**0. Pandas를 이용하여 데이터 파일 읽기**

pd.read\_csv(file\_path)

**1. 데이터 구조 보기**

|  |  |
| --- | --- |
| df.head() | 데이터의 앞부분 5개 보기 |
| df.shape | (Row, Col) 개수 출력 |
| df.info() | 각 컬럼별 데이터 타입 및 Null값 등 요약 정보 |

**2-1 수치형 데이터 분석**

|  |  |
| --- | --- |
| num\_cols = [col for col in trn.columns if trn[col].dtype in ['int64', 'float64']] | 수치형 데이터 Column 분리 |
| pd.options.display.float\_format = '{:.4g}'.format  pd.options.display.float\_format = '{:.2f}'.format | 소수자리를 e 출력 방식  소수자리 precision 방식 |
| df[num\_cols].describe() | Count, mean, std, min, 25%, 50%, 75%, max로  수치적 데이터 분포 요약 |
| df[num\_cols].hist(bins=50, figsize=(20,15)) | 수치형 데이터의 분포 시각화 |
| f, ax = plt.subplots(figsize=(20,15))  sns.countplot(data=trn[col], x=col, alpha=0.5)  plt.show() | 수치형 데이터의 분포 시각화2 (범주형도 가능) |

**2-2 범주형 데이터 분석**

|  |  |
| --- | --- |
| cat\_cols = [col for col in trn.columns if trn[col].dtype in ['object']] | 범주형 데이터 Column 분리 |
| for col in cat\_cols:  uniq = np.unique(trn[col].astype(str))  print('-' \* 50)  print('# col {}, n\_uniq {}, uniq {}'.format(col, len(uniq), uniq)) | 범주형 데이터의 unique 값 보기 |
| df[cat\_cols].describe() | Count, unique, top, freq로  범주형 데이터 분포 요약 |
| for col in cat\_cols:  print(trn[col].value\_counts()) | 범주형 데이터의 unique 값의 분포 요약 |
| f, ax = plt.subplots(figsize=(20,15))  sns.countplot(data=trn[col], x=col, alpha=0.5)  plt.show() | 데이터 분포의 시각화 |

**2-3 수치형 데이터의 상관관계 분석**

|  |  |
| --- | --- |
| df.plot(kind=”scatter”, x=col1, y=col2, alpha=0.1) | 두 수치 데이터의 산점도 분포 |
| corr\_matrix=df.corr() | 각 변수간 표준 상관관계 계수 : 피어슨 r값  (standard correlation coefficient) |
| corr\_maxtrix[col] | col과 다른 값들의 상관관계 계수 [-1,1] |
| From pandas.plotting import scatter\_matrix  attributes = list of columns  scatter\_matrix(df[attributes], figsize=(12,8)) | 각 컬럼 간 산점도 분포 시각화 |
|  |  |

**3. 데이터 정제**

Raw 데이터 특성상 Null값이 존재하거나, 학습처리에 편한 데이터로 변환하거나,

필요없는 데이터들을 삭제하거나.. 등의 정제가 필요하다.

|  |  |
| --- | --- |
| Series.replace(‘value’, -99, inplace=True) | Value 값을 가진 것을 -99로 변경한다. |
| Series.astype(np.int8) | 데이터 타입 변경 |
| Series.fillna(-99, inplace=True) | Nan 또는 NA 값을 -99로 변경한다. |
| Series.factorize(na\_sentinel=-99, sort = True) | - Categorical Data를 유니크 값을 0으로 시작하는 정수형으로 변경한다.  - na\_sentinel는 NA값을 변경할 값 설정  - sort가 있는 경우, 유니크 값을 정렬한 후 0~ 부여한다.  - 반환 값은 라벨 리스트를 반환한다. |
| Series.map(func) | 각 Value에 func적용 |
|  |  |