به نام هستی بخش

گزارش تمرین سری اول

درس شناسایی الگو

نویسنده: حمیدرضا ابوئی

شماره دانشجویی: ۴۰۲۶۱۷۵۰۹

استاد: دکتر دلیری

آذر ۱۴۰۲

۱ -

```python
# Gathering every feature's data type abd other informations
data.info()
```

[4] ✓ 0.0s

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
```

```python
data.describe().T
```

✓ 0.1s

	count	mean	std	min	25%	50%	75%	max
longitude	20640.0	-119.569704	2.003532	-124.3500	-121.8000	-118.4900	-118.01000	-114.3100
latitude	20640.0	35.631861	2.135952	32.5400	33.9300	34.2600	37.71000	41.9500
housing_median_age	20640.0	28.639486	12.585558	1.0000	18.0000	29.0000	37.00000	52.0000
total_rooms	20640.0	2635.763081	2181.615252	2.0000	1447.7500	2127.0000	3148.00000	39320.0000
total_bedrooms	20433.0	537.870553	421.385070	1.0000	296.0000	435.0000	647.00000	6445.0000
population	20640.0	1425.476744	1132.462122	3.0000	787.0000	1166.0000	1725.00000	35682.0000
households	20640.0	499.539680	382.329753	1.0000	280.0000	409.0000	605.00000	6082.0000
median_income	20640.0	3.870671	1.899822	0.4999	2.5634	3.5348	4.74325	15.0001
median_house_value	20640.0	206855.816909	115395.615874	14999.0000	119600.0000	179700.0000	264725.00000	500001.0000

با توجه به این که ۰ جزو دامنه اعداد نیست، احتمال این که مقادیر از دست رفته با ۰ جایگزین شده باشند نمی‌باشد.

```python
data.isnull().sum()
```

[9] ✓ 0.0s

```
longitude                 0
latitude                  0
housing_median_age        0
total_rooms               0
total_bedrooms          207
population                0
households                0
median_income             0
median_house_value        0
ocean_proximity           0
dtype: int64
```

۲۰۷ مقدار از دست رفته است و نوع متغیر، float 64 می‌باشد.(با توجه به تصویر اول)

Missign value management

Feature total_rooms has missing values. the data type of this feature is float64. The total number of missing values is 207. The total number of entries is 20640. That means, missing values take around 1% of our data + we have plenty of data, so I don't mind to remove all the rows containing missing value. (removeing that row) However in this exercise, we replace the missing data with median

```python
# Replacing missing values with median
filled_data = data.fillna(data.median())
```

[17] ✓ 0.0s Python

```python
filled_data.info()
```

[19] ✓ 0.0s

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20640 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
```

مشاهده می‌شود که با این روش (جایگزینی) تعداد داده‌ها تغییری نمی‌کند.

۲ -

در این سوال برای مشخص کردن داده‌های پرت، از روش ۳ فاصله انحراف استاندارد استفاده می‌کنیم. بدین صورت که از میانگین، به اندازه سه برابر انحراف استاندارد فاصله می‌گیریم و اگر داده‌ای خارج از این بازه باشد، به عنوان داده پرت علامت خورده و حذف می‌شود. همچنین می‌توان از روش IQR در باکس پلات استفاده کرد که از Q1 به اندازه ۱.۵ برابر IQR فاصله میگیریم و هرچه داده کمتر بود حذف می‌شود و همچنین هرچه داده بیشتر از ۱.۵ برابر IQR بیشتر از Q3 باشد علامت خورده و حذف می‌شود.

می‌توان تعداد داده‌های پرت را در هر ویژگی در تصویر زیر مشاهده کرد:

```python
for x in numeric_col:
    # q75,q25 = np.percentile(heart.loc[:,x],[75,25])  # a box plot of the quartile range and min/max values method
    # IQR = q75-q25
    # max = q75+(1.5*IQR)
    # min = q25-(1.5*IQR)

    max_data = filled_data[x].mean() + 3*filled_data[x].std()       # Z-score method
    min_data = filled_data[x].mean() - 3*filled_data[x].std()

    filled_data.loc[filled_data[x] < min_data,x] = np.nan   #filling the outliers values with 'nan'
    filled_data.loc[filled_data[x] > max_data,x] = np.nan   #filling the outliers values with 'nan'
```

[31] ✓ 0.0s

```python
filled_data.isna().sum()
```

[32] ✓ 0.0s

```
longitude               0
latitude                0
housing_median_age      0
total_rooms           373
total_bedrooms        375
population            342
households            363
median_income         345
median_house_value      0
ocean_proximity         0
dtype: int64
```

۳ -

جهت نرمال‌سازی ابتدا ویژگی هدف را از دادگان جدا می‌کنیم. (در اینجا، فرض بر قیمت خانه میباشد)

سپس داده‌های غیر عددی جدا می‌شوند.

Standardization (normalization)

Standard Scaler It standardize features by removing the mean and scaling to unit variance The standard score of a sample x is calculated as:

z = (x - u) / s

First we want to seperate our target column. We consider median_house_value as our target column.
Second, we seperate our not numerical data.
Then we use standart scaler formula to standardize our data features.

```
# remove target
target_data = removed_outliers['median_house_value']
features = removed_outliers.drop(columns='median_house_value')
# remove object data
objective_features = features['ocean_proximity']
numeric_features = features.drop(columns='ocean_proximity')
```

در ادامه می‌توان پیاده سازی فرمول نرمالیزاسیون و نتیجه آن را مشاهده کرد:

```
# Using standard scaler formula
normalized_features = (numeric_features - numeric_features.mean())/numeric_features.std()
normalized_features
```

[83] ✓ 0.0s

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income
0	-1.320375	1.038968	0.964812	-1.053373	-1.258591	-1.280698	-1.250024	2.871928
1	-1.315382	1.029643	-0.645900	3.281869	2.118273	1.409283	2.531766	2.857069
2	-1.325367	1.024980	1.850704	-0.644178	-1.047753	-1.055562	-1.059440	2.205302
3	-1.330360	1.024980	1.850704	-0.778717	-0.892217	-0.975341	-0.902488	1.197497
4	-1.330360	1.024980	1.850704	-0.532642	-0.736680	-0.966284	-0.753010	0.075696
...	...	...	...	...	...	...	...	...
20635	-0.751237	1.785020	-0.323758	-0.506153	-0.411782	-0.603997	-0.487687	-1.351388
20636	-0.811146	1.789682	-0.887507	-1.180942	-1.186007	-1.236706	-1.294867	-0.729274
20637	-0.816138	1.761705	-0.968042	-0.095563	-0.028126	-0.394388	-0.102782	-1.264173
20638	-0.866063	1.761705	-0.887507	-0.370219	-0.290810	-0.738561	-0.416685	-1.159790
20639	-0.826123	1.733729	-1.048578	0.274595	0.424657	0.097287	0.259702	-0.834281

19746 rows × 8 columns

و در تصویر زیر می‌توان میانگین و انحراف معیار ویژگی‌های نرمال‌سازی شده را مشاهده کرد.

```
normalized_features.describe().T
```
✓ 0.1s

	count	mean	std	min	25%	50%	75%	max
longitude	19746.0	-6.251884e-15	1.0	-2.378772	-1.105700	0.536812	0.781442	2.633636
latitude	19746.0	1.148614e-15	1.0	-1.450978	-0.802846	-0.644310	0.969026	2.936736
housing_median_age	19746.0	1.763223e-17	1.0	-2.256612	-0.806971	-0.001615	0.642670	1.850704
total_rooms	19746.0	6.621081e-17	1.0	-1.665424	-0.669970	-0.215464	0.443118	4.731829
total_bedrooms	19746.0	6.387184e-17	1.0	-1.697549	-0.683971	-0.214770	0.448851	4.451317
population	19746.0	-1.370996e-16	1.0	-1.693446	-0.685512	-0.208069	0.466043	4.536596
households	19746.0	-6.162283e-17	1.0	-1.713405	-0.682008	-0.211153	0.465234	4.422661
median_income	19746.0	1.252248e-16	1.0	-2.013394	-0.738701	-0.145899	0.576508	3.640376

نرمال‌سازی داده‌های توزیع شده بر اساس چند مقیاس را به یک مقیاس تبدیل می‌کند. پس از نرمال‌سازی، همه‌ی متغیرها تأثیر مشابه‌ی بر مدل مورد استفاده دارند و باعث بهبود پایداری و عملکرد الگوریتم یادگیری می‌شوند.

با تشکر.