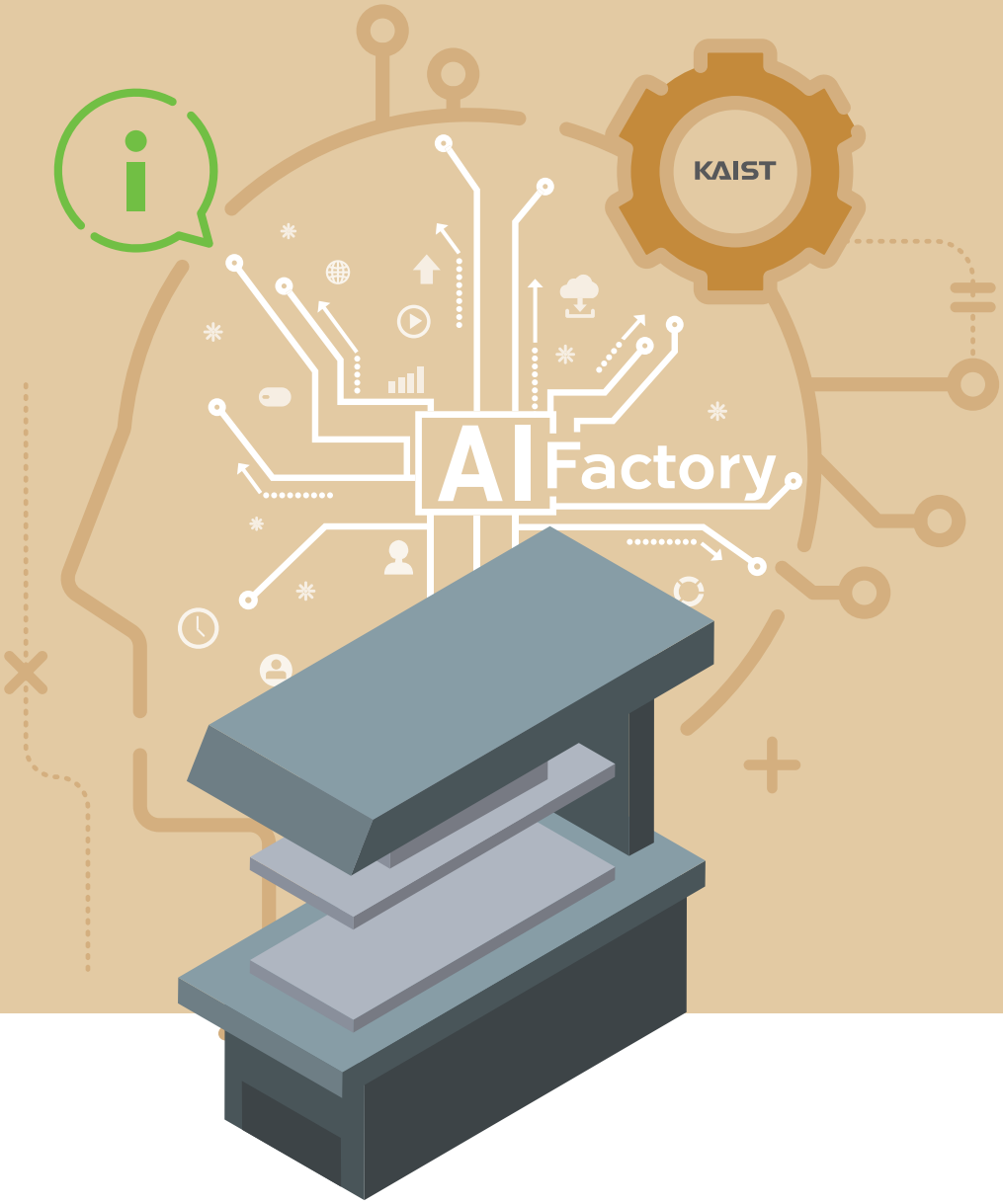
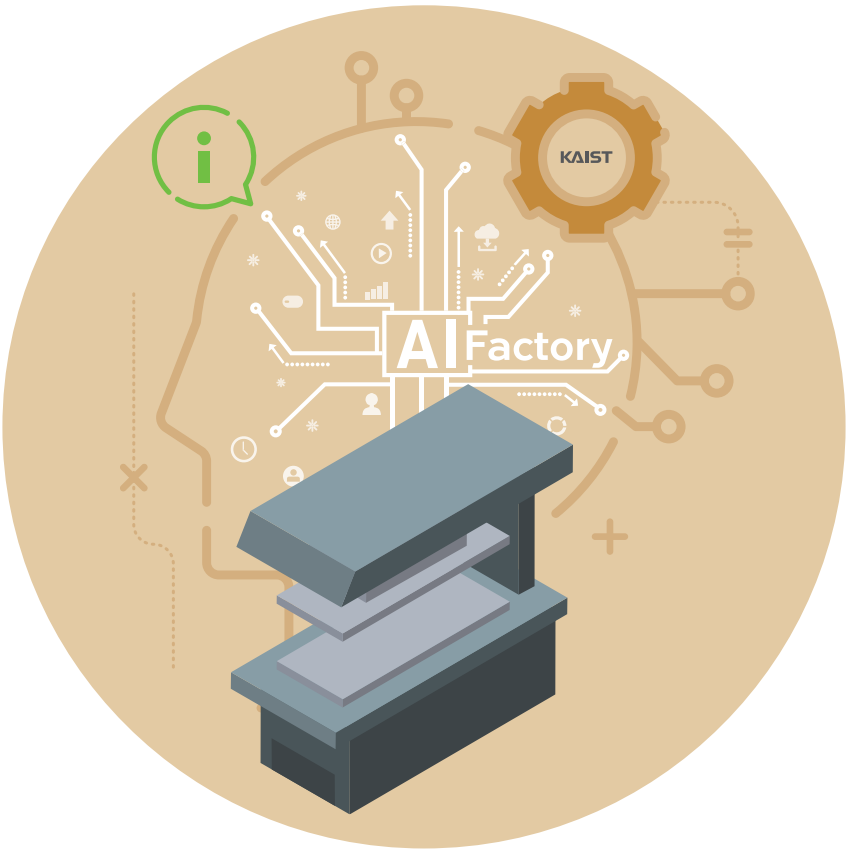
Guide book

「프레스 AI 데이터셋」 분석실습 가이드북



중소벤처기업부
KOSMO 스마트제조혁신추진단
KAIST

「프레스 AI 데이터셋」
분석실습 가이드북



중소벤처기업부

KOSMO
스마트제조혁신추진단

KAIST

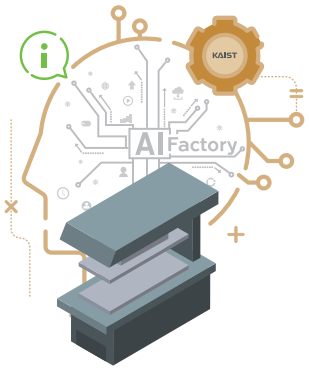
Contents

1 분석요약 04

2 분석 실습

1. 분석 개요 05
   - 1.1 분석 배경 05
     - 1) 공정(설비) 개요
     - 2) 이슈사항(Pain point)
   - 1.2 분석 목표 06
     - 1) 분석목표 설정
     - 2) 제조 데이터 정의 및 소개
     - 3) 제조 데이터 분석 기대효과
     - 4) 시사점(implication)
2. 분석실습 08
   - 2.1 제조데이터 소개 08
     - 1) 데이터 수집 방법
     - 2) 데이터 유형/구조
     - 3) 데이터 (품질) 전처리
   - 2.2 분석 모델 소개 14
     - 1) AI 분석모델
   - 2.3 분석 체험 17
     - 1) 필요 SW, 패키지 설치 방법 및 절차 가이드
     - 2) 분석 단계별 프로세스
       - [단계 ①] 라이브러리/데이터 불러오기
       - [단계 ②] 데이터 종류 및 개수 확인
       - [단계 ③] 데이터 정제 (전처리)
       - [단계 ④] 데이터 특성 파악
       - [단계 ⑤] 데이터 정규화
       - [단계 ⑥] 가우시안 혼합 모델 구축
       - [단계 ⑦] 결과 분석 및 해석
3. 유사 타 현장의 「프레스 AI 데이터셋」 분석 적용 29

부 록 분석환경 구축을 위한 설치 가이드 30



「프레스 AI 데이터셋」 분석실습 가이드북

- **필 요 SW :** Python, Anaconda - Jupyter Notebook
- **필요 패키지 :** Pandas, matplotlib, scikit-learn
- **분 석 환 경 :** [운영체제] Ubuntu 14.0 이상, [CPU] Intel Xeon 2.3 GHz, [RAM] 13GB, [GPU] Tesla K80
- **필요 데이터 :** Press_RawDataSet.xlsx, Press_error.xlsx (1차 가공 데이터), input_data.csv (2차 가공 데이터)
- **주 관 기 관 :** 한국과학기술원(KAIST)
- **수 행 기 관 :** 울산과학기술원(UNIST), 주식회사 이피엠솔루션즈

1 분석요약

No	구분		내용
1	**분석 목적** (현장 이슈, 목적)		- 본 AI 데이터 셋과 AI 모델을 통해 실제 프레스 공정에서 발생하는 품질 문제를 해결하기위해 주요 변수간의 상관관계 분석을 통해 다양한 기계학습 방법과 알고리즘을 활용하여 프레스 공정 데이터에 대한 불량 예측 모델을 만든다.
2	**데이터셋 형태 및 수집방법**		- **분석에 사용된 변수명 :** Press time(가압 시간), Pressure(압력) - **수집 방법 :** 생산 Lot 단위 불량 이력 데이터 - **확장자 :** csv
3	**데이터 개수 데이터셋 총량**		- **데이터 개수 :** Row 수(125,526개) - **데이터셋 총량 :** 2.41 MB
4	**분석적용 알고리즘**	**알고리즘**	혼합 가우시안 모델(Gaussian Mixture Model)
		알고리즘 간략소개	- 혼합 가우시안 모델은 가우시안 분포를 여러 개 활용하여 군집화 (Clustering)을 진행하는 방식의 모델이다. 이는 전형적인 비지도 학 습 모델 중 하나이며, 각 군집들이 정규 분포를 이룰 것이라는 가정 을 가지고 k개의 군집으로 나누어주는 알고리즘이다. 군집 개수인 k 는 직접 설정해주어야 하며 k 값에 따라 결과가 크게 달라질 수 있 다. 현 프레스 데이터 셋에는 정상 샘플, 그리고 3가지 타입의 비정 상 샘플로 4개의 집단으로 군집화를 진행하였으며 비정상 샘플 집 단에 속해 있는 데이터들을 불량품으로 판별 한다.
5	**분석결과 및 시사점**		- 프레스 공정에서의 품질문제 해결을 위해 불량품의 원재료 및 금형정보를 수집하고, 분석하여 불량 예측 AI모델을 개발하였다. - 제조기업의 프레스 공정조건 최적화 및 품질예측을 통해 판금 생산 공정 현장에 널리 적용 될 것으로 판단된다.

2 분석 실습

1. 분석 개요

1.1 분석 배경

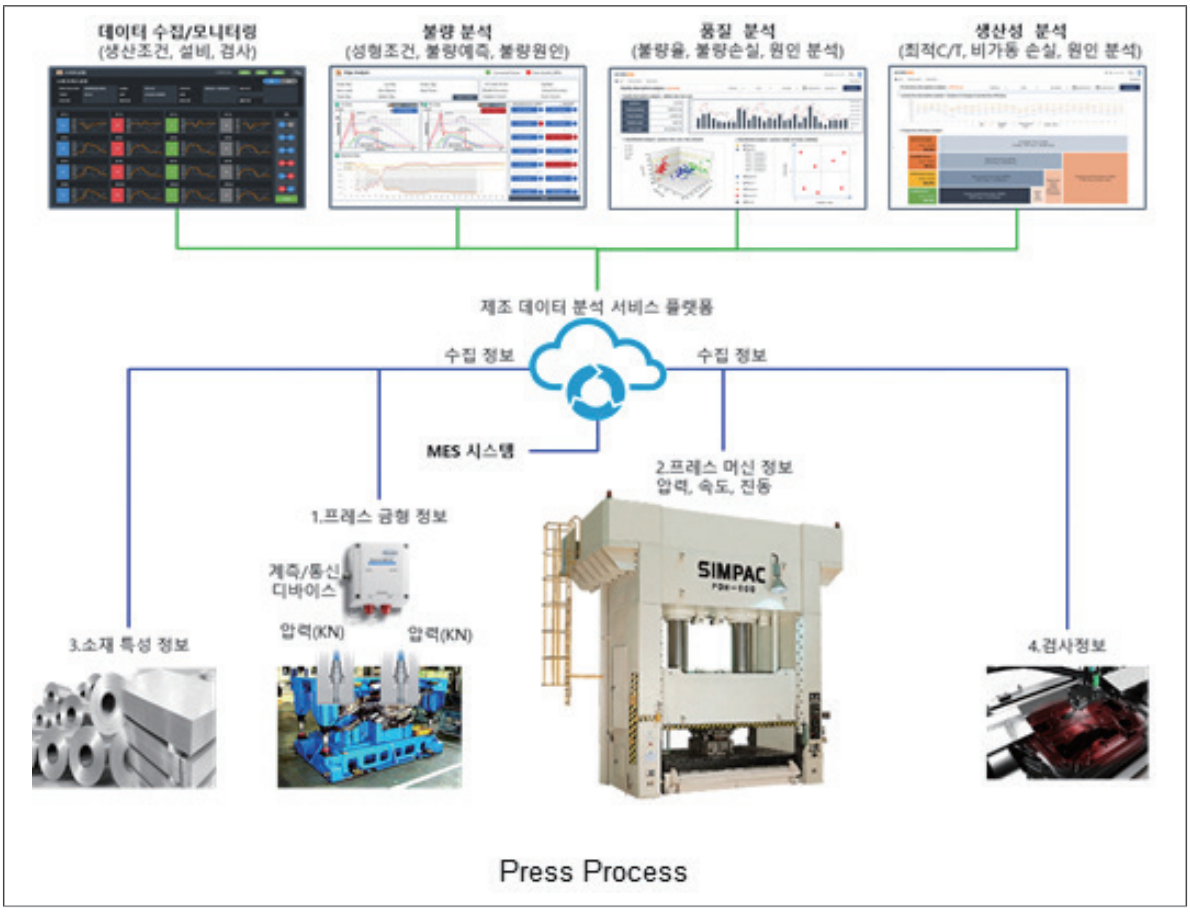
1) 공정(설비) 정의 및 특징

• **프레스 가공 개요**

- 프레스 가공은 제품/부품의 소재, 프레스기계, 금형 등 3가지로 구성된 가공방법이며, 기술이 발전되더라도 쉽게 소멸되지 않는 특수한 생산기술의 한 분야이다.
- 프레스는 전단가공, 굽힘가공, 드로잉가공, 성형가공, 압축가공 등 5가지로 분류하고 있다.
- **전단가공**은 전단기나 금형을 사용하여 재료에 파단강도 이상의 압력을 가하여 잘라내는 가공법이다.
- **굽힘가공**은 가공판재의 중립면을 기준으로 인장과 압축이 동시에 작용하는 가공법으로 재료에 힘을 가하여 굽힘응력을 발생시켜 여러 가지 모양의 제품을 만드는 가공법이다.
- **드로인가공법**은 금속판이나 소성이 큰 재료를 다이 속으로 끌고 들어가면서 이음매가 없는 컵, 그릇모양의 용기를 주름이나 균열이 발생하지 않게 성형하는 가공법이다.
- **성형가공법**은 재료를 금형의 상하형 사이에 넣고 압력을 가해 원하는 형상으로 만드는 가공법으로 재료변형이 작은 그룹에 한정한다.
- **압축가공법**은 재료에 강한 압축력을 가하여 소재변형을 일으키면서 금형내부의 형상대로 제품이 성형되도록 하는 가공법이다.

분석 실습 데이터는 성형가공 공정 및 데이터를 대상으로 한다.

• 프레스 공정 과정



[그림 1] 프레스 공정 및 관리 시스템 구성도

2) 이슈사항(Pain point)

• 공정의 문제 상황

- 프레스 공정 후 주름, 터짐 및 버(Burr) 발생 등의 불량이 발생하고 있으며 이로 인한 폐기비용이 발생하고 있다. 공정 불량의 90%가 금형 관리 미흡과 공정 조건을 작업자의 경험적 판단에서 운영하는 것에 기인하는 것으로 파악하고 있으며, 이를 인공지능(Artificial Intelligence, AI)을 활용한 분석을 통해 개선하고자하며 우선 문제 정의 및 문제에 따른 영향 후보 인자들을 정의하고 데이터 수집하여 빅데이터 분석 방법론에 의해 해결하고자 한다.

• 문제 해결 장애 요인

- 프레스 공정이 로봇 기반 자동화 공정으로 구성되어 있으나 PLC 데이터는 HMI 판넬에 디스플레이되지만 데이터 수집이 되지 않고 있어 데이터 수집을 위해 별도 센서를 추가하여야 한다. 검사 또한 작업자 육안으로 처리되고 있으며 데이터를 시스템에 등록되지 않아 이를 해결해야 한다.

• **극복 방안**

- 프레스 설비의 압력값 데이터 수집을 위해 별도 스트레인 센서를 프레스 구조물에 설치하여 프레스가 기동할 때마다 압력값 데이터를 구할 수 있도록 하였으며, 검사 결과는 인원 투입으로 육안 확인 후 기록하도록 만들어 양품, 불량품 및 불량유형별 라벨링을 해서 데이터를 만들었다.

1.2 분석 목표

1) 분석목표 설정

- 프레스 공정에 투입되는 품목번호, 치수 등과 설비 공정 조건 데이터 압력값 및 프레스 속도 등을 압력센서로부터 수집함과 더불어 검사 결과를 양품, 불량품, 불량 유형별 분류하여 데이터 수집하여 머신에 학습시켜 불량 원인 분석과 이에 따른 공정 조건 분석을 AI 모델을 활용하여 예측과 최적화 분석을 통해 운영함으로써 품질 및 생산성을 향상하고자 하는 목적이 있다.

2) 제조 데이터 정의 및 소개

공정 변수 조건		내 용
독립변수	속도 관련	프레스가 가해질 때의 속도
	압력 관련	금형의 좌,우측의 균형에 따라 행해지는 압력값 (pressure 1,2) 및 압력 값의 합 (Pressure 5)
종속변수	불량 여부	제품 각각에 대한 불량 선별값 (표시 없음 / Unlabeled)

3) 제조 데이터 분석 기대효과

- 원재료에 따른 프레스 공정 조건 최적화 및 품질 예측 모델은 프레스 판금 생산 공정의 생산현장에 널리 사용될 수 있을 것으로 판단한다.

4) 시사점(implication)

- 요약기술 프레스 공정의 품질 문제에 영향을 줄 것으로 판단되는 설비 공정 변수와 원재료 및 금형정보를 정의하고 수집하여 데이터를 바탕으로 AI 분석 모델을 통해 품질 이슈와 설비 조건 변수와의 상관관계 분석을 통해 설비/공정 조건을 최적화하고, 불량예측 모델을 개발 적용함으로써 중소기업에서 작업자의 경험적 의존에서 벗어나 AI활용으로 실질적인 품질개선에 기여할 수 있다는 점에서 시사하는 바가 크다고 할 수 있다.

2. 분석실습

2.1 제조데이터 소개

1) 데이터 수집 방법

- **제조 분야** : 자동차 부품 제조
- **제조 공정명** : 자동차 부품 제조 공장 중 프레스 연속 공정 설비
- **수집 장비** : 센서에서 수집된 MES 데이터, 불량선별(인력투입)
- **수집 기간** : 2020/05/04 ~ 2020/05/29
- **수집 주기** : 사이클 타임 4초, Proessing time은 0.5초

2) 데이터 유형/구조

- **데이터셋 구조, 컬럼 수, 데이터 개수, AI 데이터셋 주요 변수 정의**

1) 1차 가공 데이터 : Press_RawDataSet.xlsx, Press_error.xlsx

prod_dt

	idx	Machine_Name	Item No	working time	Press time(ms)	Pressure 1	Pressure 2	Pressure 5
0	1	Press-01	ED5260	2020-05-04	550.0	275.0	274.0	549.0
1	2	Press-01	ED5260	2020-05-04	550.0	275.0	274.0	549.0
2	3	Press-01	ED5260	2020-05-04	550.0	275.0	275.0	550.0
3	4	Press-01	ED5260	2020-05-04	550.0	275.0	275.0	550.0
4	5	Press-01	ED5260	2020-05-04	549.2	274.6	276.0	550.6
...	...	...	...	...	...	...	...	...
61107	4121	Press-01	ED5260	2020-05-29	550.0	275.0	267.0	542.0
61108	4122	Press-01	ED5260	2020-05-29	550.0	275.0	267.0	542.0
61109	4124	Press-01	ED5260	2020-05-29	549.8	274.9	269.0	543.9
61110	4125	Press-01	ED5260	2020-05-29	550.6	275.3	267.0	542.3
61111	4126	Press-01	ED5260	2020-05-29	550.6	275.3	267.0	542.3

61112 rows × 8 columns

[그림 2] 비식별화 원본 데이터 예시

- 1차 가공 데이터는 제품의 설비 공정 조건에 대한 4가지 환경 조건이 작업 시간 순서에 따라서 나열된 xlsx(엑셀파일)의 형태로 추출할 수 있다.

	A	B	C	D	E	F
1	idx	Machine_Name	Item No	working time	defect	defect type
2	1	Press-01	ED5260	2020-05-04 0:00	0	1
3	2	Press-01	ED5260	2020-05-04 0:00	1	2
4	3	Press-01	ED5260	2020-05-04 0:00	0	3
5	4	Press-02	ED5260	2020-05-05 0:00	1	1
6	5	Press-03	ED5260	2020-05-05 0:00	1	2
7	6	Press-04	ED5260	2020-05-05 0:00	0	3
8	7	Press-05	ED5260	2020-05-06 0:00	1	1

[그림 3] 비식별화 원본 데이터 중 추계 일별 불량률 예

- 또한, 추가적인 검수자의 인력을 통한, 일별로 수렴한 제품들의 (각 물품 당 불량여부가 아닌) 하루 총 불량 개수 및 불량 유형에 대하여 정리된 형태로 파일을 정리할 수 있다.

- 데이터 속성정의 표

공정 변수	항목 설명	판정범위	수집범위	단 위
idx	생산순번	-	-	-
Machine_Name	생산설비	-	-	-
Item No	생산품목	-	-	-
working time	작업시간	-	-	-
Press time	프레스 가압 시간	2000 - 50	5000 - 0	0.01초 (ms)
Pressure 1	프레스 압력 1	200-350	100-700	압력 (bar)
Pressure 2	프레스 압력 2	200-350	100-700	압력 (bar)
Pressure 5	프레스 압력 5 (압력 합계)	400-700	200-1400	압력 (bar)
defect	불량 수	-	-	개수 (수)
defect type	불량유형	-	-	1. 파임불량, 2. 용접부족, 3. 크랙발생

2) 2차 가공 데이터 : input_data.csv

idx	Press time(ms)	Pressure 1	Pressure 2	Pressure 5	PassOrFail
1	550	275	274	549	
2	550	275	274	549	
3	550	275	275	550	
4	550	275	275	550	
5	549	275	276	551	
6	549	275	276	551	
7	550	275	274	549	
8	550	275	274	549	
9	551	275	276	551	
10	551	275	276	551	

[그림 4] 2차 가공 데이터 예시

- 1차 가공 데이터에서, 모델링에 필요한 정보가 부재한 클래스를 지워낸 데이터를 정리하였다. 이렇게 정리한 데이터셋과, 1차 가공 데이터 중 '일별 추계 불량 수'를 활용하여 AI 모델을 학습 시켜, 불량 예측을 한다.

- 데이터 속성정의 표

공정 변수	항목 설명	단 위
idx	생산순번	-
Machine_Name	생산설비	-
Item No	생산품목	-
working time	작업시간	-
0 (Press time*)	프레스 가압 시간	0.01초 (ms)
1 (Pressure 1*)	프레스 압력 1	압력 (bar)
2 (Pressure 2*)	프레스 압력 2	압력 (bar)
3 (Pressure 5*)	프레스 압력 5 (압력 합계)	압력 (bar)

2차가공 데이터셋인 input_data.csv에는 0,1,2,3으로 각각 처리되어 있다. 해당 부분은 1차 가공 데이터에서 모델에 기입하기 위하여 속성의 이름을 숫자로 변환한 것으로, 1차 가공 데이터의 속성과 순서 및 이름이 같다.

- 기술통계

구 분	개 수	평 균	표준편차	최소값	중간값	최대값	최빈값
0 (Press time)	64350	550.263	22.327	25.5	550	3550.2	550
1 (Pressure 1)	64348	275.062	1.24	174.5	275	286.9	275
2 (Pressure 2)	64343	269.777	3.156	167	269	277	267
3 (Pressure 5)	64355	544.841	5.105	144.6	543.9	941.4	542

• **독립변수/ 종속변수 정의**

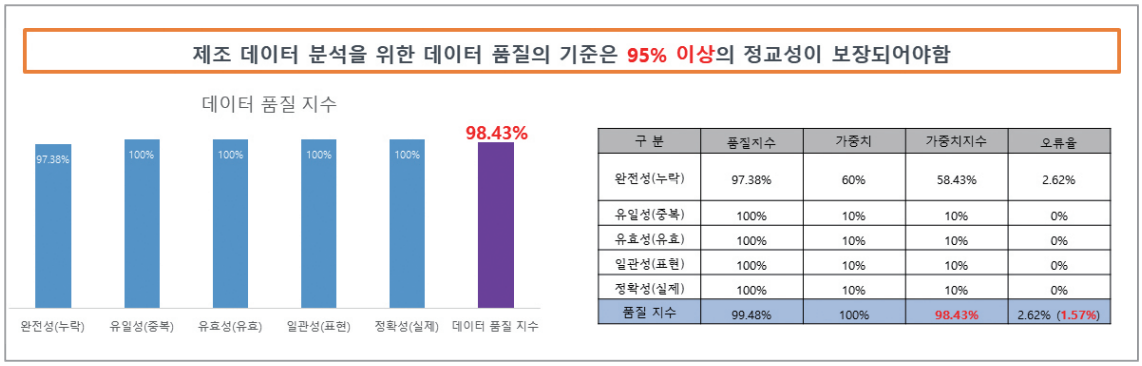
- 독립변수란, 다른 변수에 영향을 받지 않는 변수로, 입력값이나 원인을 나타낸다.
- 종속변수란, 독립변수의 변화에 따라 어떻게 변화하는지를 알고 싶어하는 변수로, 결과물이나 효과를 나타낸다.

공정 변수 조건	내 용
독립변수	가압 시간
	프레스 가압력 1
	프레스 가압력 2
	프레스 가압력 5 (가압력1, 2의 합)
종속변수	(표시 없음)

3) 데이터 (품질) 전처리

• **데이터 품질 전처리 목적**

- 실제 공정에서 발생하는 데이터들은 값이 의미 없거나 누락 및 오타가 발생하여 데이터 품질이 떨어진다.
- 아무리 좋은 분석 방법론을 가지고 있더라도 품질이 낮은 데이터를 이용하면 좋은 결과를 얻을 수 없기 때문에, 데이터 품질 전처리는 데이터 분석에서 가장 중요한 단계이다.
- 따라서 데이터들의 5가지 품질 지수를 파악하고, 데이터 전처 리를 통해 품질 지수를 향상시킨다.



구 분	품질지수	가중치	가중치지수	오류율
완전성(누락)	97.38%	60%	58.43%	2.62%
유일성(중복)	100%	10%	10%	0%
유효성(유효)	100%	10%	10%	0%
일관성(표현)	100%	10%	10%	0%
정확성(실제)	100%	10%	10%	0%
품질 지수	99.48%	100%	98.43%	2.62% (1.57%)

[그림 5] 데이터 품질 지수

• **데이터 품질 지수**

- **완전성(Completeness)** : 필수항목에 누락이 없어야 한다.
- **유일성(Uniqueness)** : 데이터 항목은 유일해야 하며 중복되어 서는 안된다.
- **유효성(Validity)** : 데이터 항목은 정해진 데이터 유효범위 및 도메인을 충족해야 한다.
- **일관성(Consistency)** : 데이터가 지켜야 할 구조, 값, 표현되는 형태가 일관되게 정의되고, 서로 일치해야 한다.
- **정확성(Accuracy)** : 실제 존재하는 객체의 표현 값이 정확히 반영되어야 한다.

<예시>

idx	Machine_Name	Item N	working time	Press time(ms	Pressure	Pressure	Pressure
1843	Press-01	ED5260	2020-05-07 0:0		275.2	273.2	548.4
1844	Press-01	ED5260	2020-05-07 0:0		275.5	273.1	548.6
1885	Press-01	ED5260	2020-05-07 0:0		274.8	276	550.8
1886	Press-01	ED5260	2020-05-07 0:0		275.6	273.2	548.8
1619	Press-01	ED5260	2020-05-15 0:0		274.8	267	541.8
561	Press-01	ED5260	2020-05-20 0:0		274.9	269	543.9
755	Press-01	ED5260	2020-05-26 0:0		274.9	269	543.9
3879	Press-01	ED5260	2020-05-28 0:0		275.2	267	542.2
744	Press-01	ED5260	1900-01-00 0:00		274.9	269	543.9

-> 컬럼에 null 값이 들어가 있으므로 완전성을 위배한다.

idx	Machine_Name	Item N	working time	Press time(ms	Pressure	Pressure	Pressure
2912	Press-01	ED5260	1900-01-00 0:00	551	275.5	269	1644.5
105	Press-01	ED5260	8159-02-28 0:00	549	274.5	269	543.5
107	Press-01	ED5260	8277-03-19 0:00	550.2	275.1	267	542.1
108	Press-01	ED5260	8336-03-29 0:00	550.2	275.1	267	542.1
111	Press-01	ED5260	8513-04-27 0:00	550.2	275.1	267	542.1
136	Press-01	ED5260	9988-12-25 0:00	549.8	274.9	267	541.9
2	Press-01	ED5260	1900-01-00 0:00	550	275	267	542
138	Press-01	ED5260	1900-01-00 0:00	549.2	274.6	267	541.6
140	Press-01	ED5260	1900-01-00 0:00	550.2	275.1	267	542.1
142	Press-01	ED5260	1900-01-00 0:00	551	275.5	267	542.5

-> working time에 유효하지 않은 날짜형식이 들어있으므로 유효성을 위배한다.

[그림 6] 데이터 유효성 검사

- **데이터 품질 지수 :** 세부 설명

◦ 완전성 품질 지수 = ((1-결측치)/전체 데이터수) * 100

① Null 값이 30%이상인 데이터들은 데이터의 완전성이 떨어지기 때문에 컬럼별 Null값의 비율을 확인하여 삭제한다.

② 데이터의 결측치를 확인하기 위해 isnull()함수를 사용한 뒤 sum()함수를 이용하여 총 결측치 개수를 구한다.

③ 구한 결측치의 개수를 이용하여 완전성 품질 지수를 구한다.

◦ 유일성 품질지수 = ((1-중복데이터수)/전체 데이터수) * 100

① 이 데이터는 유일한 값을 가지는 컬럼이 존재하지 않으므로 유일성을 판단하지 않는다.

② 유일성을 판단하고 싶다면 idx와 workingtime 두 컬럼을 이용하여 합성키를 만든 뒤 기본키로 설정하여 중복을 판단하면 된다.

◦ 유효성 품질지수

① 데이터가 유효범위내에 들어가 있는가.

② 데이터가 형식에 맞는가.

③ 수집된 날짜 안에 들어가 있는가. 등을 검증하는 것이다.

◦ 일관성 품질지수 = (일관성만족 데이터 수/전체데이터 수)*100

여기 용접 데이터는 다른 데이터를 참조하는 컬럼이 없으므로 일관성을 판단하지 않는다.

◦ 정확성 품질지수 = (1-(정확성 위배데이터수/전체데이터수))*100

여기 용접데이터는 상관관계가 있는 데이터가 존재하지 않기 때문에 정확성을 판단하지 않는다.

- 프레스 데이터 품질 지수 확인 결과는 다음과 같다.

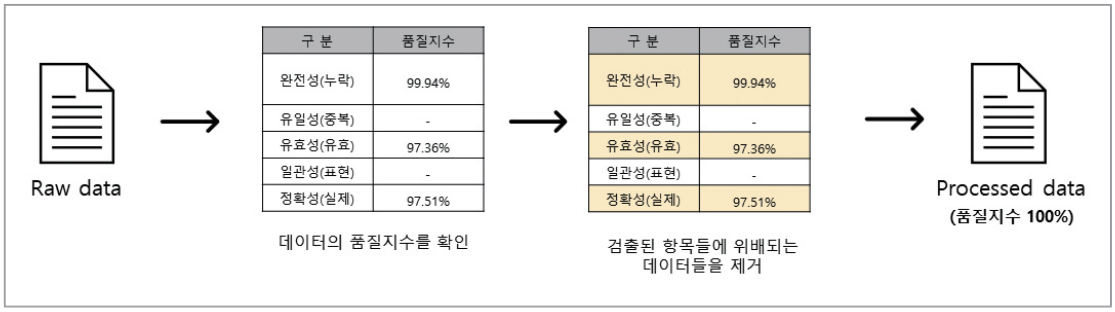
구 분	품질지수	가중치	가중치지수	오류율
완전성(누락)	99.94%	70%	69.99%	0.06%
유일성(중복)	-	-	-	-
유효성(유효)	97.36%	15%	14.60%	2.64%
일관성(표현)	-	-	-	-
정확성(실제)	97.51%	15%	14.63%	2.49%
품질 지수	98.27%	100%	**99.22%**	**0.78%**

[표 1] 프레스 데이터 품질 지수

- 데이터 전처리 방법

◦ 개발 언어 : Python (파이썬)

◦ 개발환경 : Jupyter notebook (주피터랩)



[그림 7] 데이터 전처리 과정 도식화

- 데이터 전처리 결과물

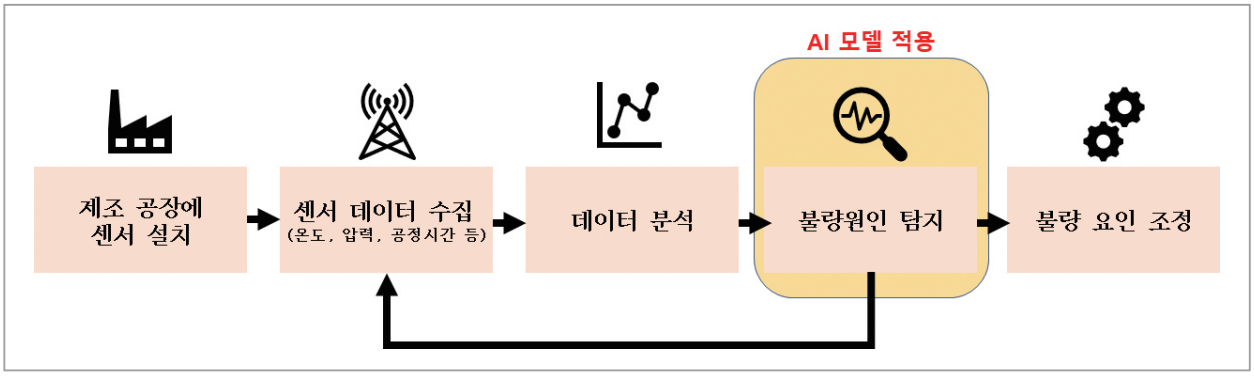
prod_dt

	idx	Machine_Name	Item No	working time	Press time(ms)	Pressure 1	Pressure 2	Pressure 5
0	1	Press-01	ED5260	2020-05-04	550.0	275.0	274.0	549.0
1	2	Press-01	ED5260	2020-05-04	550.0	275.0	274.0	549.0
2	3	Press-01	ED5260	2020-05-04	550.0	275.0	275.0	550.0
3	4	Press-01	ED5260	2020-05-04	550.0	275.0	275.0	550.0
4	5	Press-01	ED5260	2020-05-04	549.2	274.6	276.0	550.6
...	...	...	...	...	...	...	...	...
61107	4121	Press-01	ED5260	2020-05-29	550.0	275.0	267.0	542.0
61108	4122	Press-01	ED5260	2020-05-29	550.0	275.0	267.0	542.0
61109	4124	Press-01	ED5260	2020-05-29	549.8	274.9	269.0	543.9
61110	4125	Press-01	ED5260	2020-05-29	550.6	275.3	267.0	542.3
61111	4126	Press-01	ED5260	2020-05-29	550.6	275.3	267.0	542.3

61112 rows × 8 columns

◦ 총 61,112개로 약 5%(3,247개)가 outlier로 검출하여 삭제하였다.

2.2 분석 모델 소개



[그림 8] 제조 공정에서의 데이터 흐름 및 AI능 모델 적용 단계 도식화

1) AI 분석모델

- 해당 AI 방법론(알고리즘) 선정 이유 기술

◦ **사용 알고리즘 :** 가우시안 혼합 모델(Gaussian Mixture Model)

해당 프레스 데이터의 경우 클래스 변수가 없기 때문에 비지도 학습 알고리즘을 적용할 수 있다. 현재 데이터에서 알 수 있는 정보는 일자(working time)별 제품 생산 정보와 불량 타입(defect type) 및 각 불량 타입별 개수(defect)이다. 정상과 불량 타입

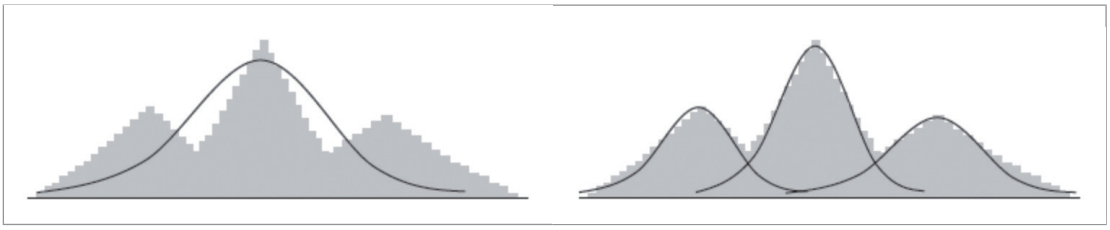
3가지로 구성되어 있어 데이터가 4개의 군집으로 구성되어 있다고 가정하고, 클래스 변수의 정보 없이도 군집화 할 수 있는 알고리즘을 선정하고자 하였다. 가우시안 혼합 모델을 통해 대부분의 데이터를 설명하는 군집은 정상 군집으로, 나머지 소량의 데이터가 포함된 나머지 3개의 군집을 불량 군집으로 분류할 수 있다.

- 적용하고자 하는 AI 분석 방법론(알고리즘의 구체적 소개

◦ 가우시안 혼합 모델이란?

가우시안 혼합 모델(GMM)은 비지도 학습 알고리즘 중 하나이다. 가우시안 혼합 모델은 분포 기반 군집분석에서 가장 대표적이라고 할 수 있다. 여기서 군집 분석이란 주어진 데이터들을 특성에 따라 유사한 것 끼리 묶음으로써 각 유형별 특징을 분석하는 기법을 뜻하며, 분포 기반 군집분석에서는 각 군집은 '어떠한 확률 분포에 따라 형성된다.'라는 가정을 하고 있다.

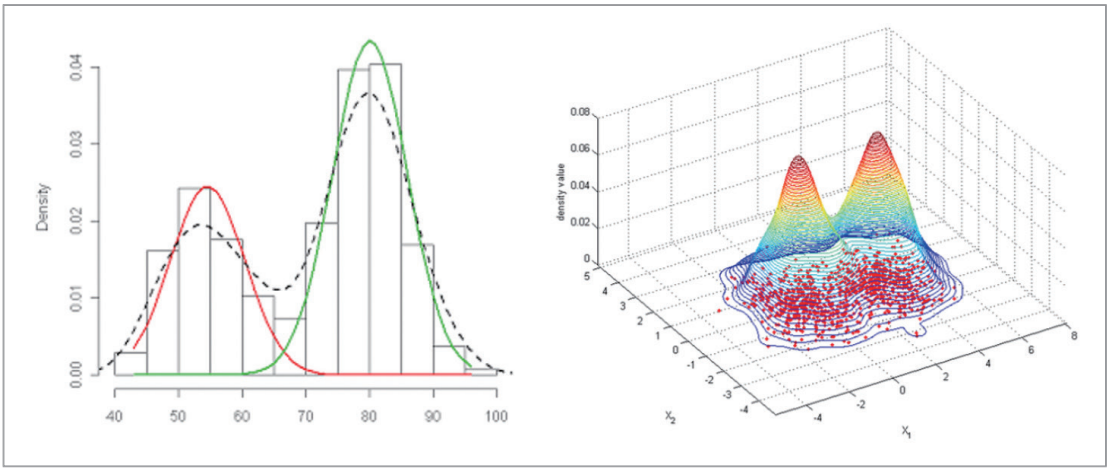
가우시안 혼합 모델(GMM)은 전체 데이터의 확률 분포가 여러 개의 가우시안 분포(정규 분포)의 조합으로 이뤄져 있다고 가정하고 각 분포에 속할 확률이 높은 데이터 끼리 군집을 묶는 방법이다. 모델화하기에는 데이터들의 평균을 중심으로 하나의 그룹으로 뭉쳐있는 단봉형의(unimodal) 형태만 표현이 가능한 가우시안 확률 분포의 한계점([그림 9],[그림 10] 참고)을 완화시키기 위해 고안된 것이다.



[그림 9] 단봉형의 가우시안 확률 분포

[그림 10] 3개의 가우시안 확률 분포로 이루어진 혼합모델

예를 들어, 분석할 데이터의 분포가 아래 [그림 11]과 같다면 이 데이터는 두 개의 정규 분포가 결합된 형태라고 생각 할 수 있다. 그러면 각 데이터가 정규 분포 상으로 볼 때 둘 중 어떤 분포에 속할 확률이 더 높은지를 계산하여 군집을 나눌 수 있다.



[그림 11] 혼합 가우시안 모델

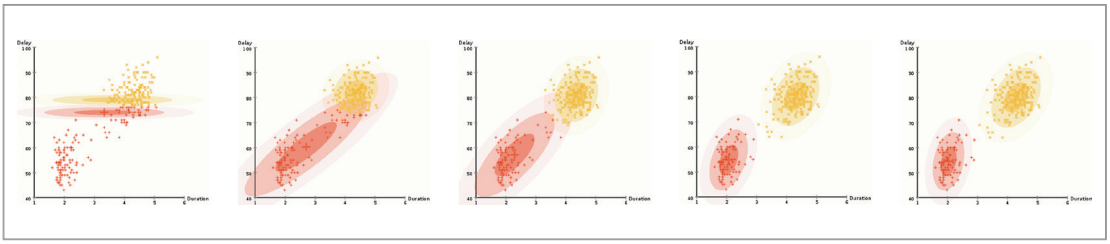
가우시안 혼합 모델에서 모수는 각 분포의 평균과 분산, 각 분포(군집)가 선택될 사전 확률로 크게 3가지가 있다. 이 모수 추정을 위해서 가우시안 혼합 모델은 Expectation and Maximization(EM) 알고리즘을 적용한다. EM알고리즘을 4단계로 정리할 수 있다.

1) **Initialization** :필요한 모수에 대해 초기값을 선정한다.

2) **E(Expectation) Step** : x가 특정 군집에 속할 확률을 계산한다.

3) **M(Maximization) Step** : 계산된 확률을 통해 모수 재추정한다.

4) **Stop** : 수렴 조건이 만족될 때까지 E step과 M step을 반복한다.



[그림 12] EM 알고리즘 과정

EM 알고리즘은 모수 값을 통해 사후 확률을 추정하고 이를 통해 다시 파라미터를 추정하며 로그가능도(log-likelihood)가 최대화할 수 있게 반복하는 기법이다.

가우시안 혼합 모델은 확률 분포의 차이를 고려하여 군집을 묶는 방식이기 때문에 k 평균 알고리즘에 비해 좀 더 통계적으로 엄밀한 결과를 얻을 수 있다는 장점이 있다. 반면에, 계산 양이 많기 때문에 대량의 데이터에 사용하기 어렵고, 유형들의 분포가 정규분포와 차이가 크다면 결과가 좋지 못하다는 단점이 있다. 현실에 존재하는 여러 개의 확률 분포를 혼합해서 표현하며, 몇 개의 데이터 확률 분포를 혼합할 것인지는 사용자가 지정하고 개수에 맞춰서 모델이 만들어지게 된다.

2.3 분석 체험

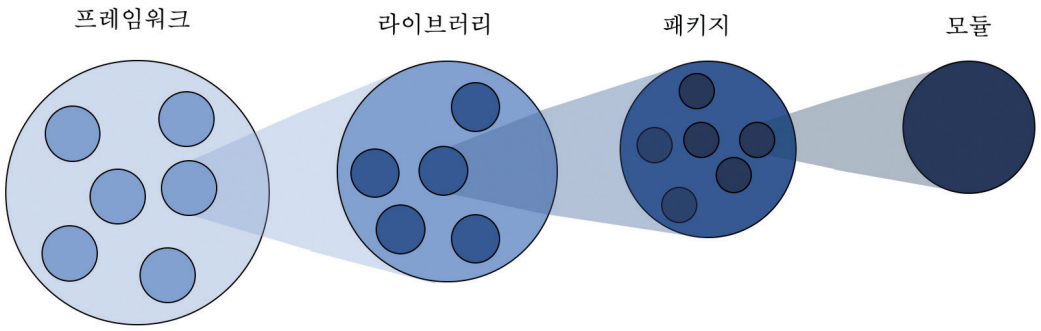
1) 필요 SW, 패키지 설치 방법 및 절차 가이드

• SW 설치는 '부록(분석환경 구축을 위한 설치 가이드)' 참고

• 패키지 설치 가이드

- 패키지 설치 전 파이썬 관련 용어 정리

프레임워크	– 라이브러리의 모음 – 프레임워크가 곧 프로그램의 구성요소이다.
라이브러리	– 여러 패키지의 모음 – 파이썬을 설치할 때, 기본적으로 설치되는 라이브러리를 표준라이브러리라고 한다. 파이썬 공식이 아닌 외부에서 개발한 모듈과 패키지를 묶어 외부 라이브러리라고 한다.
패키지	– 여러 모듈들의 모음 – 패키지 안에 여러 가지 폴더가 존재할 수 있다.
모 듈	– 특정 함수, 변수, 클래스 등이 구현되어 있는 파이썬 파이(.py)을 칭한다. 대표적으로 아래 모듈들이 존재한다. – 예) numpy: 수치해석 모듈, pandas: 데이터 분석 모듈



[그림 13] 파이썬 관련 용어 개념

- 패키지 설치 과정

① 주피터 노트북 내에서 패키지 및 모듈 설치하기

필요 패키지 및 모듈 설치

예) datetime 모듈 설치

```
!pip install datetime
```

```
Defaulting to user installation because normal site-packages is not writeable
Collecting datetime
  Downloading DateTime-4.3-py2.py3-none-any.whl (60 kB)
     |████████████████████████████████| 60 kB 1.3 MB/s eta 0:00:011
Requirement already satisfied: zope.interface in ./Python/anaconda3/lib/python3.8/site-packages (from datetime) (4.
7.1)
Requirement already satisfied: pytz in ./Python/anaconda3/lib/python3.8/site-packages (from datetime) (2020.1)
Requirement already satisfied: setuptools in ./Python/anaconda3/lib/python3.8/site-packages (from zope.interface->d
atetime) (49.2.0.post20200714)
Installing collected packages: datetime
Successfully installed datetime-4.3
Note: you may need to restart the kernel to use updated packages.
```

* pip install 패키지 및 모듈 이름 은 설치를 뜻하며 영구적이다.

```
!pip install pandas
!pip install numpy
!pip install matplotlib
!pip install scikit-learn
!pip install keras
!pip install datetime
!pip install seaborn
```

* 이번 분석을 위한 필요 패키지 및 모듈은 pandas, numpy, matplotlib,sckit-learn,keras,datetime 이기 때 문에 위와 같이 시작 전에 설치를 하면 된다.

- 모듈 설치 확인하기

```
pip list
```

```
cycler                        0.10.0
Cython                        0.29.21
cytoolz                       0.10.1
dask                          2.20.0
DateTime                      4.3
decorator                     4.4.2
defusedxml                    0.6.0
diff-match-patch              20200713
distributed                   2.20.0
docutils                      0.16
```

* pip list를 실행하면 현재 설치된 패키지 목록을 볼 수 있다. 직전에 설치한 'datetime'이 목록에 있는 것을 확인 가능

② 패키지 및 모듈 임포트 하기

예) 다양한 임포트 방법으로 Press_RawDataSet.xlsx를 읽어보기

i. import

import를 사용하여 해당 모듈 전체를 임포트한다.

```
import pandas
pandas.read_excel('./Press_RawDataSet.xlsx')
```

pandas를 임포트를 하고 '.'을 사용하여 pandas 의'read_excel' 함수를 이용하여 'Press_RawDataSet.xlsx' 파일을 불러온다.

ii. from, import

해당 모듈에서 특정한 타입만 임포트한다.

예) pandas에서 'read_excel'만 임포트

```
from pandas import read_excel
read_excel('./Press_RawDataSet.xlsx')
```

from, import를 사용해서 필요한 함수만 import로 사용할 수 있다.

iii. * import

해당 모듈 내에 정의된 모든 것을 임포트하나 다른 모듈들과 섞일 수 있으므로 일반적으로 사용이 권장되지 않는다.

```
from pandas import *
```

iv. as

모듈 임포트할 때, 별명(alias)를 지정할 때 사용한다.

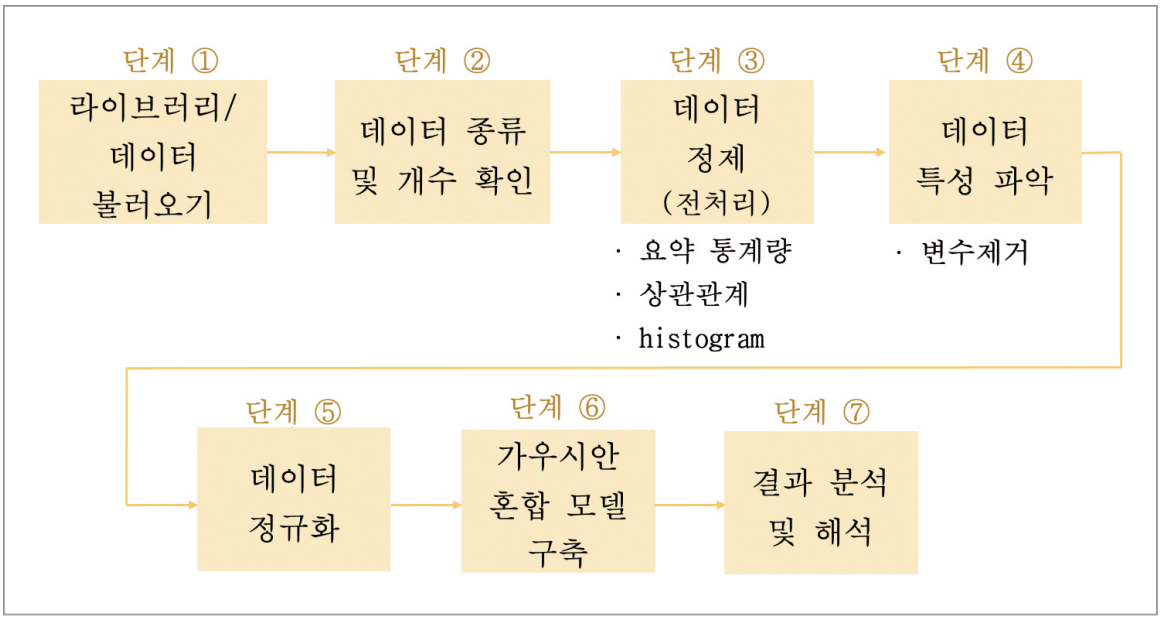
```
import pandas as pd
pd.read_excel('.Press_RawDataSet.xlsx')
```

'pandas'를'pd'로 별명 지정 후에는 'pd'로 불러올 수 있다.

* 한번 설치한 모듈 및 패키지는 영구적이며 필요시 버전 업그레이드가 필요하다. 또한, 분석, 시각화 등 상황에 따라 필요한 패키지를 임포트 한다.

2) 분석 단계별 프로세스

크게 아래의 흐름으로 데이터셋을 준비하고 모델을 구현할 수 있다.



[그림 14] 비지도 학습 적용을 위한 모델 구현 과정

[단계 ①] 라이브러리/데이터 불러오기

①-1. 필요 데이터 다운로드 및 불러오기(import) 가이드

```
import pandas as pd
import numpy as np
import datetime
import matplotlib.pyplot as plt
import sklearn
from sklearn.preprocessing import StandardScaler
from sklearn.mixture import GaussianMixture
import seaborn as sns
```

[코드 1] 필요 라이브러리 불러오기

- xlsx 파일을 불러올 때에는 pandas의 'read_excel('파일 경로')'함수를 사용한다. 프레스 데이터를 불러온다.

```
press_data = pd.read_excel('./Press_RawDataSet.xlsx')
```

[코드 2] 프레스 데이터 불러오기

- 프레스 데이터의 일자별로 발생한 불량의 수가 집계된 데이터를 불러온다.

```
press_error = pd.read_excel('./Press_error.xlsx')
```

[코드 3] 불량 수 데이터 불러오기

[단계 ②] 데이터 종류 및 개수 확인

②-1. 데이터 종류 및 개수 확인 가이드

- 데이터의 열과 일부 값을 확인하기 위하여 'head' 함수로 상위 5개의 데이터를 확인할 수 있다. 함수를 이용하여 프레스 데이터의 데이터를 확인한다.

```
press_data.head()
```

	idx	Machine_Name	Item No	working time	Press time(ms)	Pressure 1	Pressure 2	Pressure 5
0	1	Press-01	ED5260	2020-05-04	550.0	275.0	274.0	549.0
1	2	Press-01	ED5260	2020-05-04	550.0	275.0	274.0	549.0
2	3	Press-01	ED5260	2020-05-04	550.0	275.0	275.0	550.0
3	4	Press-01	ED5260	2020-05-04	550.0	275.0	275.0	550.0
4	5	Press-01	ED5260	2020-05-04	549.2	274.6	276.0	550.6

[코드 4] press_data의 상위 5개 데이터

- 'head' 함수를 이용하여 불량 수 데이터를 확인한다. defect type 1은 '주름발생', defect type 2는 '터짐발생', defect type 3은 '버 발생'을 의미한다. defect는 해당 불량 타입별 개수이고, 하루에 3종류의 불량 타입에 대한 개수가 집계된다.

```
press_error.head()
```

	idx	Machine_Name	Item No	working time	defect	defect type	Unnamed: 6
0	1	Press-01	ED5260	2020-05-04	0	1	주름발생
1	2	Press-01	ED5260	2020-05-04	1	2	터짐발생
2	3	Press-01	ED5260	2020-05-04	0	3	버 발생
3	4	Press-02	ED5260	2020-05-05	1	1	NaN
4	5	Press-03	ED5260	2020-05-05	1	2	NaN

[코드 5] press_error의 상위 5개 데이터

[단계 ③] 데이터 정제 (전처리)

③-1. 불필요 데이터 제거 가이드

- 입력 데이터에서 필요 없는 변수(feature) 제거를 위한 pandas 패키지의 'drop' 함수를 사용한다, 열(column, 해당 데이터에서는 변수를 의미함)의 이름을 아래와 같이 작성하고 'axis=1'로 두면 해당 열은 모두 지워지고, 'inplace=True'를 통해 현 상태를 불러온 프레스 데이터 (press_data)에 업데이트 한다. 참고로 'axis=0'은 행을 의미하고, 'inplace=False'로 입력할 경우 상태가 업데이트 되지 않고 데이터는 'drop'함수를 사용하기 전의 원래 형태를 유지하게 된다.

```
press_data.drop(['idx', 'Machine_Name', 'Item  No'], axis=1, inplace=True)
```

[코드 6] 일부 열 삭제

- 'dropna' 함수를 이용하여 값이 없는 데이터의 행(row)을 삭제한다. 마찬가지로 'axis=0'으로 설정하여 값이 누락된 행은 삭제하고, 현 상태를 유지('inplace=True') 한다.

```
press_data.dropna(axis=0, inplace=True)
```

[코드 7] 값이 누락된 행 삭제

[단계 ④] 데이터 특성 파악

④-1. 프레스 데이터/describe 함수를 통한 통계량 파악 가이드

- 프레스 데이터의 데이터구조 및 변수 모양을 'describe' 함수로 확인한다.

```
press_data.describe()
```

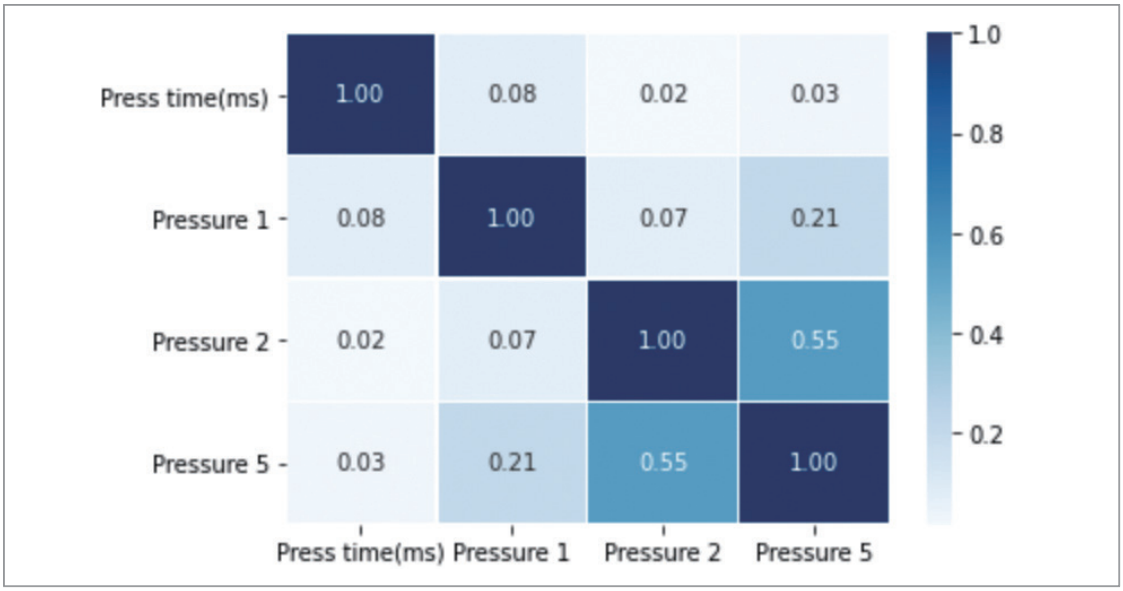
	Press time(ms)	Pressure 1	Pressure 2	Pressure 5
count	62687.000000	62687.000000	62687.000000	62687.000000
mean	550.286045	275.063257	269.826332	544.896529
std	22.443744	1.254687	3.177548	5.017656
min	25.500000	174.500000	167.000000	144.600000
25%	549.800000	274.900000	267.000000	542.100000
50%	550.000000	275.000000	269.000000	543.900000
75%	550.400000	275.200000	273.100000	548.100000
max	3550.200000	286.900000	277.000000	941.400000

[코드 8] 데이터 변수 구조 확인하기

④-2. 프레스 데이터/corr 함수를 통한 통계량 파악 가이드

- 변수간 상관관계를 'corr' 함수를 이용하여 시각화 한다.변수별 상관성이 높지 않다는 것을 확인할 수 있다.

```
# 변수들 사이 상관 관계로 히트맵 그리기
sns.heatmap(data = press_data.corr(), annot =True, fmt ='.2f', linewidths =.5, cmap ='Blues')
plt.show()
```



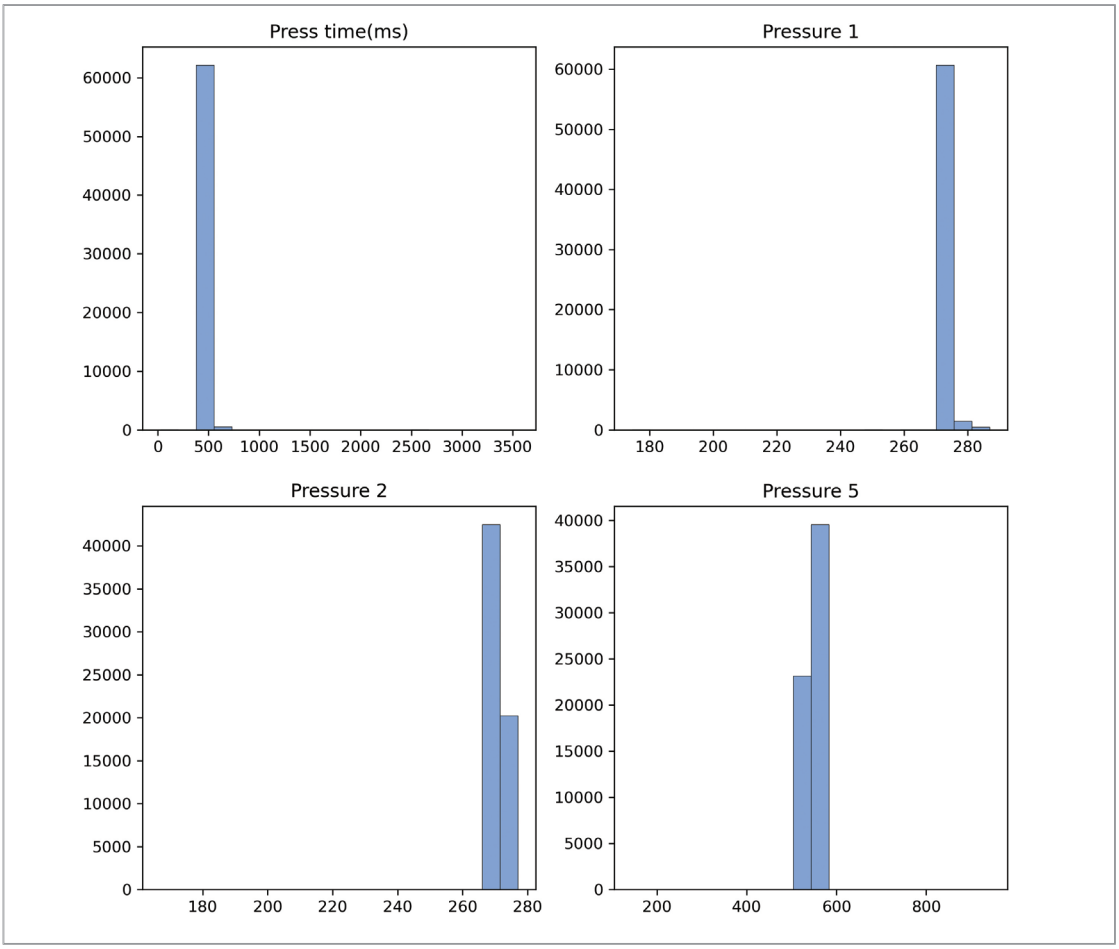
[코드 9] 변수들 사이 상관관계 _ 히트맵

④-3. 프레스 데이터/Histogram을 통한 변수별 데이터 파악 가이드

- 데이터의 변수 'working time'를 제외한 나머지 변수별 히스토그램을 확인한다. 각 변수별로 막대그래프의 개수를 설정해주고, 데이터의 각 변수(총 4개)마다 해당 막대그래프의 개수가 할당될 수 있도록 index와 value로 설정해준다. matplotlib 패키지의 pyplot(plt로 표현)을 이용하여 히스토그램을 그려준다. subplot() 함수를 이용하여 한번에 모든 변수의 히스토그램 결과를 볼 수 있도록 한다.

```
press_hist = press_data.drop(['working time'], axis=1)

plt.figure(figsize = (10,10))
# 각 변수의 막대그래프 개수
bin = [20,20,20,20]
for index, value in enumerate(press_hist):
    sub = plt.subplot(2,2,index +1)
    sub.hist(press_hist[value], bins = bin[index], facecolor = (144/255,171/255,221/255), linewidth=.3,
edgecolor ='black')
    plt.title(value)
```



[코드 10] 프레스 데이터의 클래스 변수별 히스토그램 확인

④-4. 불량 수 데이터/groupby 함수를 통한 일자 및 불량 타입별 불량 데이터 수 파악 가이드

- ‘groupby’ 함수를 이용하여 일자별 불량의 개수를 확인한다. 일자(‘working time’)과 불량 타입(‘defect type’)을 함수의 인자로 넣어 일자 및 타입별 불량의 수를 확인할 수 있다.

```
press_error.groupby(['working time', 'defect type']).sum()
```

		defect
working time	defect type	
2020-05-04	1	0
	2	1
	3	0
2020-05-05	1	1
	2	1
	3	0
2020-05-06	1	1
	2	0
	3	0
2020-05-07	1	0
	2	0
	3	1

[코드 11] 일자 및 불량 타입별 불량의 개수

④-5. 불량 수 데이터/groupby 함수를 통한 불량 타입별 데이터 수 파악 가이드

- 'groupby' 함수를 이용하여 불량 타입별 불량의 개수를 확인한다. 프레스 데이터 내에서 불량 타입 1(주름발생)의 총 개수는 29개, 불량 타입 2(터짐발생)의 총 개수는 23개, 불량 타입 3(버 발생)의 총 개수는 11개임을 알 수 있다.

```
press_error.groupby('defect  type')['defect'].sum()
```

```
defect  type
1      29
2      23
3      11
Name:  defect,  dtype:  int64
```

[코드 12] 불량 타입별 불량의 개수

[단계 ⑤] 데이터 정규화

⑤-1. StandardScaler를 통한 데이터 정규화 가이드

- sklearn 패키지의 정규화(standardization) 함수를 사용하여 데이터를 정규화한다. raw data는 변수마다 수치의 크기와 평균, 분산 등의 통계치가 다르므로 모델의 성능에 악영향을 미칠 수 있다. 정규화를 수행함으로써 모델의 성능을 제고하고 데이터를 정제하는 결과를 얻게 된다. 'working time' 열(index 값: 0)을 제외한 나머지 열(index 값: 1,2,3,4)만 가우시안 혼합 모델에 입력되므로 'iloc' 함수 내부에 해당 열의 index 값을 넣어 원하는 열 4개가 존재하는 데이터를 설정하고, 정규화를 실행한 후 변수 이름을 'input_data'로 한다.

```
scaler = StandardScaler()
input_data= scaler.fit_transform(press_data.iloc[:,[1,2,3,4]])
```

[코드 13] 정규화

[단계 ⑥] 가우시안 혼합 모델 구축

⑥-1. 가우시안 혼합 모델 구축 가이드

◦ AI 분석모델 구축을 위한 방법론(알고리즘) 적용 및 학습 네트워크 구축 실습

- 가우시안 혼합 모델 구축

  sklearn 패키지의 가우시안 혼합 모델(GMM)을 사용하여 모델을 불러오고, 예상 군집의 개수를 n_components에 할당한다. 현재 데이터는 정상 데이터와 불량 데이터로 크게 분류할 수 있지만, 불량 데이터는 3개의 불량 타입이 있기 때문에 총 4개의 군집이 있다고 가정하고 n_components 값을 4로 할당한다. 'fit' 함수를 이용하여 가우시안 혼합 모델에 데이터를 학습시키고, 'predict' 함수로 해당 데이터의 예측값을 'gmm_labels' 변수에 저장한다.

```
gmm = GaussianMixture(n_components = 4)
gmm.fit(input_data)
gmm_labels = gmm.predict(input_data)
```

[코드 14] 가우시안 혼합 모델 구축

- 학습된 모델의 예측값인 'gmm_labels'를 이용하여 군집별 할당된 데이터의 개수를 확인한다. 데이터가 적은 하위 3개(군집 번호가 각 0, 1, 3)의 군집이 3개의 불량 타입에 해당된다고 볼 수 있다.

```
press_data['gmm_cluster'] = gmm_labels
press_data['gmm_cluster'].value_counts()
```

```
1     61605
2      1067
3        12
0         3
Name: gmm_cluster, dtype: int64
```

[코드 15] 모델 훈련

- 4개의 군집의 변수별 통계량을 확인한다. 군집 하나를 'data_in_cluster'에 불러온 후, 유의미한 4개의 변수 'Press time(ms)', 'Pressure 1', 'Pressure 2', 'Pressure 5'만 확인할 수 있도록 통계량을 확인 시 불필요한 변수('gmm_cluster')를 제외한다. 이후 4개의 변수에 대한 평균과 분산을 각각 함수 'mean'과 'var'로 확인할 수 있다.

```
clusters = [0, 1, 2, 3]
for cluster in clusters: # 군집 하나씩 가져오기

    # 프레스 데이터를 군집별로 'data_in_cluster'에 불러오기
    data_in_cluster = press_data[press_data['gmm_cluster'] == cluster]
    print("+"*20) # 군집별 구분을 위한 기호
    print("cluster 번호 {}₩n".format(cluster))
    for col in data_in_cluster.columns: # 변수 하나씩 가져오기
        if col !='working time' and col !='gmm_cluster':
            print(col) # 변수 이름
            print("mean: {:.2f}".format(data_in_cluster[col].mean())) # 해당 변수의 평균
            print("variance: {:.2f}₩n".format(data_in_cluster[col].var())) # 해당 변수의 분산
```

```
++++++++++++++++++++
cluster 번호 0

Press time(ms)
mean: 549.80
variance: 0.52

Pressure 1
mean: 174.90
variance: 0.13

Pressure 2
mean: 270.00
variance: 13.00

Pressure 5
mean: 544.90
variance: 13.03

++++++++++++++++++++
cluster 번호 1 (정상)

Press time(ms)
mean: 550.00
variance: 0.39

Pressure 1
mean: 275.00
variance: 0.10

Pressure 2
mean: 269.81
variance: 8.74

Pressure 5
mean: 544.81
variance: 8.77

++++++++++++++++++++
cluster 번호 2 (정상)

Press time(ms)
mean: 558.01
variance: 171.94

Pressure 1
mean: 279.00
variance: 42.98

Pressure 2
mean: 270.56
variance: 87.96

Pressure 5
mean: 549.69
variance: 950.41

++++++++++++++++++++
cluster 번호 3

Press time(ms)
mean: 1333.67
variance: 2176016.59

Pressure 1
mean: 275.15
variance: 0.23

Pressure 2
mean: 270.43
variance: 8.86

Pressure 5
mean: 545.58
variance: 8.75
```

[코드 16] 군집의 변수별 통계량 확인

[단계 ⑦] 결과 분석 및 해석

⑦-1. 분석결과에 대한 해석 가이드

◦ 분석결과에 대한 논의 및 해석(implication)

- 가우시안 확률 분포로 구성된 4개의 군집의 변수별 통계량을 비교함으로써 군집마다 유의미한 차이가 있음을 볼 수 있다. 군집 2의 변수별 평균과 분산을 통해 대부분의 데이터(전체 프레스 데이터 중 43669개)가 해당 군집에 몰려있어 정상 데이터를 가진 군집으로 분류할 수 있으며, 군집 2와 비교했을 때 군집 1은 변수 'Pressure 2'와 'Pressure 5'의 분산 부분에서, 군집 3은 모든 변수의 통계 결과

에서 눈에 띄는 차이가 있어 예외 군집, 즉 불량임을 암시할 수 있다. 군집 0의 경우 군집 2 다음으로 많은 데이터(전체 프레스 데이터 중 19753개)로 구성되어 있는데, 군집 2와 변수별 통계량을 비교하였을 때에도 군집 2와 군집 1, 군집 3의 차이보다는 상대보다 적은 차이가 나는 것을 확인할 수 있다. 클래스 변수가 없는 프레스 데이터를 가우시안 혼합 모델로 분석한 결과, 대부분의 데이터가 속할 것으로 예상되는 군집 2를 제외한 나머지 군집들에 속하는 데이터들은 모두 20690개(19753+496+441개)로, 해당 모델의 가정으로는 불량에 해당하는 군집의 데이터의 총합이 실제 불량의 수인 63개(29+23+11개)와 큰 차이가 있다. 그 이유로 프레스 공정의 특성을 제시할 수 있는데, 공정 특성상 불량 발생률이 상당히 낮고 불량이 발생할 수 있는 조건에서도 실제로 불량이 나지 않는 경우 또한 많은 것으로 알려져 있다. 이와 같은 문제점을 보완하기 위해 불량 데이터를 더 많이 수집하거나, 소수의 클래스 변수가 있는 데이터를 수집함으로써 모델의 성능을 높일 수 있다.

3. 유사 타현장의 「프레스 AI능 데이터셋」 분석 적용

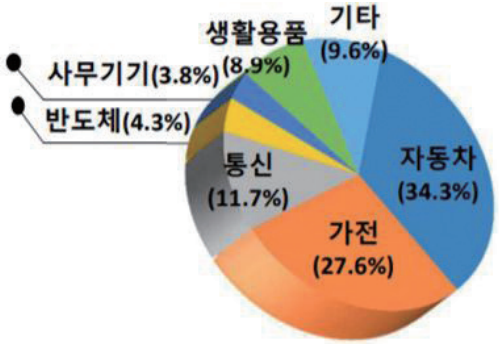
3.1 본 분석이 적용 가능한 제조현장 소개

1) 프레스 가공 장.단점

- 프레스 가공은 프레스기계, 금형, 피가공재의 3가지에 의해 원하는 제품을 얻을 수 있으며, 그 중 제품의 품질이 결정되는 중요한 요소는 금형의 완성도에 따라 결정된다.
- 재료에 에너지를 가하여 원하는 형상으로 변형시키는 것이며, 프레스 가공은 제품의 강도가 높고 경량이며, 재료의 이용률이 좋다.
- 생산성이 높은 가공법으로 제품의 정도가 높고 균일성 있는 제품을 대량으로 생산할 수 있는 특징을 가지고 있다
- 단점으로는 고가의 프레스 금형이 필요하며, 금형제작에 장시간소요 및 다품종 소량생산체제에서는 원가가 높은 단점이 있다.

2. 적용 분야

- 산업 현황



구 분	2012	2013	2014	2015	2016
플라스틱금형	30,616	34,299	36,529	35,366	36,265
프레스금형	2,476	27,424	33,409	36,249	40,286
다이캐스팅금형	1,325	1,391	1,562	1,494	1,514
기타금형	9,226	8,556	9,503	9,512	8,380
금형부품	7,675	5,613	5,790	5,670	5,193
계	51,318	77,283	86,793	88,291	91,638

[그림 15] 프레스 금형 산업 현황 통계

15년 이후 자동차 산업 생산 비중 강화 및 중국 등 신흥국 성장 영향으로 프레스 금형이 플라스틱 금형을 역전하는 현상이 발생하였다.

전기, 전자, 자동차, 생활용품등 공산품을 제조하는 대부분의 산업이 금형 산업과 연관되어 있으며, 기계 설비, 공구, 금형부품 및 소재 등을 공급하는 산업 또한 금형 산업과 연관관계가 있다.

3.2 본 「프레스 AI 데이터셋」 분을 원용하여 타 제조현장 적용 시, 주요고려사항

- 프레스 가공분야의 대부분에 적용될 것으로 생각되나. 로봇활용한 자동화 공정에서는 프레스의 압력, 속도 뿐 아니라 로봇에 의해 이송되는 자재의 위치 데이터의 정확한 수집과

자재의 길이, 자재의 두께등의 데이터를 수집하여 분석하는데 포함하여야 제품이 생산되는데 불량을 줄이고, 최상의 품질을 얻을 수 있는 모델을 만들 수 있을 것으로 판단된다.

- 수집하는 데이터의 형태에 대한 논의 및 관리가 필요하다. 현재는, 현장에서 수집되는 데이터와 AI 분석을 위한 데이터에는 많은 전처리 과정이 필요하다. 데이터 수집을 하면서 현장 실무자와 AI 분석 전문가가 꾸준히 커뮤니케이션하면서 데이터 수집 방향 및 형태에 대해서 논의하여 수집 환경을 구축하는 것이 중요하다.

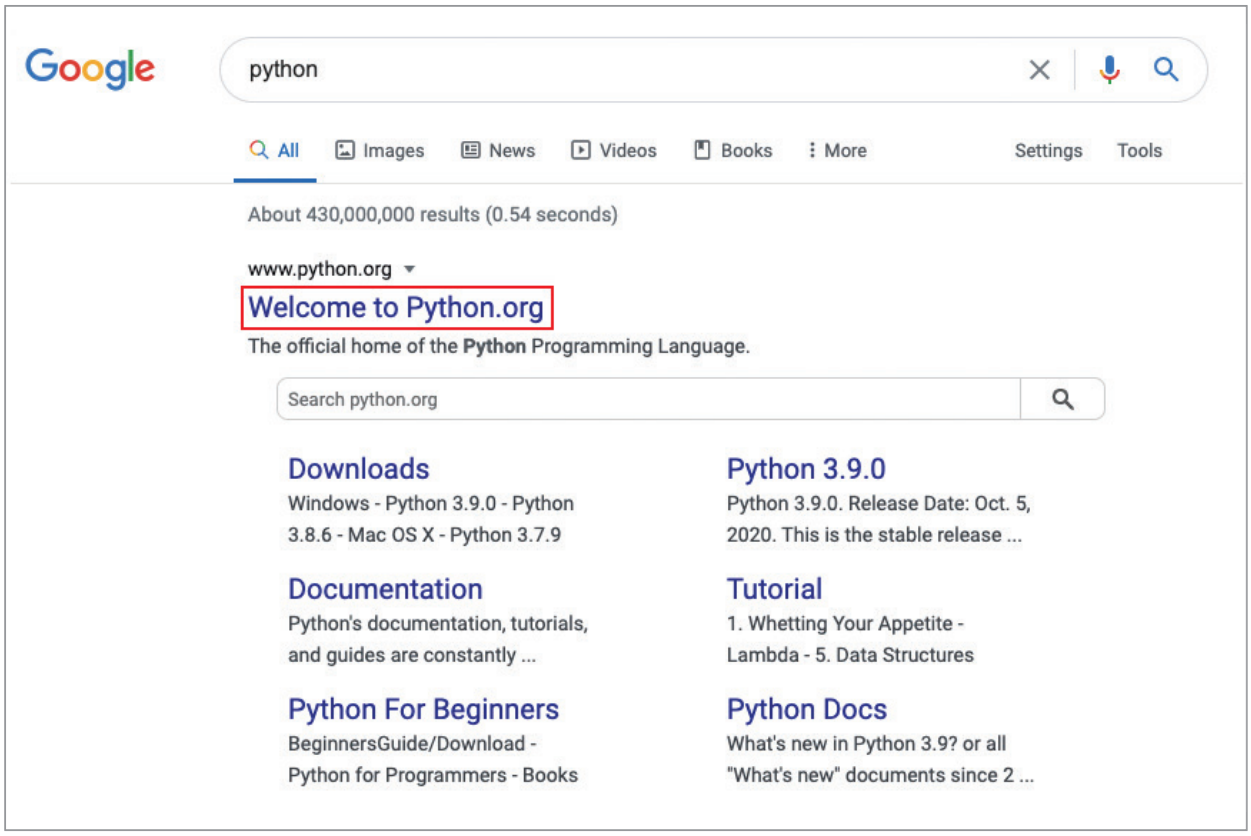
부록 파일 설치 방법

1. 파이썬(python) 설치

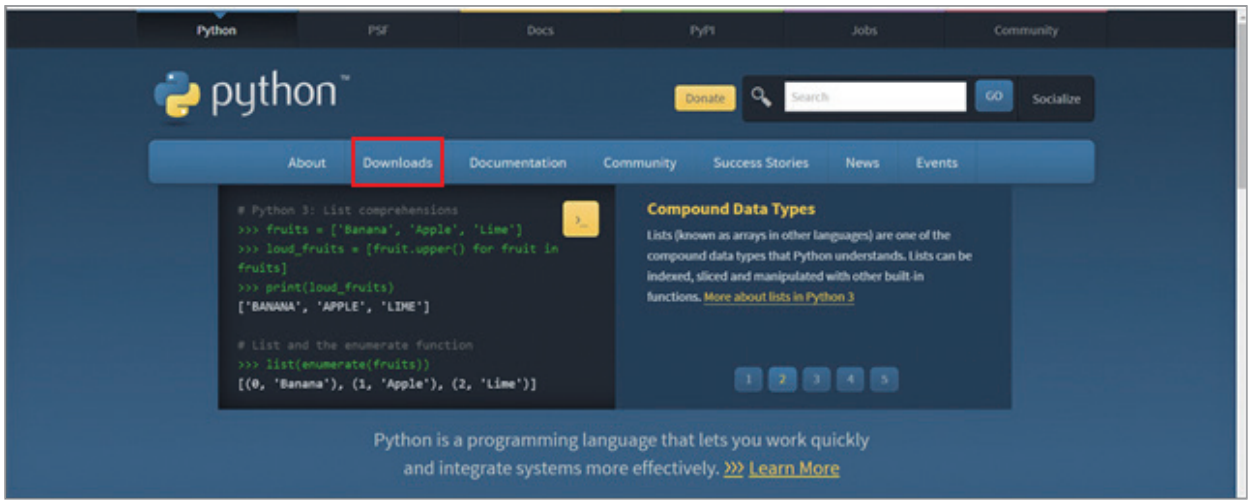
파이썬이란, 컴퓨터 언어 및 데이터 분석에 활발하게 쓰이는 도구입니다. 데이터 분석을 위해서 다운로드 및 설치가 간편하고 활용도가 높은 파이썬을 설치하고 적용하여 봅니다.

① google.com 등의 검색 엔진에 'python'을 검색

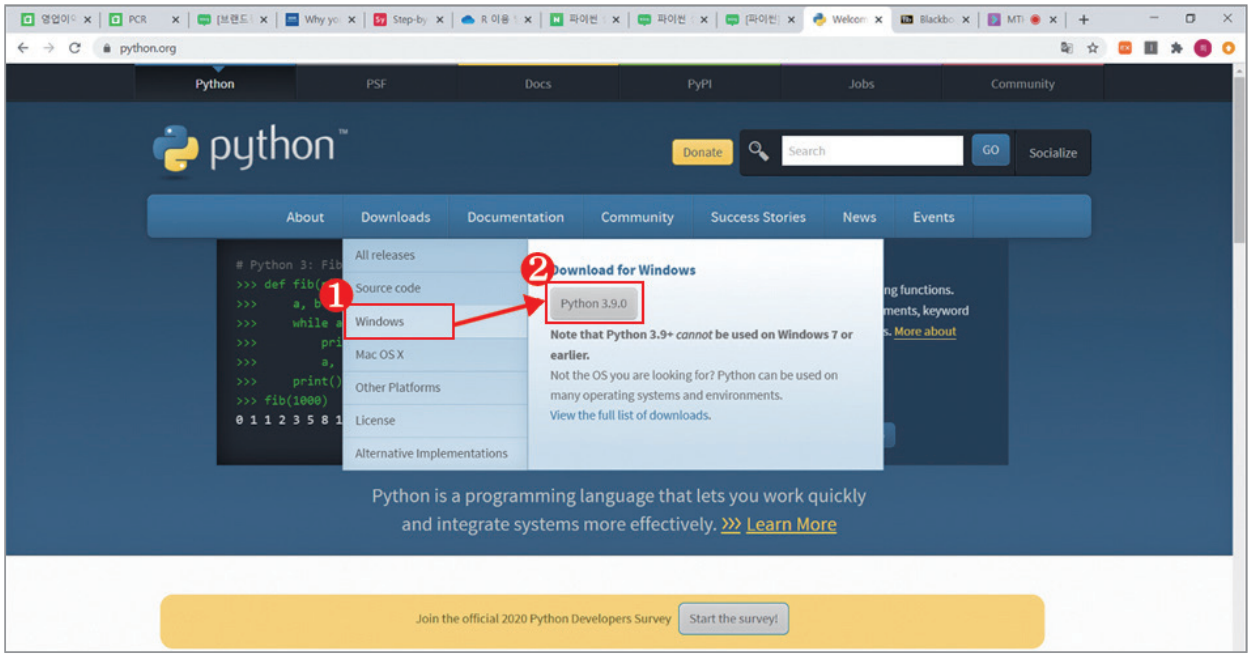
② 제일 처음에 보이는 'Welcome to python.org'를 클릭



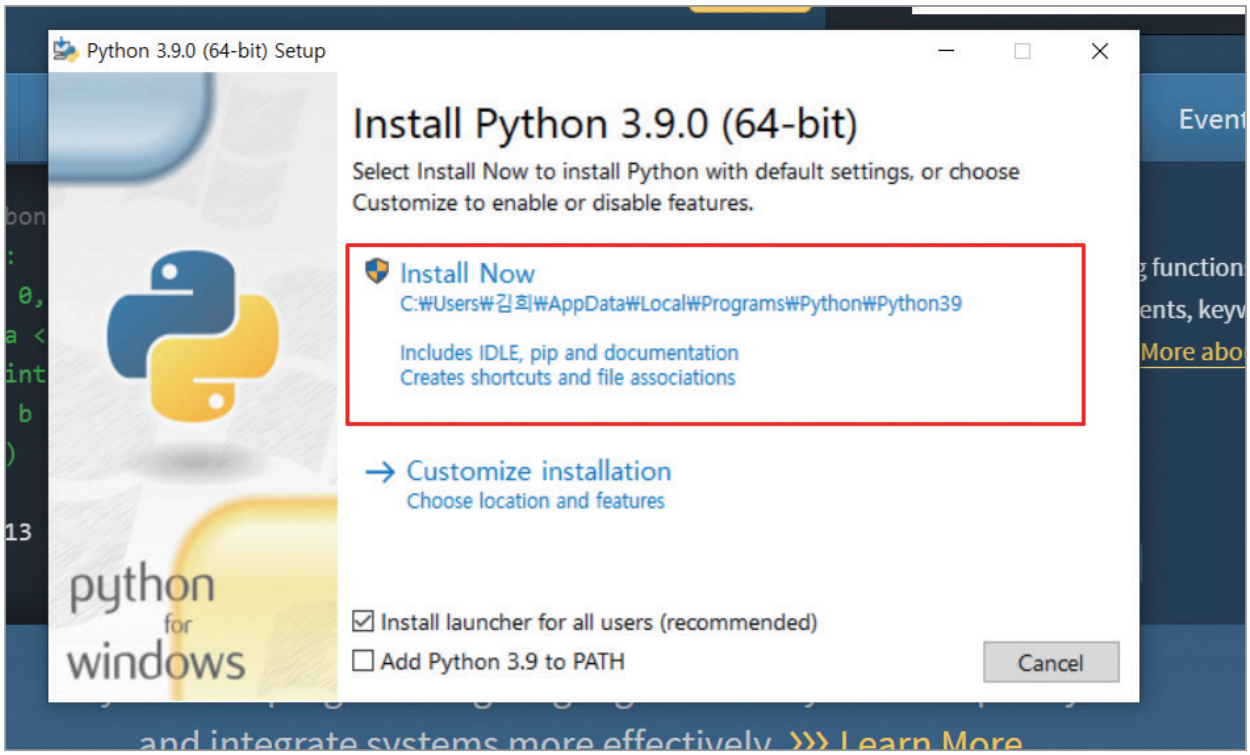
③ 클릭해서 보이는 페이지 정면의, 왼쪽 2번째 'Downloads' 클릭



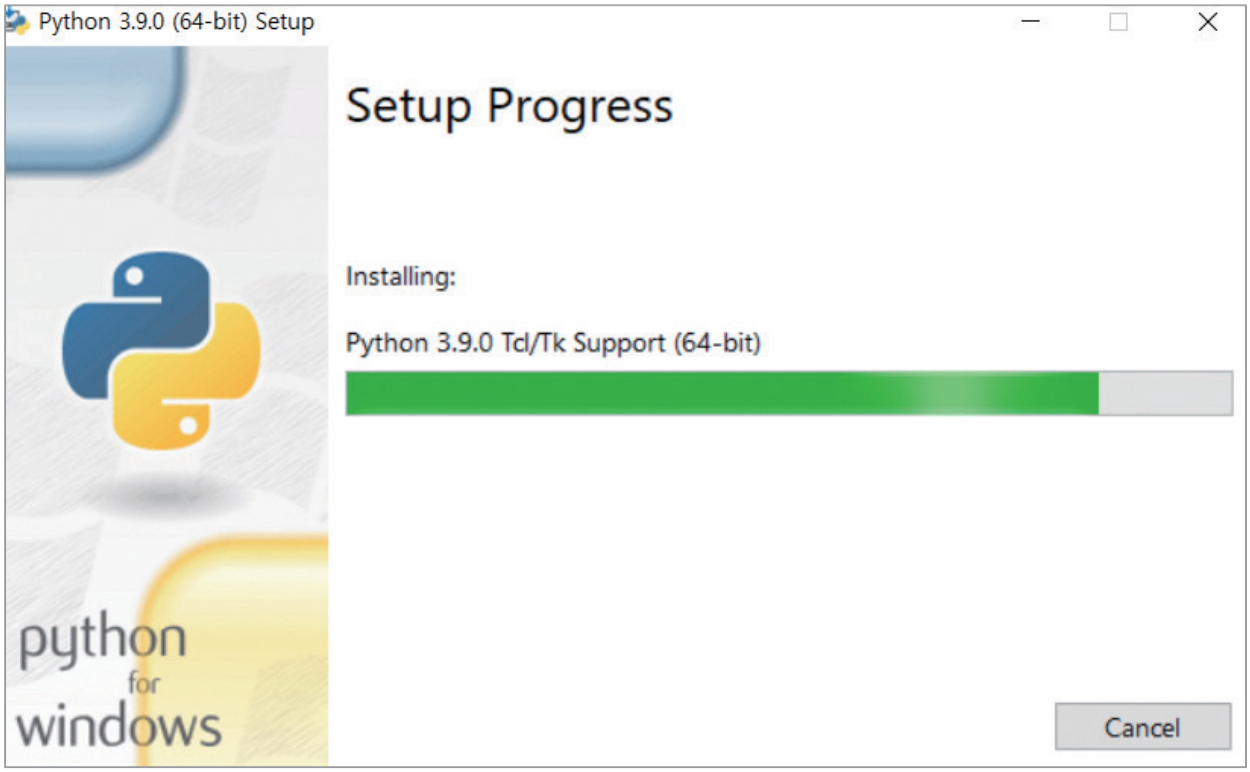
④ 위에서 3번째, Windows 탭을 선택한 후, python3.9.0 다운로드
(python3.9.0은 숫자가 업데이트 될 수 있습니다)



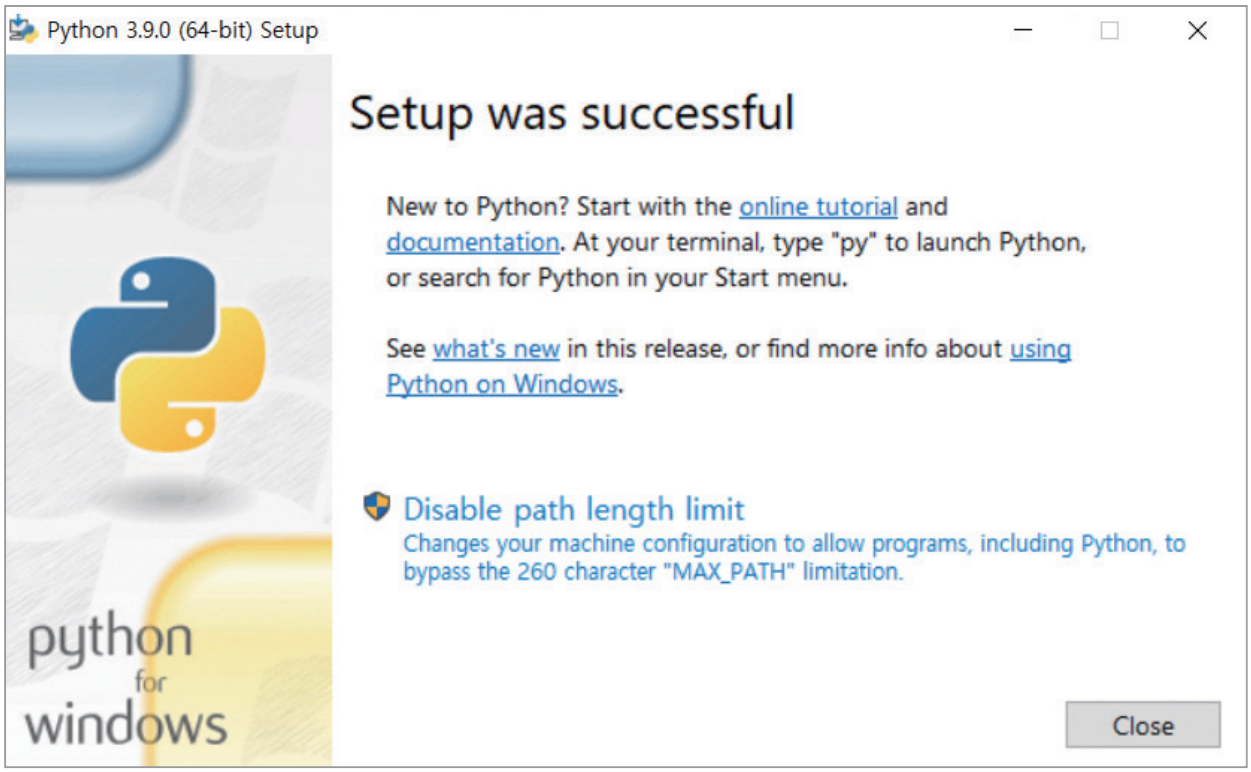
⑤ 아래와 같은 설치창이 뜨면, 'Install Now'를 클릭



⑥ 아래와 같은 설치 진행창이 완료가 될 때까지 유지



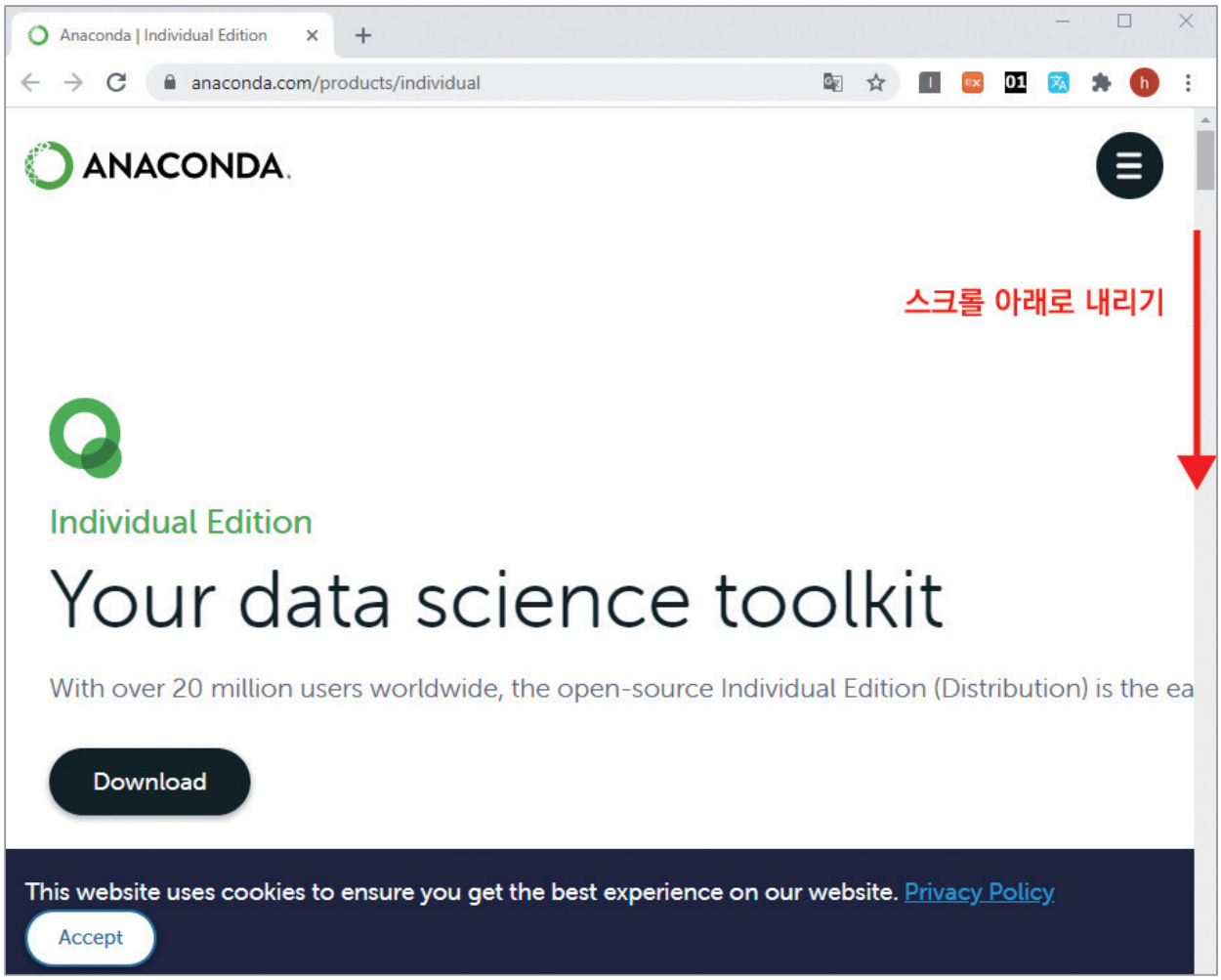
⑦ 완료가 되면 아래와 같은 창이 뜨는 것 확인 후 종료 **[설치완료]**



2. 아나콘다(anaconda) 설치

아나콘다란? 파이썬과 같은 분석 도구를 사용할 때 필요한 고급 기능 및 분석을 보조하는 도구입니다. 아나콘다를 설치함으로써 많은 기능들을 바로 쓸 수 있고, 결과물들 또한 쉽게 볼 수 있는 기능을 지원합니다. 아나콘다를 설치하고 분석을 할 수 있는 환경을 만들어봅니다.

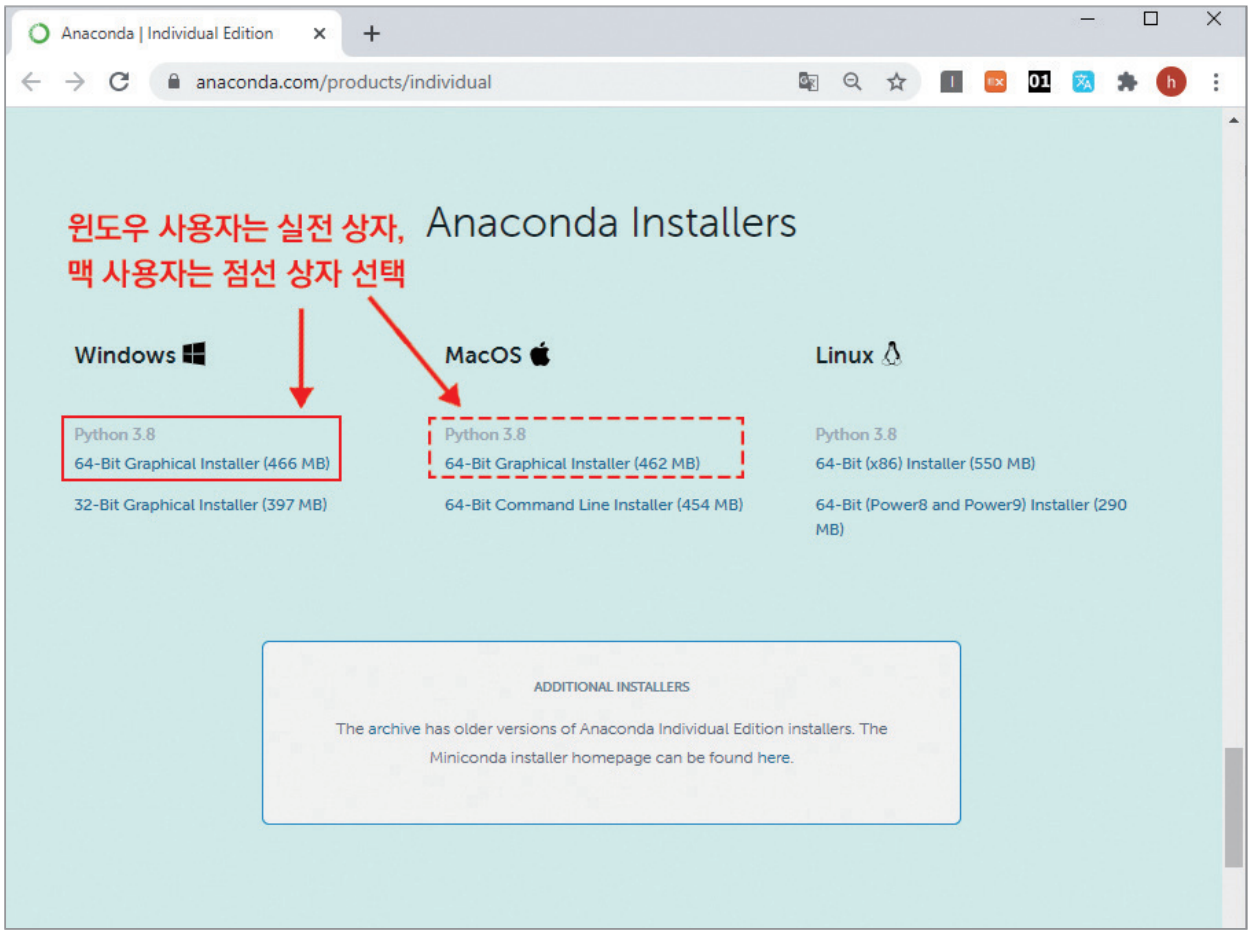
① https://www.anaconda.com/distribution/ 로 접속 후 스크롤 내림



② 스크롤을 다음과 같은 화면이 나올 때 까지 아래로 내린 후, 컴퓨터 사용환경에 맞는 파일 다운받기 (본 부록은 Windows 설치 기준)

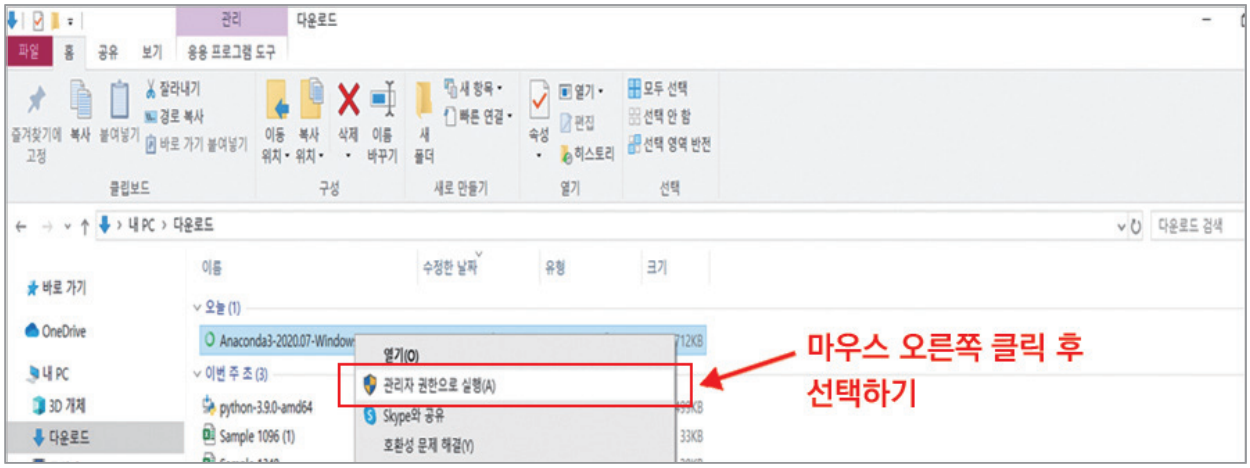
▶ Windows : 64-Bit Graphical Installer

▶ MacOS : 64-Bit Graphical Installer

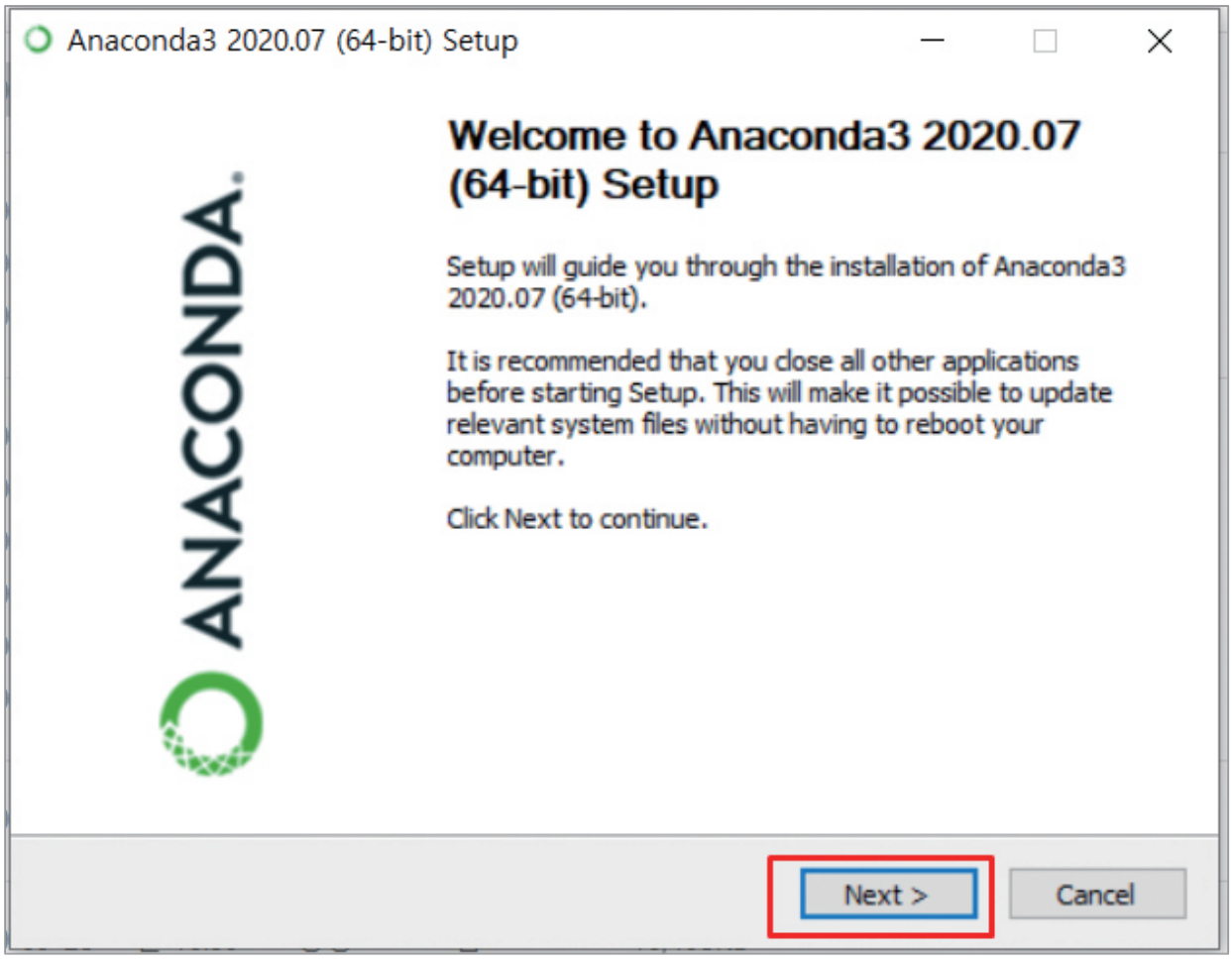


③ 다운을 받은 파일에 가서, 아나콘다 설치 파일 위에서, **마우스 오른쪽을 클릭**한 후, 방패모양의 **'관리자 권한으로 실행'** 선택

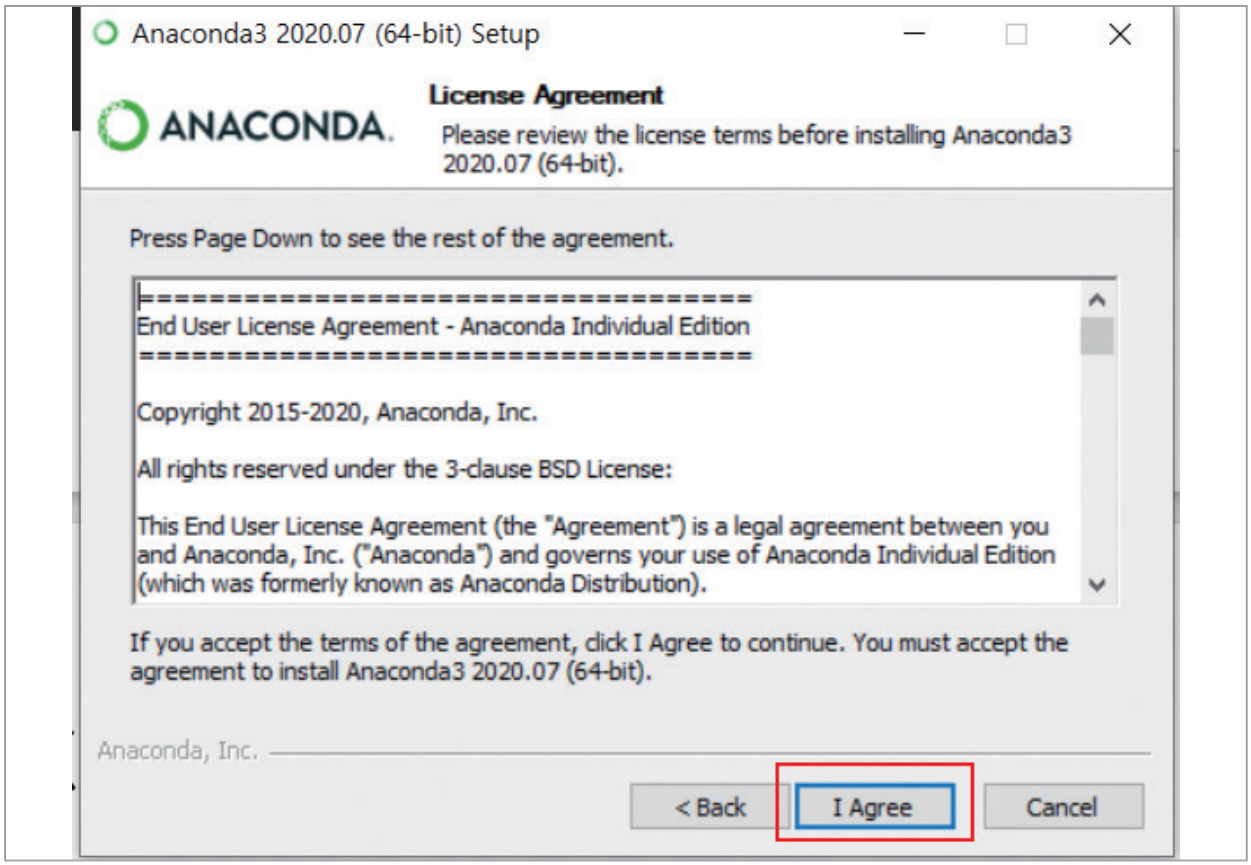
▶ **(예) '다운로드' 파일로 아나콘다를 다운 받은 경우**



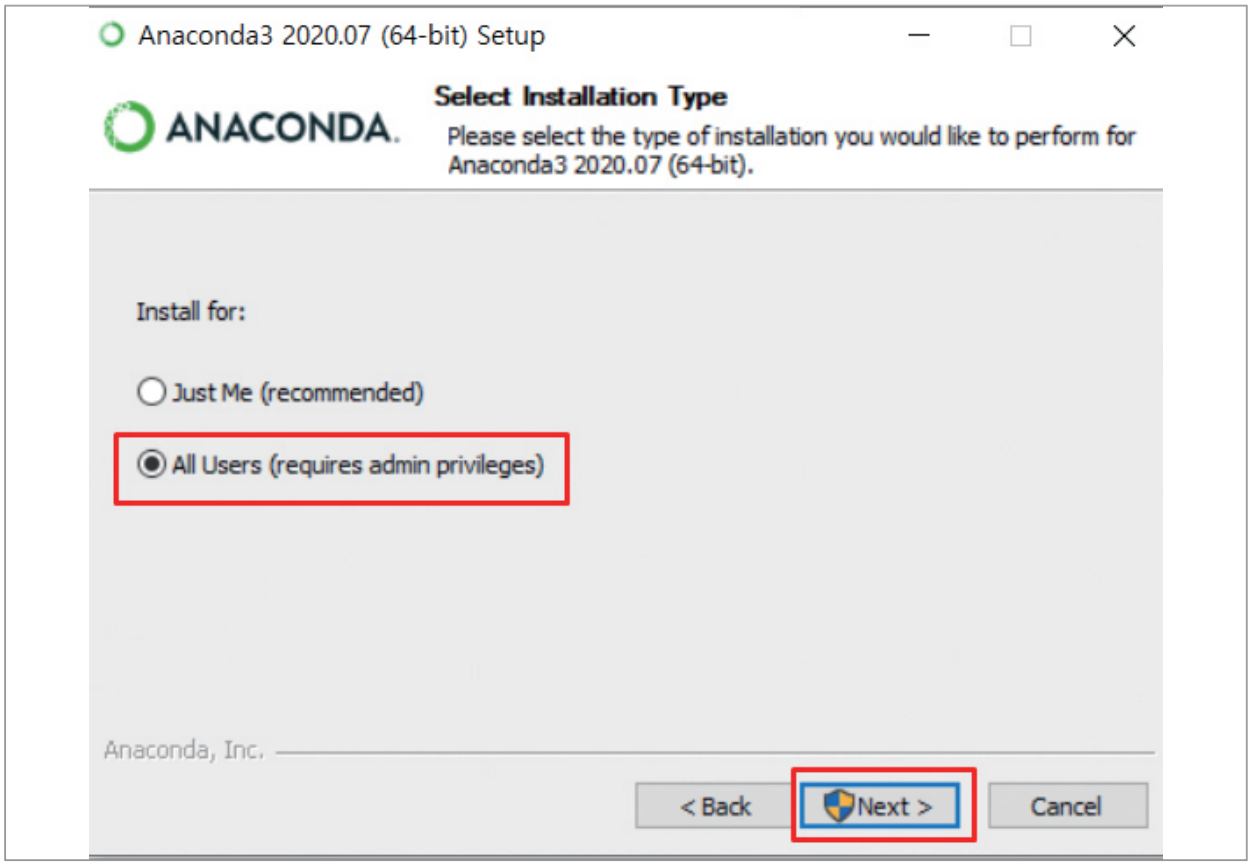
④ 파일을 실행 한 후, 'Next' 버튼을 클릭



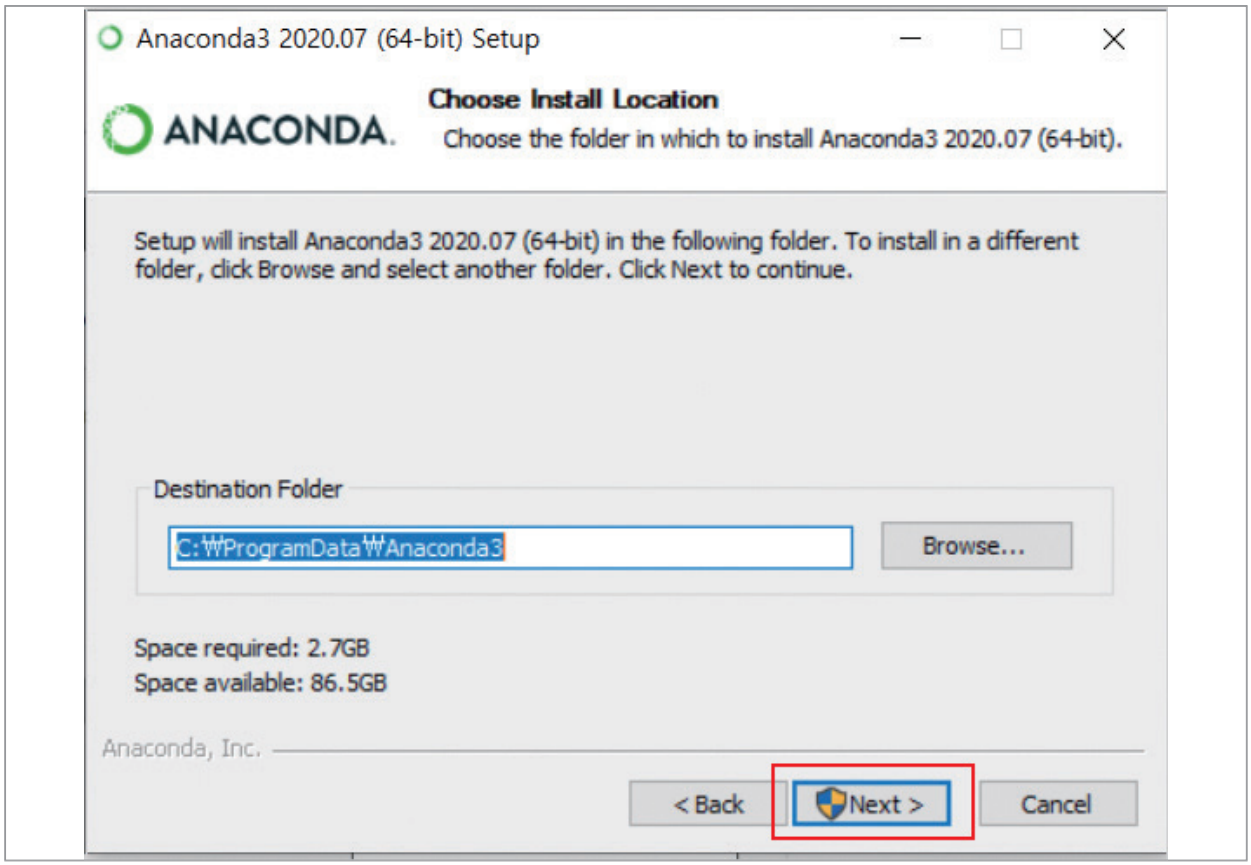
⑤ 다음 창이 나타나면 'I agree'를 선택



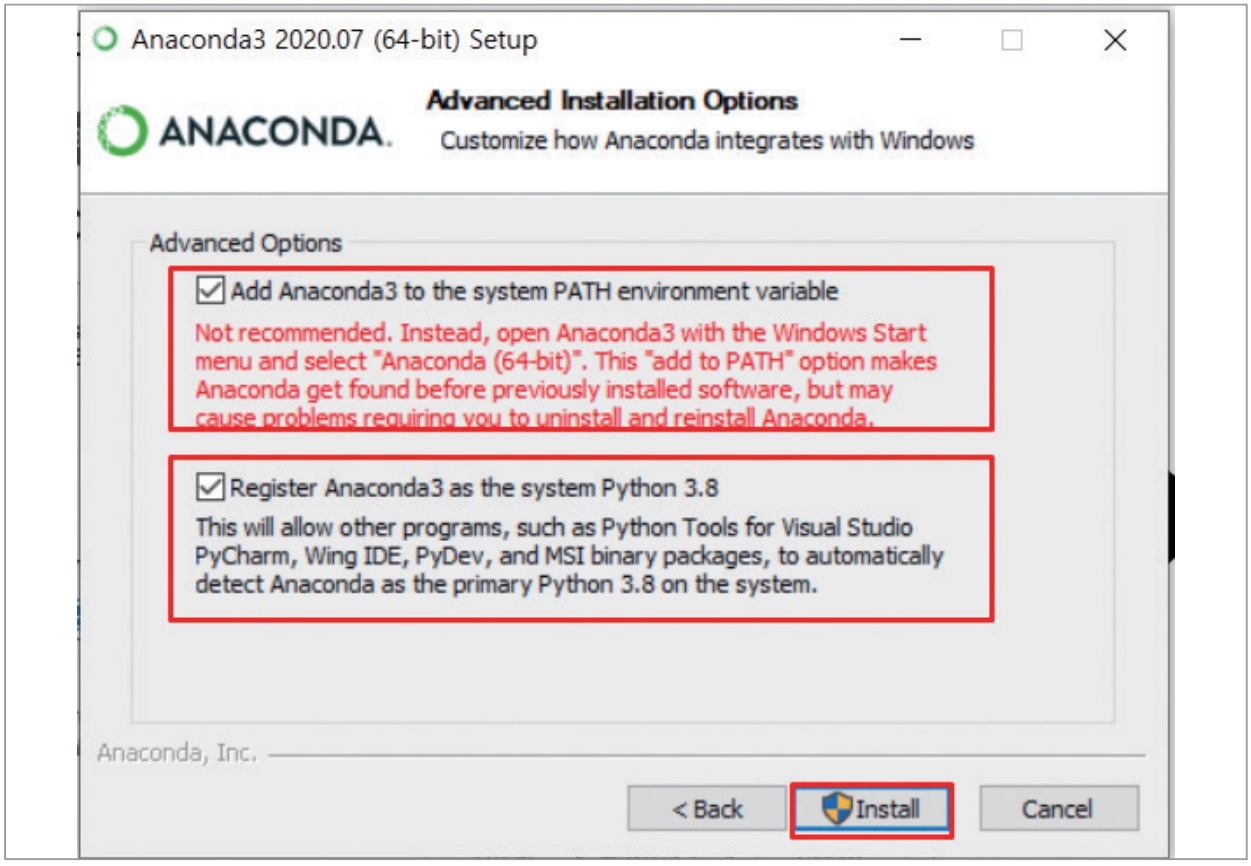
⑥ 셋팅 창이 뜨면 화면의 'All Users'를 선택 후 아래의 'Next' 클릭



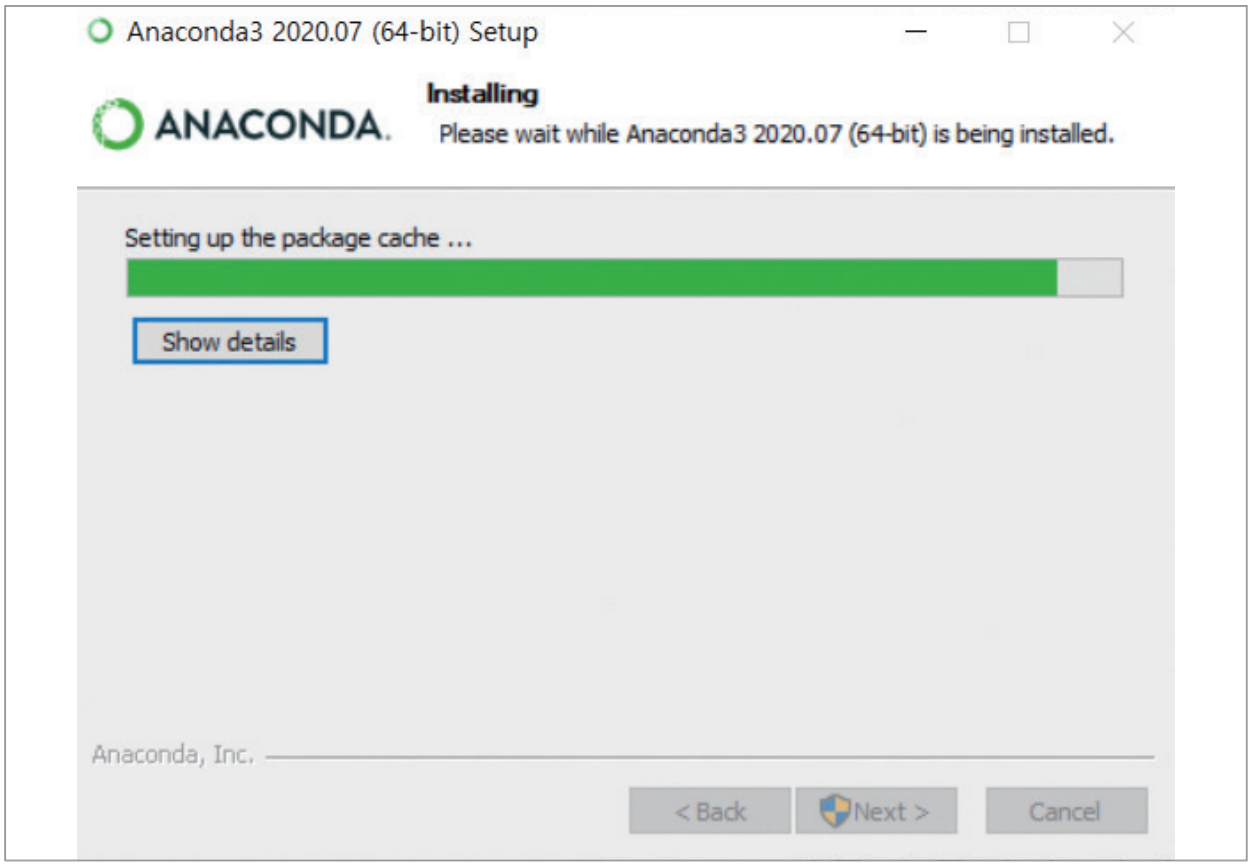
⑦ 다운로드 받을 경로를 물어보는 창이 뜨면, 아래의 'Next' 클릭



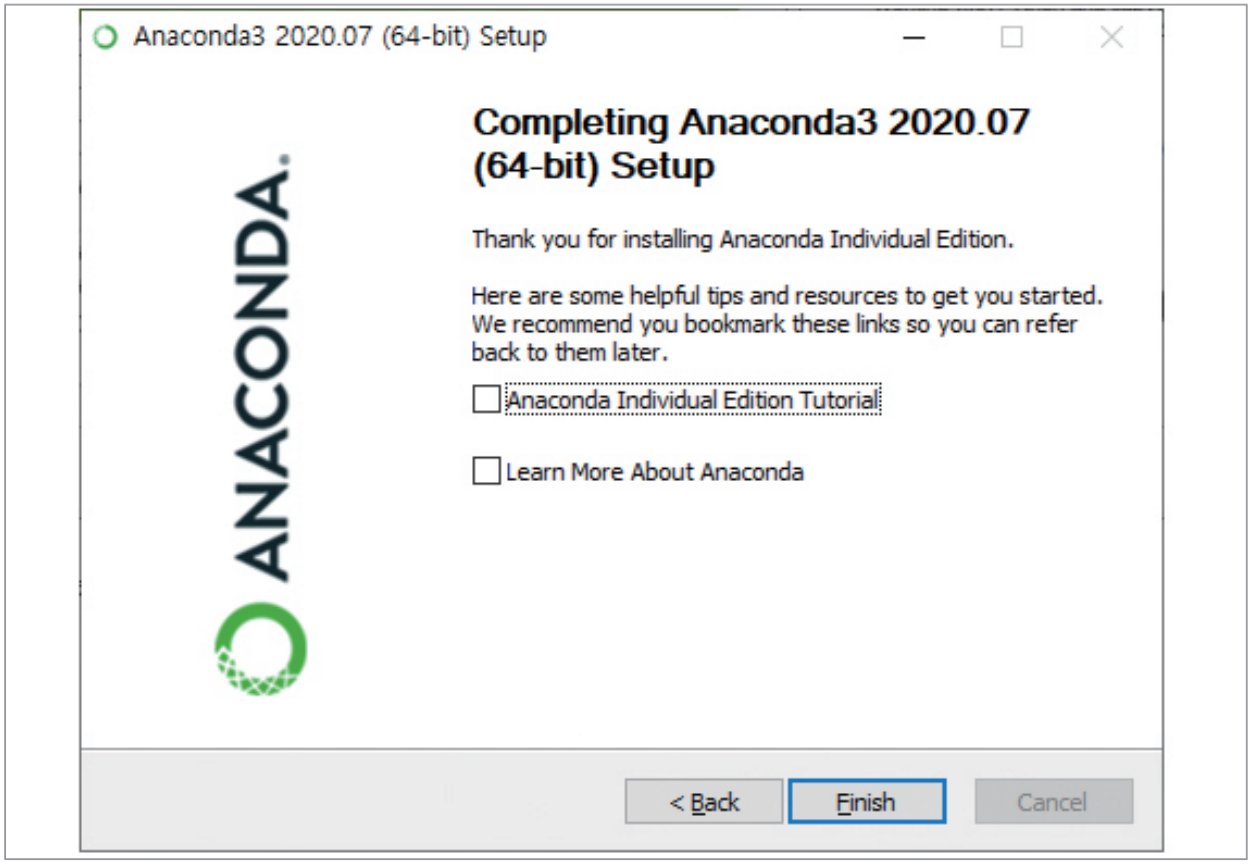
⑧ 고급 옵션 선택창이 뜨면, 아래와 같이 모두 선택 후, 'Install' 클릭



⑨ 다음과 같은 설치창이 뜨면 완료가 될 때까지 대기 (5분이상 소요)

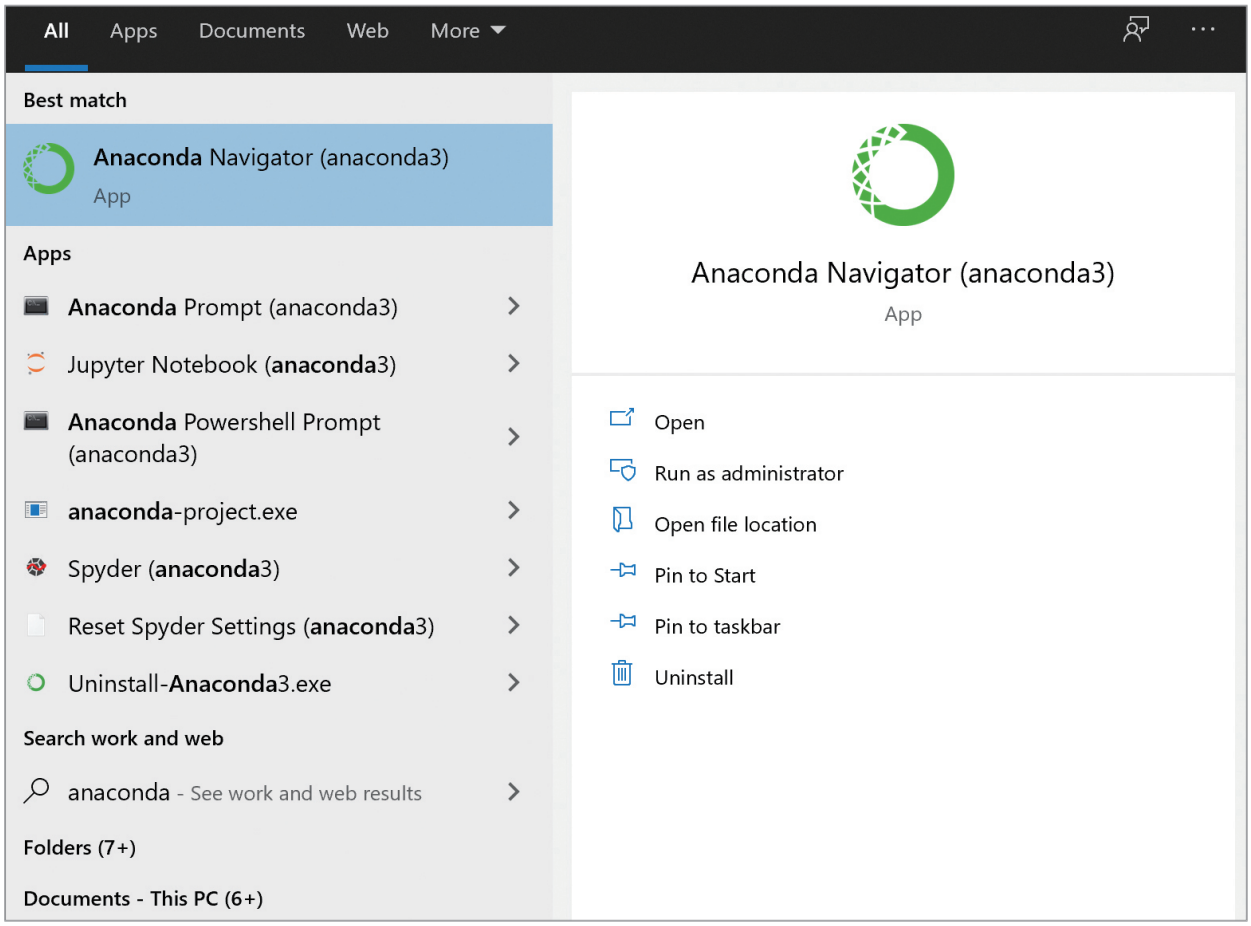


⑩ 마지막 화면에서, 모두 체크 해제한 후, 'Finish' 눌러 설치 완료



⑪ 설치 확인하기

화면상의 '홈( )'키를 눌러서 화면과 같이 anaconda prompt가 잘 깔렸는지 확인하기

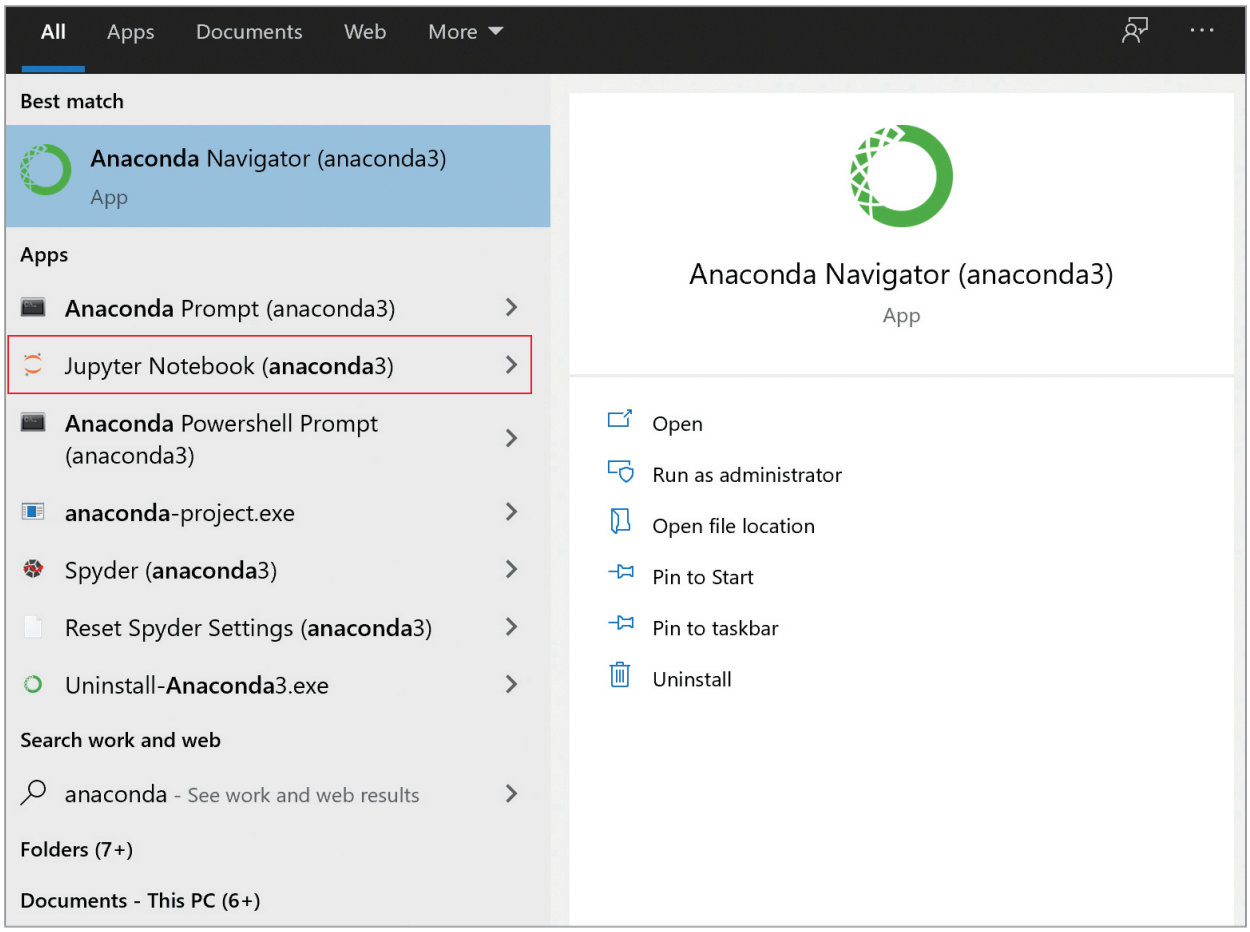


- Anaconda Navigator, Anaconda Prompt, Jupyter Notebook 등 다른 응용 프로그램들도 잘 깔려있는지 확인이 된다면, 설치 완료

3. 주피터 노트북 (Jupyter notebook) 실행

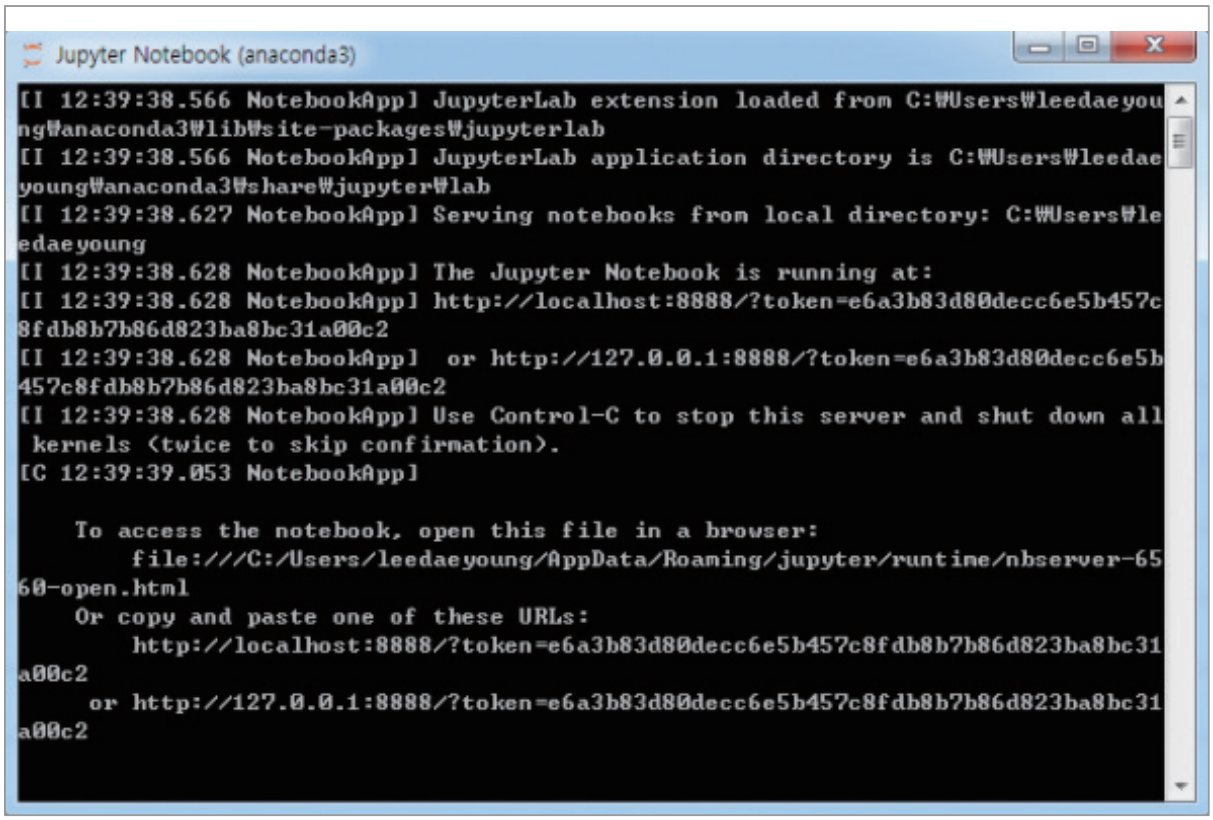
주피터 노트북이란, 실제로 코딩(분석 문장 작성)을 할 수 있는 도구이다. 쉽게 얘기하자면, 문서 도구로 마이크로소프트 사의 'Word' 파일이나, 국내에서는 '한글' 파일 등과 같은 도구라고 생각할 수 있다. 데이터 분석에 다양한 입력, 실행 도구가 있지만, 본 가이드북에서는 주피터 노트북을 활용하는 방법을 공유하기로 한다. [부록2]를 참고하여, Anaconda를 설치하였다면, 주피터 노트북(Jupyter notebook) 설치도 확인한다.

① 윈도우 키( )를 눌러서 시작화면을 연 후, anaconda를 검색.폴더 리스트 중 'Jupyter notebook(anaconda3)'를 실행한다.

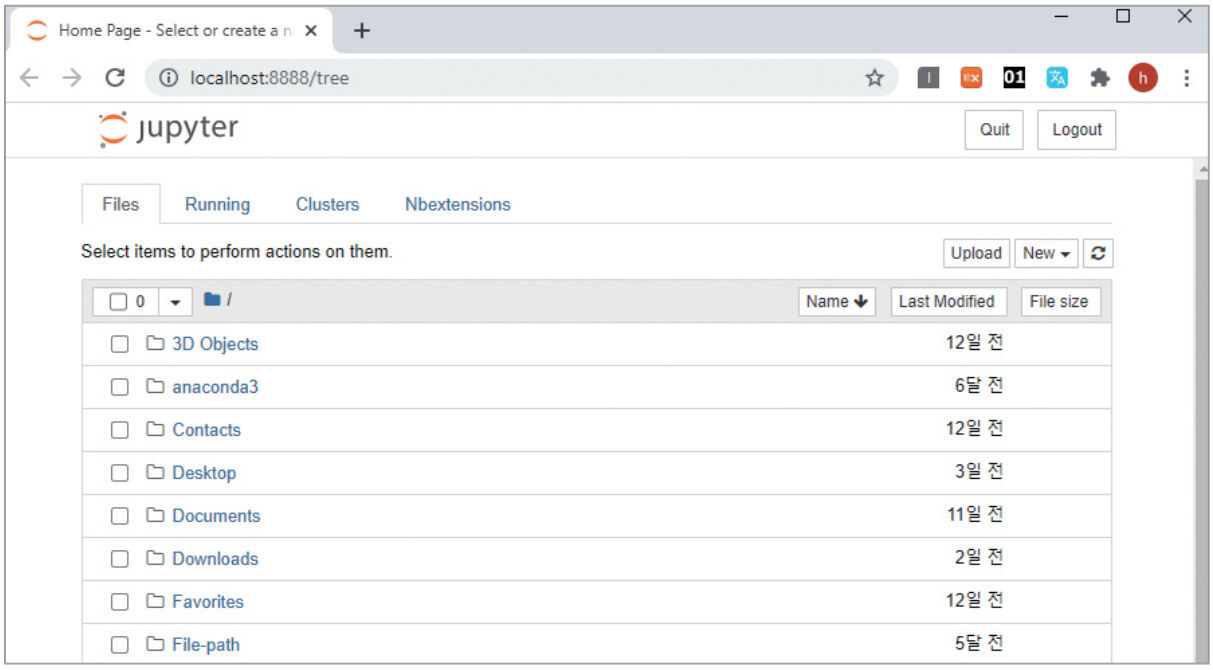


② jupyter notebook을 클릭하게 되면, 2개의 윈도우가 실행.

(1) 검은색 배경의 화면은, 주피터 노트북이 실행되는 환경에 대한 상태를 나타내주는 '상태 표시창' 같은 곳. **분석을 하는 동안 종료하면 안된다.**



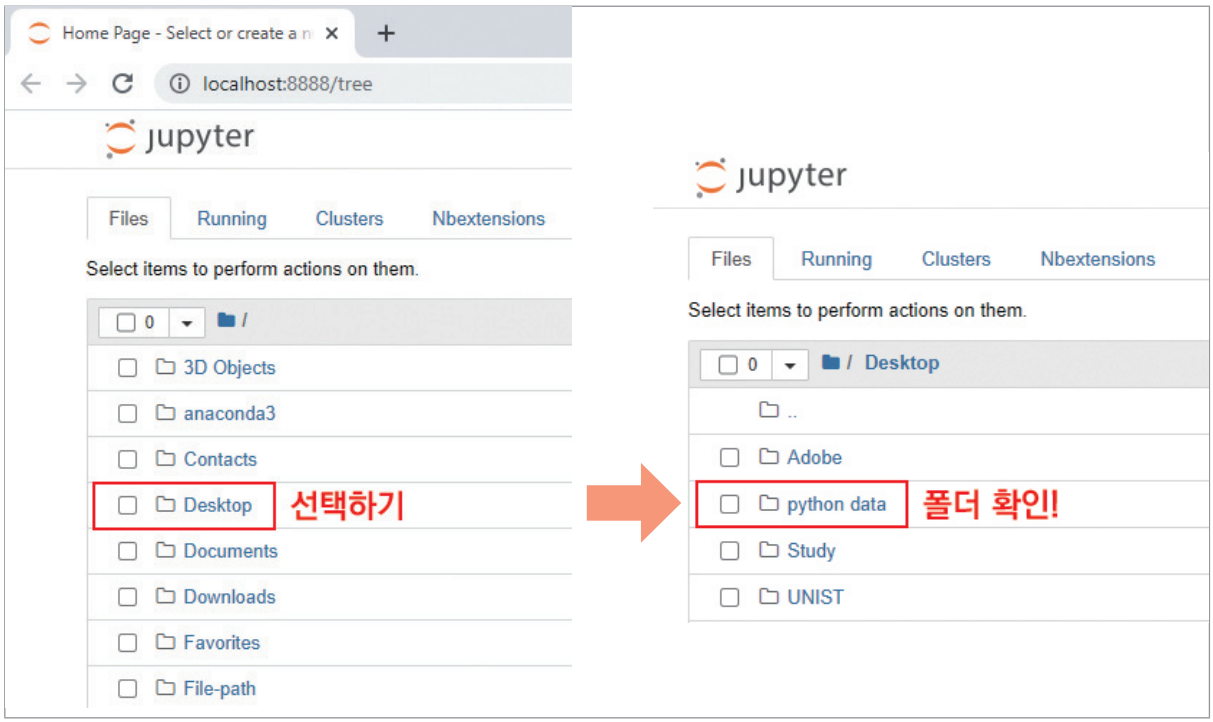
(2) 사용하는 인터넷 프로그램(크롬 / 인터넷 익스플로러)에 주피터 노트북이 열림. (다른 확장 프로그램 사용하고 싶다면, 기본 브라우저를 변경해주어야함). **이 창을 주로 사용.**



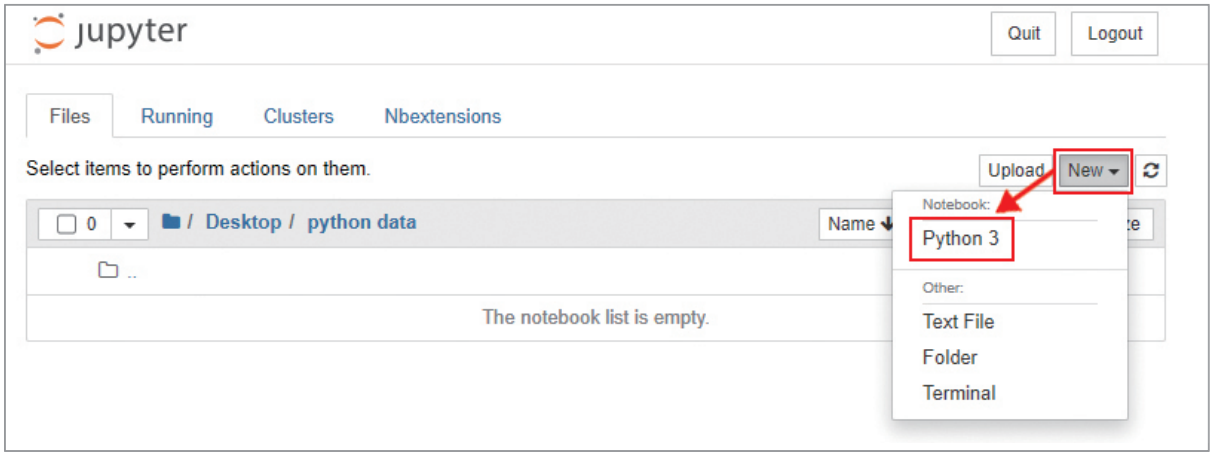
③ 데이터를 저장하고, 불러오고, 분석할 경로의 폴더를 하나 생성

▶ (예) '바탕화면'에 'python data' 폴더 생성 후 (미리 생성), 파이썬 코드 실행 후 저장하기

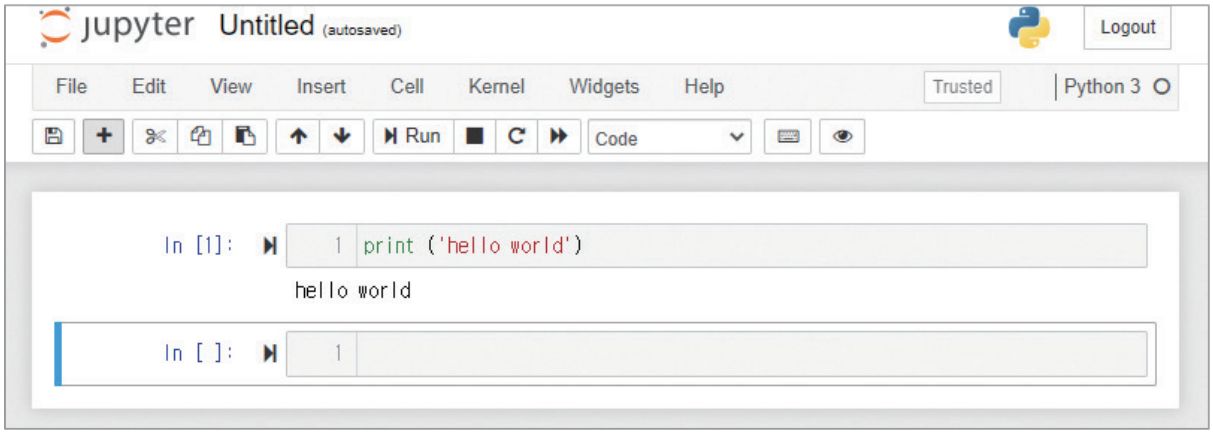
(1) 주피터에서 'python data' 폴더 확인



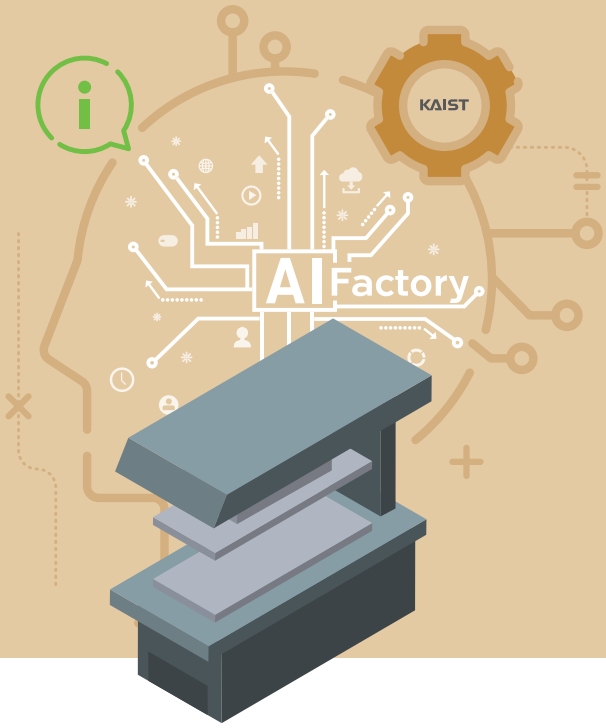
(2) python data에서 파이썬 파일 생성해보기: 오른쪽 상단의 'New'를 누른 후, 'Python 3' 선택



(3) 'hello world' 출력 확인해보기: 보이는 In[] 옆의 회색 창에 print('hello world') 입력 후, shift + Enter 키 눌러서 실행. : 자동으로 저장되며, 확장자는 '(파일이름).ipynb'로 저장



「프레스 AI 데이터셋」
분석실습 가이드북



중소벤처기업부

KOSMO
스마트제조혁신추진단

KAIST

34141 대전광역시 유성구 대학로 291 한국과학기술원(KAIST)
T. (042)350-2114 F. (042)350-2210(2220)