

## ✓ 탐색적 데이터 분석(EDA)

---

## ✓ 탐색적 데이터 분석(Exploratory Data Analysis)란?

- 수집한 데이터가 들어왔을 때, 이를 다양한 각도에서 관찰하고 이해하는 과정
    - 데이터를 분석하기 전에 그래프나 통계적인 방법으로 자료를 직관적으로 바라보는 과정
- 

## ✓ 필요한 이유

- 데이터의 분포 및 값을 검토함으로써 데이터가 표현하는 현상을 더 잘 이해하고, 데이터에 대한 잠재적인 문제를 발견할 수 있음
    - 이를 통해, 본격적인 분석에 들어가기에 앞서 데이터의 수집을 결정할 수 있음
    - 다양한 각도에서 살펴보는 과정을 통해 문제 정의 단계에서 미처 발생하지 못했을 다양한 패턴을 발견하고, 이를 바탕으로 기존의 가설을 수정하거나 새로운 가설을 세울 수 있음
- 

## ✓ EDA의 목적

- 데이터 이해

- 데이터의 구조, 변수 유형, 변수 간의 관계를 파악한다.

- 이상치 및 오류 탐지

- 데이터 중 이상치나 오류가 있는지 확인하여 데이터의 정제를 돕는다.

- 패턴 발견

- 변수 간의 상관 관계나 패턴을 발견하여 가설을 수립하고 향후 분석 방향을 설정한다.


- 중요 변수 식별

- 분석에 중요한 변수를 식별하고, 불필요하거나 중복되는 변수를 제거한다.


---

```
1 #https://velog.io/@alainau331/%ED%83%90%EC%83%89%EC%A0%81-%EB%8D%B0%EC%9D%B4%ED%84%B0-%EB%B6%84%EC%84%9DEDA%EC%9D%B4%EB%9E%80
2 import pandas as pd
3 doc = pd.read_csv("/content/drive/MyDrive/Colab Notebooks/ML/DataSet/covid.csv",encoding='utf-8-sig')
```


```
1 #데이터 일부 확인하기 (head, tail)
2 doc.head()
```




	FIPS	Admin2	Province_State	Country_Region	Last_Update	Lat	Long_	Confirmed	Deaths	Recovered	Active
0	45001.0	Abbeville	South Carolina	US	2020-03-31 23:43:56	34.223334	-82.461707	4	0	0	0
1	22001.0	Acadia	Louisiana	US	2020-03-31 23:43:56	30.295065	-92.414197	39	1	0	38
2	51001.0	Accomack	Virginia	US	2020-03-31 23:43:56	37.767072	-75.632346	7	0	0	7



1 doc.tail()




	FIPS	Admin2	Province_State	Country_Region	Last_Update	Lat	Long_	Confirmed	Deaths	Recovered	Active
2468	82604.0	NaN	Wales	United Kingdom	2020-03-31 23:43:56	52.1307	-3.7837	0	174	0	0
2469	NaN	NaN	NaN	Nauru	2020-03-31 23:43:56	-0.5228	166.9315	0	0	0	0
2470	NaN	NaN	Niue	New Zealand	2020-03-31 23:43:56	-19.0544	-169.8672	0	0	0	0



- 1 #데이터 정보 확인하기 (shape, info)
- 2 doc.shape

 (2473, 12)


```
1 doc.info()
```



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2473 entries, 0 to 2472
Data columns (total 12 columns):
#   Column          Non-Null Count  Dtype
---  -
0   FIPS             2127 non-null   float64
1   Admin2          2172 non-null   object
2   Province_State  2289 non-null   object
3   Country_Region  2473 non-null   object
4   Last_Update     2473 non-null   object
5   Lat             2468 non-null   float64
6   Long_          2468 non-null   float64
7   Confirmed       2473 non-null   int64
8   Deaths         2473 non-null   int64
9   Recovered       2473 non-null   int64
10  Active          2473 non-null   int64
11  Combined_Key    2473 non-null   object
dtypes: float64(3), int64(4), object(5)
memory usage: 232.0+ KB
```

```
1 #데이터의 개별 속성값 확인하기
```

```
2 doc.columns
```



```
Index(['FIPS', 'Admin2', 'Province_State', 'Country_Region', 'Last_Update',
       'Lat', 'Long_', 'Confirmed', 'Deaths', 'Recovered', 'Active',
       'Combined_Key'],
      dtype='object')
```

```
1 doc.describe()
```



	FIPS	Lat	Long_	Confirmed	Deaths	Recovered	Active
<b>count</b>	2127.000000	2468.000000	2468.000000	2473.000000	2473.000000	2473.000000	2473.000000
<b>mean</b>	30087.061119	35.227913	-75.185225	347.735544	18.485645	71.979782	261.462192
<b>std</b>	15675.506681	12.218402	48.372531	3953.486140	337.414333	1458.372755	2801.888341
<b>min</b>	1001.000000	-71.949900	-175.198200	0.000000	0.000000	0.000000	0.000000
<b>25%</b>	18046.000000	33.066412	-94.628178	2.000000	0.000000	0.000000	2.000000
<b>50%</b>	29027.000000	37.451976	-85.997208	6.000000	0.000000	0.000000	6.000000
<b>75%</b>	45024.000000	41.403494	-78.868965	31.000000	1.000000	0.000000	29.000000
<b>max</b>	99999.000000	71.706900	178.065000	105792.000000	12428.000000	63153.000000	77635.000000

1 #속성 간 관계 분석

2 #속성 간 상관관계는 corr을 통해 확인할 수 있다. 이때 디폴트는 피어슨 상관관계수이다. 피어슨 상관관계수는 선형 상관 관계를 조사하며,

3 #+1에 가까울수록 양의 선형 관계

4 #-1에 가까울수록 음의 선형 관계

5 #0에 가까울수록 상관관계가 없음을 의미한다.

6 #최신 pandas에서는 corr()이 자동으로 문자컬럼을 제외해주지 않기 때문에 numeric\_only=True를 추가해야 한다.

7 doc.corr(numeric\_only=True)

8

9 #confirmed, deaths, recovered, active 간의 상관관계가 유의미하다는 것을 확인할 수 있다.



	FIPS	Lat	Long_	Confirmed	Deaths	Recovered	Active
FIPS	1.000000	0.165694	0.168710	0.003070	0.078914	NaN	0.003141
Lat	0.165694	1.000000	-0.484033	0.029011	0.024006	-0.004022	0.036768
Long_	0.168710	-0.484033	1.000000	0.140944	0.096434	0.138355	0.118332
Confirmed	0.003070	0.029011	0.140944	1.000000	0.843065	0.713796	0.936223
Deaths	0.078914	0.024006	0.096434	0.843065	1.000000	0.533003	0.800872
Recovered	NaN	-0.004022	0.138355	0.713796	0.533003	1.000000	0.427627
Active	0.003141	0.036768	0.118332	0.936223	0.800872	0.427627	1.000000

```

1 #데이터 시각화
2 %matplotlib inline
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 plt.figure(figsize=(5,5))
6 #sns.heatmap(data = doc.corr(), annot=True, fmt = '.2f', linewidths=0.5, cmap='Blues') #https://www.infllearn.com/community/questions/1069301/

```



<Figure size 500x500 with 0 Axes>  
<Figure size 500x500 with 0 Axes>

## ✓ <<<참조자료 사이트>>>

1. [AI 훈련용 빅데이터 2026년 고갈...문제점과 대책은](#)
2. ["학습 데이터 부족" AI개발에 닥친 난관... 차세대 모델 개발 지연](#)
3. [AI 데이터 고갈 위기-"2년 후 AI 성장 멈출 수도"...데이터 절벽 '경고'](#)
4. [Undersampling과 Oversampling이란?](#)

5. [불균형 데이터\(imbalanced data\) 처리를 위한 샘플링 기법](#)
6. [샘플링 편향이라는 문제](#)
7. [데이터 품질\(Data Quality\)이란?](#)
8. [데이터에 제값 매기는 데이터 품질 관리](#)
9. [과대적합\(overfitting\) 및 과소적합\(underfitting\) 개념\(+Early Stopping\)](#)
10. [EDA\(Exploratory Data Analysis\) 탐색적 데이터 분석](#)
11. [탐색적 데이터 분석\(EDA: Exploratory Data Analysis\)이란? 데이터 사이언스의 필수 요소 이해하기](#)