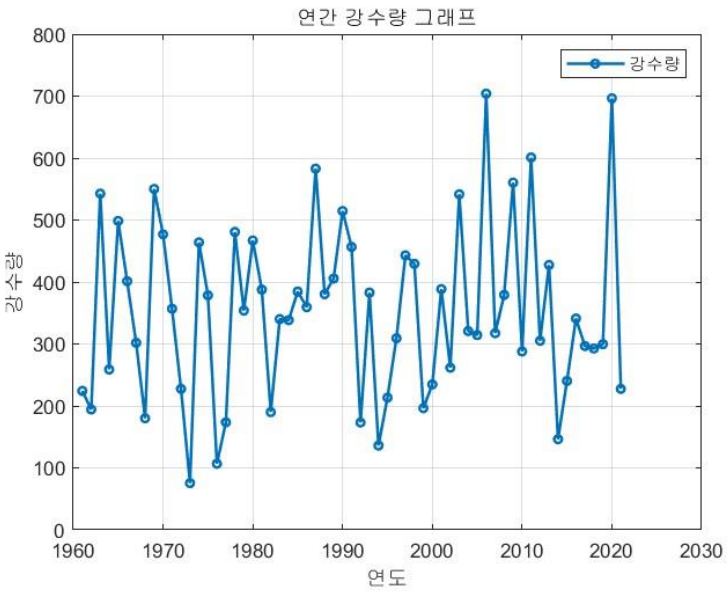
데이터 마이닝 중간고사 대체 과제

컴퓨터 공학부 202201479 박지원

장마는 우리나라 6월 하순부터 7월 하순까지 계속해서 많이 내리는 비로 기상학적으로는 장마전선의 영향을 받아 비가 오는 경우를 의미한다. 이러한 장마는 우리 삶에 영향을 미친다. 대표적으로 장마로 인해 땅꺼짐이 발생할 수 있고, 산사태가 발생할 수 있다. 뿐만 아니라 농작물에 영향을 미치기도 한다. 기사에 따르면, 2023년 역대급 장마로 인해 작물의 생육이 크게 감소하게 되고 생리장애에 취약한 상태가 되었다고 한다. 이로 인해 경도와 당도의 감소, 뿌리 및 생육저하, 석회결핍과 무름 및 썩음 증상, 일조량 감소에 따른 광합성 저하, 병해충 발생 증가, 낙과, 낙화, 열과, 일소 피해 등의 피해가 발생할 수 있다고 한다. 특히나 장마철이 지나고 복숭아가 다 떨어져 복숭아 농장이 피해를 봤다는 기사도 심심치 않게 접할 수 있다. 만일, 장마철에 비가 얼마나 올지 정확히 예측할 수 있다면, 그 해에는 농작물과 인명 피해를 미리 예방하고 준비할 수 있지 않을까 하는 생각이 들었다. 마침 여름도 다가오고 있어 이번 데이터 마이닝 중간고사 대체 과제로 장마철 강수량을 시계열을 통한 예측을 하면 어떨까 싶어 주제로 선정하게 되었다.

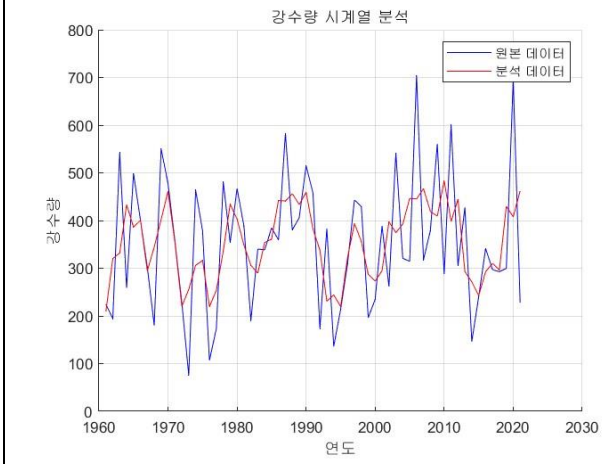
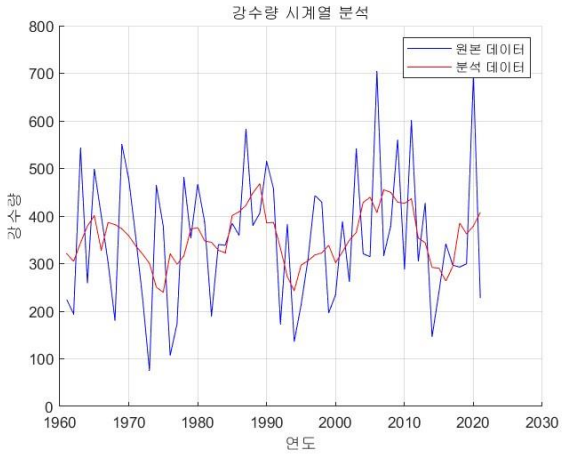
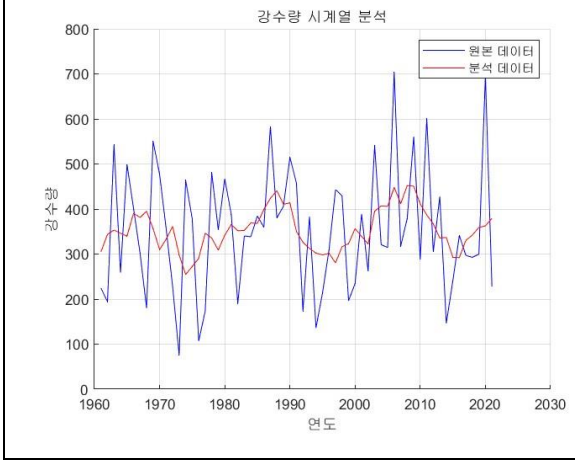
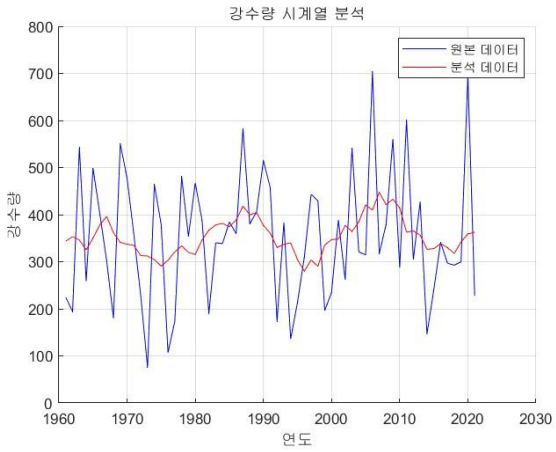
	Y	R	강수일수
2	1961	224.2	12.2
3	1962	194.3	14.4
4	1963	542.6	22.8
5	1964	258.8	14
6	1965	498.5	20.4
7	1966	401.3	19.4
8	1967	301.9	15.9
9	1968	180.1	10.6
10	1969	550.2	28.6
11	1970	476.8	21.8
12	1971	356.8	19.1
13	1972	227.5	9.6
14	1973	75.4	4.5
15	1974	464	25
16	1975	378.5	18.2
17	1976	106.7	14.1



데이터 셋은 위의 사진과 같이 우선 csv의 형태이다. 이 파일은 첫번째 열에는 Y 즉 연도를 라벨링 해 뒀고, 두번째 열에는 R 즉 그 해 장마철 평균 강수량을 라벨링 해 뒀다. 세번째 열에는 강수일수가 라벨링 되어 있지만, 이번 예측에선 쓰이지 않을 데이터 열이 되겠다. 연도는 1961년부터 2022년까지로 이루어져 있다. 우측 그래프는 매트랩으로 그려 본 연도별 장마철 강수량이며, 데이터를 한 눈에 볼 수 있어 이미지를 첨부하였다.

장마철 강수량 데이터 출처_[기상청 기상자료개방포털]
https://data.kma.go.kr/climate/rainySeason/selectRainySeasonList.do
장마와 태풍, 많은 비에 작물 생육 저하 및 생리장애 우려_[영농자재신문]
http://www.newsfm.kr/mobile/article.html?no=8184

우선, 데이터의 잡음과 불규칙한 변동을 줄여서 데이터의 일반적인 경향을 더 명확하게 파악하기 위해 스무딩을 했다. 스무딩을 할 땐, moving average 기법을 사용했다. 이는 window size를 적당히 정해야 한다는 번거로움이 있지만, 단순하고 효과적인 스무딩 방법이라 활용하기 편한 듯 보였다. 그리고 이를 통해 주기적인 패턴이나 트렌드를 파악하는데 유리하다고 생각되기도 했다. 게다가 분석 과정은 매트랩을 활용하였기에, 매트랩에는 moving average가 내장되어 있어 분석 시간을 조금이라도 더 단축할 수 있는 듯했다.

Window size = 3	Window size = 5
	
Window size = 7	**Window size = 9**
	

위의 그림들은 차례대로 window size가 3 , 5, 7 , 9 일 때의 사진이다. 확실히 window size의 크기가 커질 수 록, 데이터는 더욱 평균값에 가깝게 스무딩이 된다. 하지만 데이터들이 너무 평준화되어버리면 예측 시 어려움이 발생할 수 있다. 3은 너무 스무딩이 안된 상태로 보이고, 9는 너무 평준화된 것으로 보여 5가 적당하다고 판단되었다. 따라서 스무딩은 window size = 5로 하여금 분석과 예측을 진행하게 될 것이다.

이어서 예측을 위해 ARIMA모델을 사용하겠다. 이 모델을 사용하는 이유는 시계열에 특화되어 있기 때문에, 시계열 데이터의 패턴을 파악하고 미래 값을 예측하는데 효율적이기 때문이다. ARIMA모델은 AR, I, MA의 세가지 구성 요소로 이루어져 있으며, 이 요소들은 각각 p,d,q로 표현한다.

<자기회귀 항>

$$y_t = \alpha_0 + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \ldots + \alpha_p y_{t-p} + \epsilon_t$$

<차분 항>

$$z_t = \nabla^d y_t = (1-B)^d y_t$$

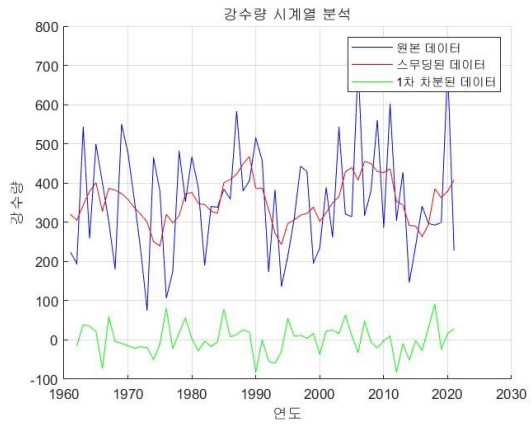
<이동평균 항>

$$\epsilon_t = \theta_0 + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \theta_q \epsilon_{t-q}$$

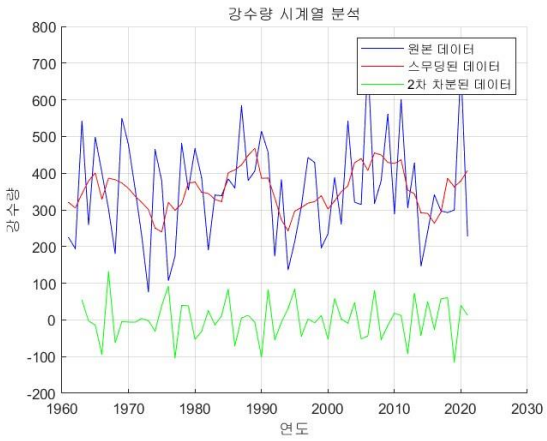
각각의 항들을 구하는 방법은 위와 같다.

우선, 차분 항을 구하도록 하겠다. 차분은 1차 혹은 2차 정도를 사용한다. 차분의 차수가 증가할수 록 모델이 복잡해지고 데이터의 손실이 증가할 가능성이 있으므로, 1차와 2차 중에 차분을 고르기로 하였다. 매트랩에는 diff라는 내장함수가 존재하기에, 실습 시 위의 복잡한 공식을 사용하지 않아 시간을 단축시킬 수 있었다.

<1차 차분>



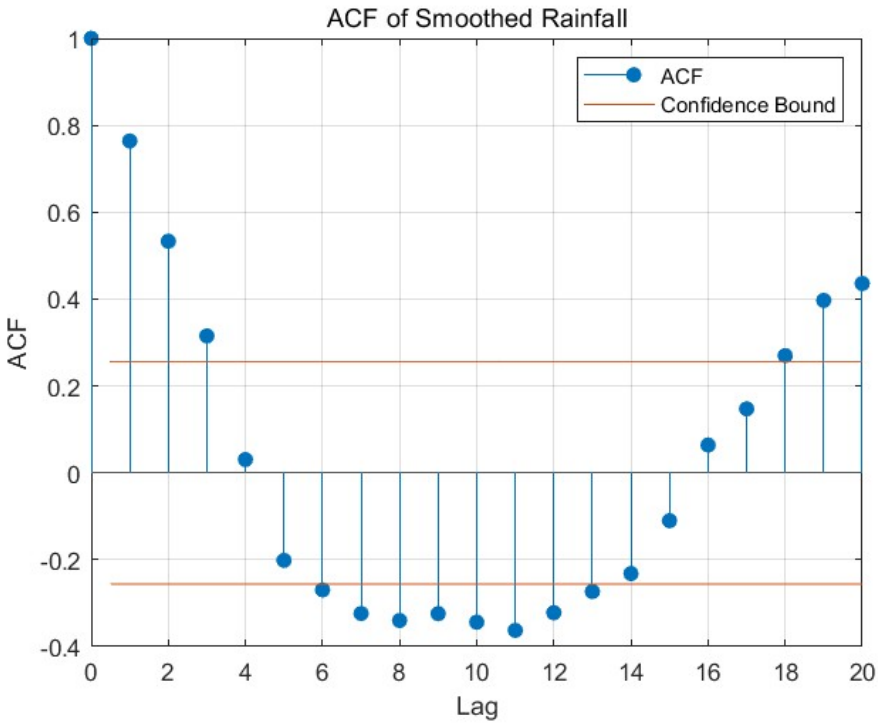
<2차 차분>



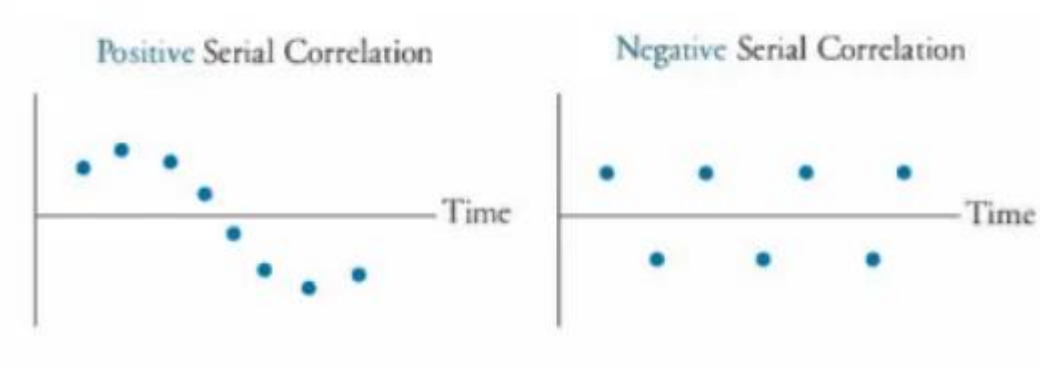
위의 그림과 같이 좌측이 1차 차분, 우측이 2차 차분의 결과이다. 차분의 경우 시계열 데이터의 추세와 계절성을 제거하여 데이터의 평균을 0에 가깝게 함으로써 정상성을 달성하고 예측 모델의 성능을 향상시키는 데에 목적으로 한다. 따라서 두 그래프 중, 데이터의 평균이 0에 더 가까운 모델은 우측 2차 차분의 그래프라고 생각되어 d = 2라고 정했다. 사실 이론으로만 수업을 들었을

때, 실제로 1차와 2차 차분의 차이가 클지 감이 안오기도 했지만, 직접 눈으로 확인하니 그 차이를 실감할 수 있었다.

다음으로 확인할 것은 ACF 와 PACF이다. ACF란 시계열에서 중요한 개념으로, 시계열 데이터의 관측치들 사이의 자기상관을 측정하는 것이다. 이는 시간 차이를 두고 데이터가 서로 얼마나 관련성이 있는지를 나타내며, 데이터의 패턴과 계절성을 이해하고 예측 모델을 구성하는 데 사용된다. 매트랩에선 마찬가지로 autocorr라는 내장 함수가 존재하여 어렵지 않게 ACF를 구할 수 있었다.



위의 그래프는 ACF의 결과이다. 위의 그래프는 positive한 ACF그래프의 형태이다.

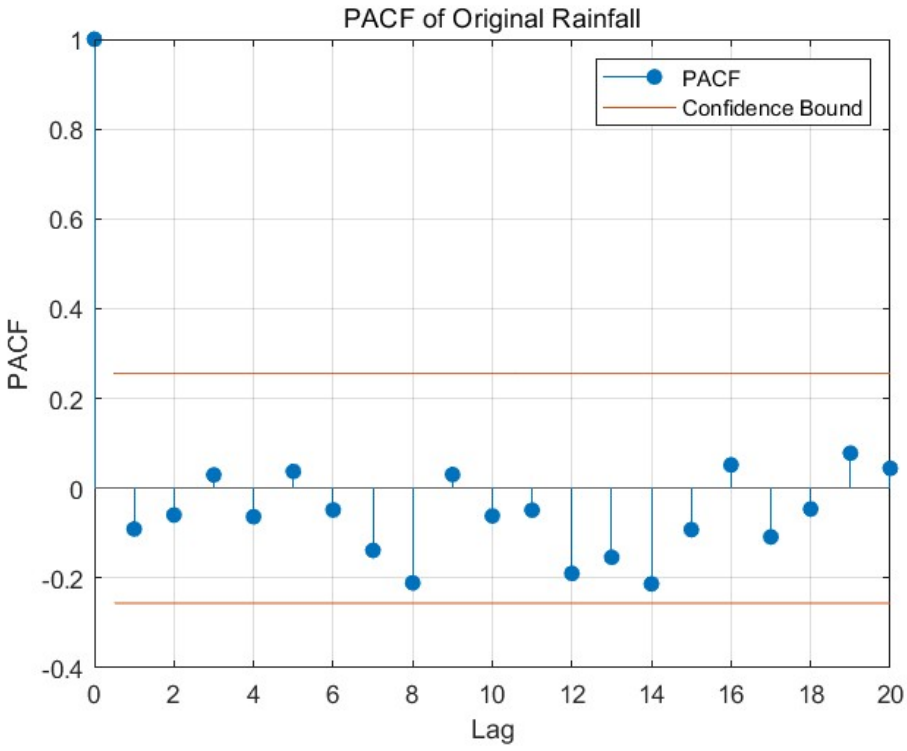


[Positive 와 Negative Serial Correlation 그래프 사진 출처]
https://medium.com/@boramlee1/%EC%9E%90%EA%B8%B0-%EC%83%81%EA%B4%80-%EA%B7%B8%EB%9E%98%ED%94%84-acf-auto-correlation-function-41142935a87d

위의 ACF는 positive serial correlation과 negative serial correlation의 혼합을 보인다. 구체적으로 분석해보자면, Lag 0에서 1값을 가지는 부분은 자기 자신과의 상관관계를 나타낸다. 이러한 형태는 시계열 분석에서 표준이다. 하지만 Lag 10에서 -0.4에 가까운 ACF 값이 되는 부분은 negative serial correlation의 모습이다. 즉 시계열 데이터의 어떤 관측치가 특정 기준보다 높다면 10시간 후의 관측치는 기준보다 낮을 가능성이 높다는 것을 의미한다. 이후 다시 Lag 16부터 양수의 값을 가지는데, 이는 positive serial correlation이 있음을 나타낸다. 따라서 이 시계열 데이터는 시간이 지남에 따라 positive serial correlation 과 negative serial correlation이 교차하는 복잡한 패턴을 보인다. 이러한 패턴을 통해 매년 6~7월에 대한 데이터는 계절성을 띈다고 볼 수 있다. 이러한 계절성은 제거하지 않기로 하였다. 왜냐하면 장마철 강수량 패턴을 이해하는 것이 목적이기 때문이다. 장마철은 일반적으로 특정 기간에 집중적으로 강수량이 발생하는 계절적 패턴을 가진 시기이기에, 이와 같은 계절성을 그대로 유지해야 장마철의 강수량 패턴을 더 잘 이해할 수 있을 것이라 생각했다.

위의 그래프를 통해 처음으로 0이하의 값이 나오는 5를 q값으로 하기로 했다.

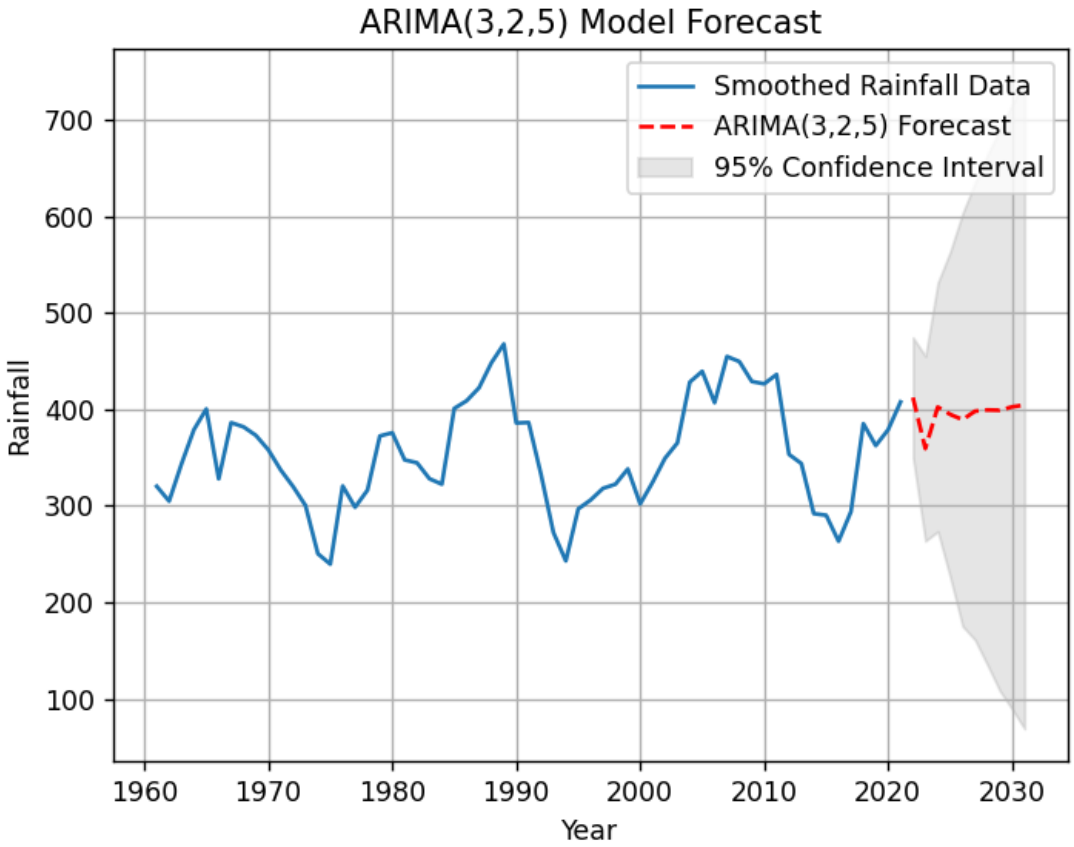
이어서 확인할 것은 PACF이다. 이는 시계열 데이터의 관측치 간의 관계를 이해하기 위해 사용된다. 즉 부분적인 자기 상관을 보여준다는 것이다. 이를 통해 ARIMA의 AR값을 구하기로 한다. 마찬가지로 매트랩에는 parcorr라는 PACF에 관한 내장 메소드가 존재하기에 복잡한 공식을 직접 쓰지 않아 실습 시 시간을 단축시킬 수 있었다.



보통 AR 모델의 차수는 PACF 그래프에서 처음으로 0에 근접하거나 유의 수준 내로 들어가는 시

차로 정한다고 한다. 위의 PACF 그래프를 보면 모두 유의 수준 내에 존재하므로 0과 처음으로 근접한 3을 AR의 차수 즉, p값으로 하였다.

따라서 차분을 통해 얻은 d값은 2, ACF를 통해 얻은 MA값은 5, PACF를 통해 얻은 AR 값은 3이므로, ARIMA(3,2,5)를 통해 데이터를 예측하기로 했다. 매트랩에서는 해당 코드 파일에서 직접 스무딩을 하고, 그 데이터를 이용하려고 했으나 자꾸만 배열이 맞지 않다는 오류가 발생해 따로 스무딩 파일을 저장하였다. 하지만 여전히 배열이 맞지 않다는 에러가 발생해 데이터를 예측하는데 있어 매트랩에서 파이썬으로 툴을 옮겨 실습을 진행하였다. 매트랩은 일반적으로 행렬 중심의 언어이기 때문에, 행렬에 매우 민감한 언어이다. 반면에 파이썬은 데이터 프레임을 사용하는 경우가 많아 매트랩에서 발생하는 오류를 해결할 수 있을 거라고 생각되었다. 파이썬에서도 역시 원본 데이터가 아닌 따로 저장한 스무딩 데이터를 불러왔다. 파이썬에선 pandas와 matplotlib 그리고 statsmodels 정도를 install하면 간단하게 ARIMA 내장 함수를 통해 모델 설정 및 추정이 가능하다.



위의 그래프는 최종적으로 나타낸 ARIMA 예측 모델이다. 그래프를 해석해보자면 2022년 까지의 데이터는 기존 데이터의 그래프를 보여주며, 2022년 이후 10년간의 빨간 그래프 선은 이제까지의 분석을 통해 알아낸 변수 값들을 이용한 ARIMA 예측이 적용된 선이다. 신뢰도 구간을 표시한 이유는, 모델 예측의 불확실성을 시각적으로 표현하여 결과를 더 신뢰하기 위해서이다. 2022년 이후

의 예측 데이터는 회색 부분과 같이 95% 이내의 신뢰도 구간 내에서 예측 값이 나타나게 된다. 즉, 예측 값은 신뢰성이 있다고 보인다. 예측 값을 마저 해석해보면, 2022년 이후 강수량 감소 추세가 보이지만 이후 대략 감소한 만큼 다시 강수량이 증가할 것으로 보인다. 다시 강수량이 증가한 이후에는 완만하게 2030년까지 강수량이 점차 증가할 것이라고 해석할 수 있다. 전체적인 그래프를 확인해 보면 2022년 이후 2030년 까지의 그래프는 이전에 가장 강수량이 많던 날보단 낮고 가장 강수량이 적던 날보단 높기에 너무 폭우가 예상된다 거나 너무 가뭄이 예상되진 않는다고 보인다. 따라서 장마철에 대한 예비를 극성으로 할 필요까진 없어 보인다.

이렇게 직접 ARIMA모델을 분석하고 예측까지 실습해보았다. 이론으로만 배웠을 땐, 직접적인 감이 잡히지 않았다. ARIMA 모델을 이용해 예측하기 위해 p,d,q라는 변수가 필요하구나 정도로 머리로만 받아들였지, 직접 실습을 해보니 각 변수들의 의미도 제대로 알게 되었다. 무엇보다 결과가 그래프로 눈에 보이게 출력이 되니 직접 만들었다는 뿌듯함까지 들었다. 맨 처음 데이터를 정하게 된 이유와 같이 실제 예측 값을 보며 장마에 대해 어떻게 대비를 해야 할 지 미리 떠올릴 수 있어 이번 과제의 목적을 이룬 것으로 보인다. 물론, 실습을 진행하며 매트랩으론 데이터를 다루기 어려웠던 난관도 있었지만, 다른 언어를 사용하여 극복하기도 했다. 이를 통해 언어의 종류가 왜 다양한지, 그리고 어느 분야에서 각각의 언어가 그 특징을 더 잘 활용할 수 있는지 몸소 경험할 수 있었다. 보고서를 쓰기 전부터 어떤 식으로 써야 할 지 감이 잡히지 않아 "시계열 분석을 활용한 서울시 미세먼지 예측*"이라는 국내 논문을 참고하였다. 물론 논문의 내용을 이해하는 건 쉽지 않았지만 시계열 분석과 예측에 대한 보고서를 어떤 식으로 작성해야 할지 틀을 잡는데 도움이 되었다. 실제로 실습을 진행하고 나니 데이터 마이닝이라는 과목에 대해 더욱 흥미를 느끼게 되었다.