

Pandas - przydatne metody

sobota, 26 września 2020

14:29

- `pd.read_csv(csv_path)` - stworzenie pandas dataframe
- `Data_frame.head()` - pobranie pierwszy 5 wierszy dataframeu pandas
- `Data_frame.info()` - krótkie informacje o atrybutach (kolumnach) dataframeu
- `Data_frame['atrybut'].value_count()` - informacje o liczbie poszczególnych wartości danego atrybutu (Dobre dla atrybutów kategorialnych)
- `Data_frame.describe()` - generuje podsumowanie atrybutów numerycznych
- `Data_frame.hist()` - rysuje histogram z użyciem matplotlib
- `Data_frame.iloc[tabela_liczb]` - wybiera elementy o indeksach zawartych w tablicy
- `Data_frame.loc[tabela_wartości_bool]` - wybiera elementy o indeksach którym wartość bool odpowiada true
- `Data_frame.loc[~tabela_wartości_bool]` - przeciwnie do powyższego
- `Data_frame['atrybut'].where(data_frame[atrybut], warunek, wartość do zamiany, inplace=True)` - zamienia wartości które spełniają warunek
- `Data_frame.plot()` - pozwala na stworzenie wykresu z danych. Parametr **alpha** pomaga przy ocenie zagęszczenia, parametr **s** pozwala określić promień punktów w stosunku do zadanego atrybutu. Parametr **c** określa jaki atrybut jest brany pod uwagę w kolorowaniu wykresu a **cmap** określa mapę kolorów (można bez cmap ale wtedy wykres jest w odcieniach szarości)
- `Data_frame.corr()` - wylicza współczynnik korelacji liniowej między każdą parą wartości

- `Scatter_matrix(dataset)` - wyświetla fajną macierz korelacji
- `Data_frame.drop(Atrybut, axis=1)` - usuwa atrybut ze zbioru
- `Data_frame.dropna(Atrybut)` - usuwa elementy dla których Atrybut jest pusty
- `Data_frame[Atrybut].median()` lub `Data_frame.median()` - oblicz mediane wartości dla konkretnego atrybutu lub dla wszystkich atrybutów (Jest to wtedy tablica)
- `Data_frame[x:y]` - dostajemy macierz z wartościami w wierszach od x do y (jeśli dodamy przecinek `[x:y, z]` otrzymamy tylko wartości w wierszach z 'z' kolumny)