	1.204004908		
17	1	165	156.8375529	7.354026464	1.020138679	144.9842	20.01578
18	2	171	164.3500755	7.433274536	1.039960732	170.6705	0.329501
19	3	147	164.675443	3.87932101	0.912799133	160.2624	-13.2624
20	4	143	162.625153	0.914515505	0.895960263	153.8227	-10.8227
21	5	164	160.3884021	-0.6611177	1.032764215	170.5736	-6.57361
22	6	160	168.120762	3.535621076	0.929069652	144.7836	15.21641
23	7	152	168.3617467	1.888302908	0.911827749	158.0676	-6.06765
24	8	150	166.0642703	-0.20458677	0.914942842	157.7573	-7.75726
25	9	159	163.3554803	-1.456688368	0.980914081	163.9508	-4.95076
26	10	169	164.1021997	-0.354984478	1.023023745	164.5218	4.478212
27	11	173	164.4076287	-0.024777739	1.050157631	171.6157	1.384296

Multiplicative Holt-Winters Exponential Smoothing

- 수요예측
  - $F_{t+1} = (L_t + T_t)S_{t+1-p}$ … $F_{t+p} = (L_t + pT_t)S_t$
  - B53 =(C$52+(A53-A$52)*D$52)*E41
  - B64까지 적용

B53 | =(C$52+(A53-A$52)*D$52)*E41

	A	B	C	D	E	F	G
41	25	189	188.9842382	-0.092431251	1.008687205	195.5039	-6.50393
42	26	218	200.480508	5.701919287	1.057676904	194.1744	23.82559
43	27	185	202.1801043	3.70075783	0.924265498	192.4724	-7.47238
44	28	199	215.3708837	8.4457686	0.904489512	182.2028	16.79718
45	29	210	211.6747926	2.374838717	1.02227287	235.5576	-25.5576
46	30	193	211.7648898	1.232467972	0.916358225	197.21	-4.20999
47	31	211	222.3486037	5.908090935	0.929810619	193.9684	17.03163
48	32	208	227.1048536	5.332170425	0.918210939	210.1206	-2.12064
49	33	216	225.8971604	2.062238637	0.970441083	228.8795	-12.8795
50	34	218	221.5810005	-1.126960639	0.998418885	230.9226	-12.9226
51	35	264	235.7766364	6.534337618	1.085540492	231.7803	32.21966
52	36	304	247.1589259	8.958313566	1.218147058	292.3037	11.69633
53	37	258.3422					
54	38	280.3643					
55	39	253.28					
56	40	255.9635					
57	41	298.4531					
58	42	275.7403					
59	43	288.1177					
60	44	292.749					
61	45	318.0948					
62	46	336.2096					
63	47	375.2718					
64	48	432.0264					



Multiplicative Holt-Winters Exponential Smoothing

- 예측 에러의 확인
- Sum of Squared Error (SSE)
- G1, H1, H4 작성
- H17=G17^2, G52까지 적용
- G2=SUM(H17:H52)

- Standard Error (Standard deviation)
- H2=SQRT(G2/(36-3))

H2 =SQRT(G2/(36-3))

	A	B	C	D	E	F	G	H
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)	Seasonal smoothing parameter (delta)		SSE	Standard Error
2	36		0.5	0.5	0.5		5196.079	12.54819
3								
4	**t**	**Demand**	**Level**	**Trend**	**Seasonal Adjustment**	**One-step Forecast**	**Forecast Error**	**Squared Error**
5	-11				0.988233399			
6	-10				1.039459514			
7	-9				0.932933292			
8	-8				0.912597756			
9	-7				1.043010605			
10	-6				0.906442452			
11	-5				0.920837589			
12	-4				0.926620944			
13	-3				0.988490753			
14	-2				1.016201453			
15	-1				1.048052656			
16	0		144.42	2.2905	1.204004908			
17	1	165	156.8375529	7.354026464	1.020138679	144.9842	20.01578	400.6316
18	2	171	164.3500755	7.433274536	1.039960732	170.6705	0.329501	0.108571
19	3	147	164.675443	3.87932101	0.912799133	160.2624	-13.2624	175.8914
20	4	143	162.625153	0.914515505	0.895960263	153.8227	-10.8227	117.1308
21	5	164	160.3884021	-0.6611177	1.032764215	170.5736	-6.57361	43.21233
22	6	160	168.120762	3.535621076	0.929069652	144.7836	15.21641	231.5391
23	7	152	168.3617467	1.888302908	0.911827749	158.0676	-6.06765	36.81638
24	8	150	166.0642703	-0.20458677	0.914942842	157.7573	-7.75726	60.17511
25	9	159	163.3554803	-1.456688368	0.980914081	163.9508	-4.95076	24.51006
26	10	169	164.1021997	-0.354984478	1.023023745	164.5218	4.478212	20.05439
27	11	173	164.4076287	-0.024777739	1.050157631	171.6157	1.384296	1.916276

Multiplicative Holt-Winters Exponential Smoothing

- 최적의 smoothing parameter 찾기
- 목적함수

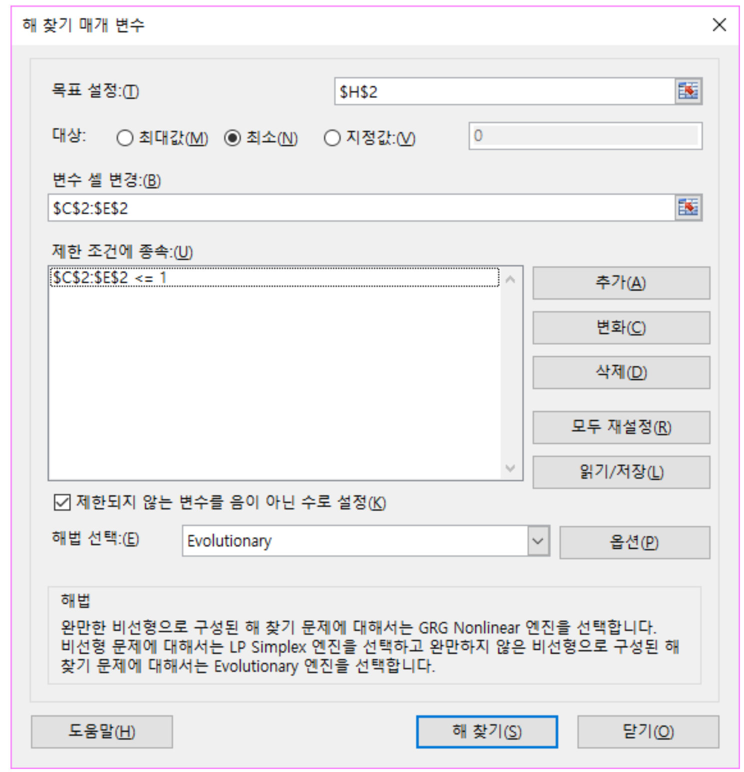
  Min H2=SQRT(G2/(36-3))

- 의사결정변수

  C2, D2, E2

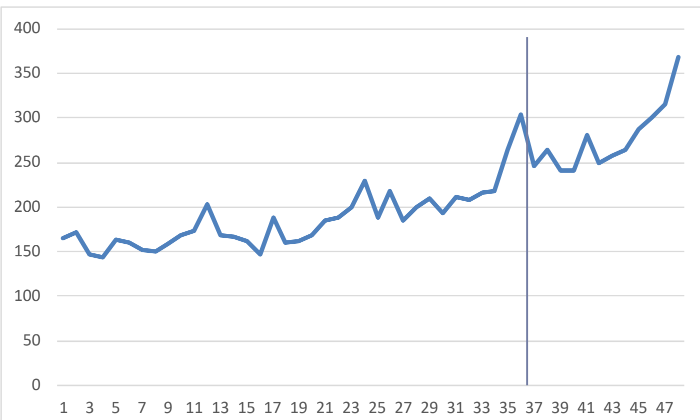
- 제약식

  0≤ C2, D2, E2 ≤1



Multiplicative Holt-Winters Exponential Smoothing

- 최적해 확인
  - Standard Error가 12.54에서 10.38로 감소
  - 최적의 smoothing parameter : 0.31, 0.23, 0



C2 | f_x 0.308084790928483

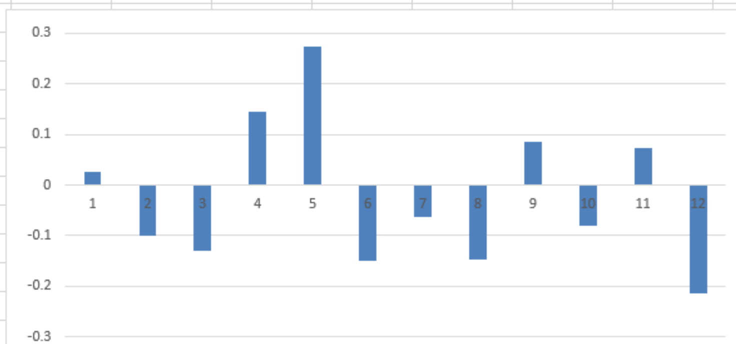
	A	B	C	D	E	F	G	H
1	Total Months		Level smoothing parameter (alpha)	Trend smoothing parameter (gamma)	Seasonal smoothing parameter (delta)		SSE	Standard Error
2	36		0.31	0.23	0		3555.97	10.38059
3								
4	**t**	**Demand**	**Level**	**Trend**	**Seasonal Adjustment**	**One-step Forecast**	**Forecast Error**	**Squared Error**
5	-11				0.988233399			
6	-10				1.039459514			
7	-9				0.932933292			
8	-8				0.912597756			
9	-7				1.043010605			
10	-6				0.906442452			
11	-5				0.920837589			
12	-4				0.926620944			
13	-3				0.988490753			
14	-2				1.016201453			
15	-1				1.048052656			
16	0		144.42	2.2905	1.204004908			
17	1	165	152.950482	3.731287539	0.988233399	144.9842	20.01578	400.6316
18	2	171	159.0930882	4.28805164	1.039459514	162.8644	8.135644	66.1887
19	3	147	161.5900569	3.874497563	0.932933292	152.4237	-5.4237	29.41657
20	4	143	162.7629551	3.250708788	0.912597756	151.0026	-8.00258	64.0413
21	5	164	163.309749	2.626385377	1.043010605	173.154	-9.15401	83.79594
22	6	160	169.1950877	3.378865003	0.906442452	150.4116	9.588444	91.93825

Autocorrelation의 확인

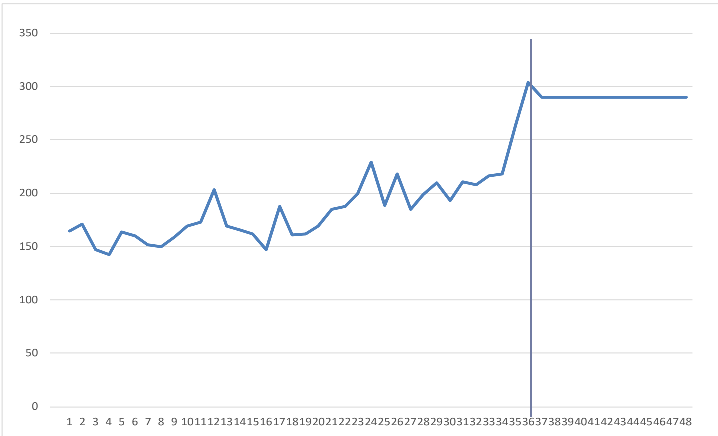
- Error term에 대한 확인이 필요 (HW Autocorrelation)
  - Holt's Autocorrelation 복사
  - HoltWintersSeasonal에서 Forecast Error (값) 복사

D40 =D38/$C38

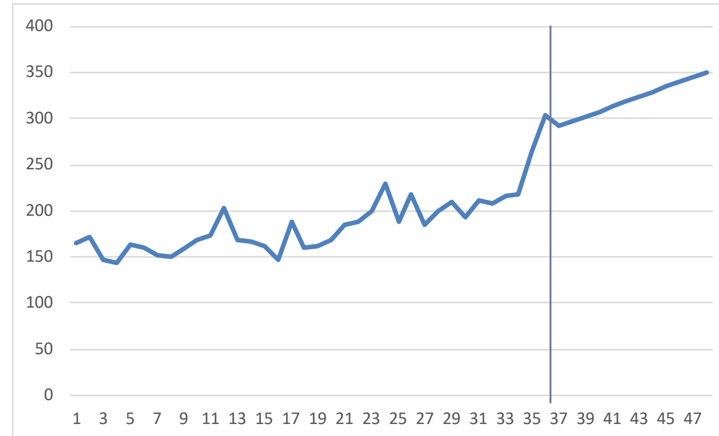
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	**t**	**Forecast Error**	**Deviations from mean**	1	2	3	4	5	6	7	8	9	10	11	12
40			Autocorrelation	0.026303	-0.10096	-0.13066	0.144774	0.273613	-0.14891	-0.06382	-0.14756	0.085241	-0.08055	0.073501	-0.21504
41			Upper cutoff	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333	0.333333
42			Lower cutoff	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333	-0.33333



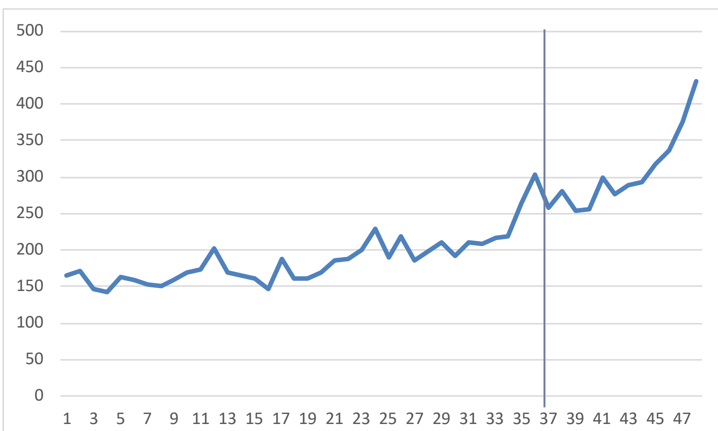
결과 비교



Simple Exponential Smoothing



Holt's Method



Winter's Method

	SSE	Standard Error
Simple Exponential Smoothing	14555	20.39
Holt's Method	14097	20.36
Winter's Method	3555	10.38

(파이썬) SES 단순평활예측법

- from statsmodels.tsa.holtwinters import SimpleExpSmoothing

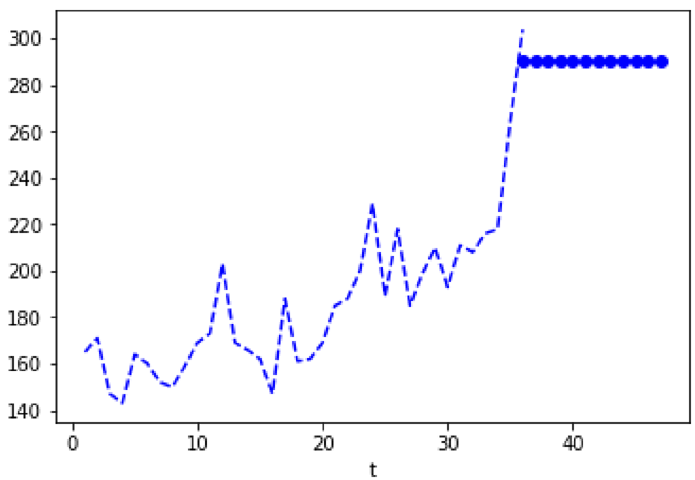
- SES=SimpleExpSmoothing(DATA).fit(smoothing_level=0.50)

  —> smoothing_level = alpha값
  —> fit()만 사용하면 자동으로 최적의 alpha값을 찾아줌

- SES.함수

  함수 :
  .forecast(기간) : 기간만큼의 예측값 ex) SES.forecast(12)
  .sse() : sum of square error
  .params : initial_level, slope나 smoothing_level 등 각종 parameters
  .forecast(12).plot(style='--', marker='o', color='blue')

(파이썬) SES 단순평활예측법



- 순서

① SimpleExpSmoothing 모듈을 사용하여 모델링하기

② forecast 함수로 12개월 예측하기

③ sse로 standard error 판단하기
sse = 20.94

④ plot 으로 그리기

⑤ forecast로 예측값 구하기

(파이썬) Holt 이중평활예측법

- from statsmodels.tsa.holtwinters import Holt
- from statsmodels.tsa.holtwinters import ExponentialSmoothing
- from scipy.stats import linregress

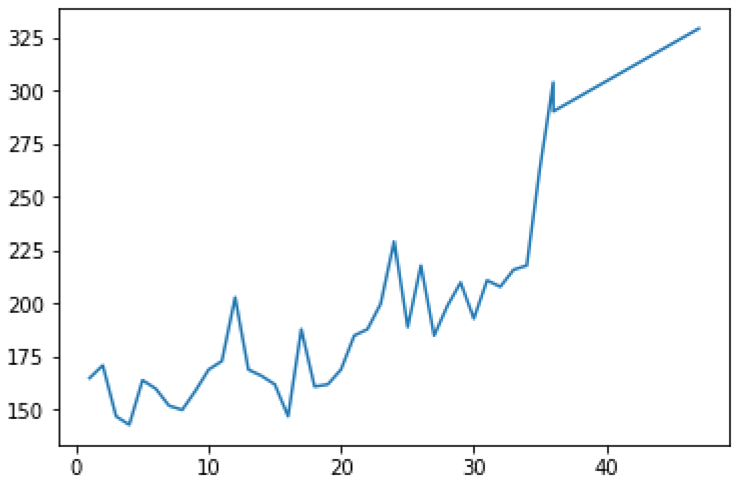
- linregress(x=x축,y=Data)

  —> slope (추세) 가 있는지 확인
  —> p-value 로 slope 가 통계적으로 유의미 한지 확인 ($p<0.05$)

- Hz = ExponentialSmoothing(Data,trend='add').fit()
  = Holt(Data).fit()

  —> initial_level 과 initial_slope 는 내부적으로 계산되므로 선택조절 X
  —> trend 은 대부분 'add' 로 설정
  —> $Y_t^{(adj)} = Y_t - \widehat{T}_t$

(파이썬) Holt 이중평활예측법



- 순서

① 데이터 절반의 initial_level과 initial_slope 확인하기

② 초기값을 토대로 level과 trend 구하기 (155.88, 0.8369)

③ level과 trend를 기준으로 12개월 예측하기

④ ExponentialSmoothing로 최적화하기 (0.66, 0.0)

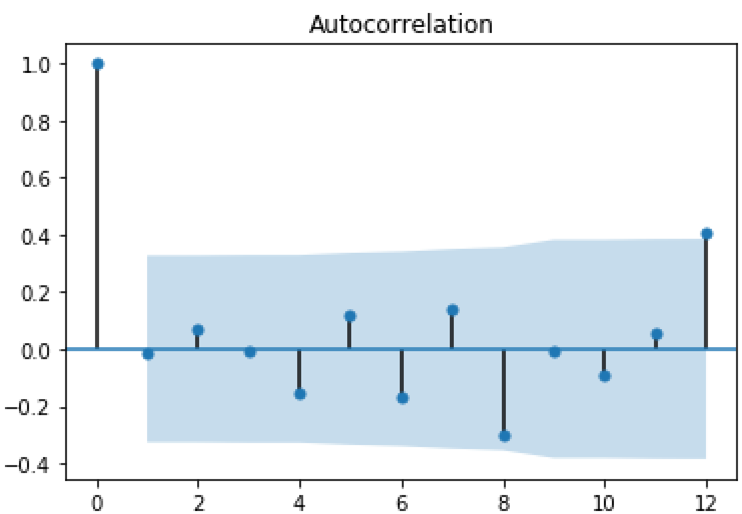
⑤ .resid 를 사용하여 'Deviations from mean' 구하기

(파이썬) Holt 이중평활예측법

- from statsmodels.tsa.stattools import acf
- from statsmodels.graphics.tsaplots import plot_acf

⑥ Deviations from mean에 대하여 plot_acf 를 하면 자기상관그래프가 그려짐
(lags=기간)

⑦ acf 를 사용하여 자기상관계수 확인가능
(nlags=기간)




```
array([ 1.        , -0.01736202,  0.06753719, -0.00440473, -0.15176781,
        0.11991841, -0.16848871,  0.13962329, -0.30455657, -0.00583059,
       -0.09231966,  0.05242069,  0.40895063])
```

(파이썬) Holt-winters 삼중평활예측법

- from statsmodels.tsa.holtwinters import ExponentialSmoothing

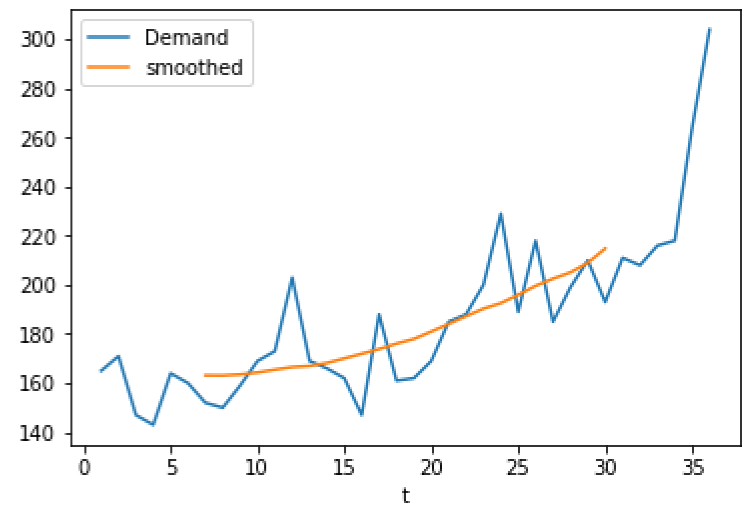
- season = ExponentialSmoothing(Data,trend='add‘,seasonal=‘mul’,seasonal_periods=12).fit()

  mul= multiplicative
  add= additive

- 순서

  ① MA(moving Average) smoothed 구하기

  (1) numpy.mean
  (2) DataFrame.plot()



(파이썬) Holt-winters 삼중평활예측법

- ② 'initial seasonal factors X 3' 만들기
  ③ 'Deseasonalized Data' 만들기
  —> Data를 Initial Seasonal Factors X3 으로 나누기
  ④Level과 Trend 와 Seasonal_adj 값으로 forecast 해보기

  ⑤ Seas =
  ExponentialSmoothing(Data,trend='add',seasonal='mul',seasonal_periods=12).fit()

  ⑥ 그래프 그리기
  Data.plot()
  Seas.forecast(12).plot()

  style='--' —> 선모양
  marker='o' —> 값모양
  color='blue' —> 색