Discretization

Many times we need to convert continuous attributes into multiple intervals, so we can reduce the data, or remove some variance. This process is called discretization.

Setup

```
import numpy as np
import pandas as pd
```

```
df = pd.read_csv('/content/sample_data/california_housing_train.csv')
df.head()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	popula
0	-114.31	34.19	15.0	5612.0	1283.0	1
1	-114.47	34.40	19.0	7650.0	1901.0	1
2	-114.56	33.69	17.0	720.0	174.0	
3	-114.57	33.64	14.0	1501.0	337.0	
4	-114.57	33.57	20.0	1454.0	326.0	

```
df.describe()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedroom
count	17000.000000	17000.000000	17000.000000	17000.000000	17000.00000
mean	-119.562108	35.625225	28.589353	2643.664412	539.41082
std	2.005166	2.137340	12.586937	2179.947071	421.49945
min	-124.350000	32.540000	1.000000	2.000000	1.00000
25%	-121.790000	33.930000	18.000000	1462.000000	297.00000
50%	-118.490000	34.250000	29.000000	2127.000000	434.00000
75%	-118.000000	37.720000	37.000000	3151.250000	648.25000
max	-114.310000	41.950000	52.000000	37937.000000	6445.00000

Discretize population

```
df['popular'] = np.select([df['population'] <1429.573941, df['population'] >= 1429
df['popular']
```

```
0        not popular
1        not popular
2        not popular
3        not popular
4        not popular
           ...
16995    not popular
16996    not popular
16997    not popular
16998    not popular
16999    not popular
Name: popular, Length: 17000, dtype: object
```

```
df['popular'].value_counts()
```

```
not popular    10862
popular         6138
Name: popular, dtype: int64
```

Discretize rooms

```
conditions = [
    (df['total_rooms'] < 1462) & (df['total_bedrooms'] < 297),
    (df['total_rooms'] > 3151) & (df['total_bedrooms'] > 648),
    (df['total_rooms'] < 2127) & (df['total_bedrooms'] > 434),
    (df['total_rooms'] > 2127) & (df['total_bedrooms'] < 434),
]

values = ['LL', 'HH', 'LH', 'HL']
df['rooms'] = np.select(conditions, values)
df['rooms']
```

```
0        HH
1        HH
2        LL
3         0
4         0
         ..
16995    HL
16996     0
16997     0
16998     0
16999     0
Name: rooms, Length: 17000, dtype: object
```

```
df['rooms'].value_counts()
```

```
0     7970
LL    3424
HH    3394
HL    1110
LH    1102
Name: rooms, dtype: int64
```

Discretize house value

```
def house_value(value):
    if value < 119400:
        return "Low"
    elif value > 265000:
        return "High"
    else:
        return "Medium"
```

```
df['house_value_category'] = df['median_house_value'].apply(house_value)
df['house_value_category']
```

```
0        Low
1        Low
2        Low
3        Low
4        Low
        ...
16995    Low
16996    Low
16997    Low
16998    Low
16999    Low
Name: house_value_category, Length: 17000, dtype: object
```

```
df['house_value_category'].value_counts()
```

```
Medium    8510
High      4247
Low       4243
Name: house_value_category, dtype: int64
```

Colab paid products - Cancel contracts here