Московский государственный технический университет им. Н.Э. Баумана
Кафедра «Системы обработки информации и управления»

Лабораторная работа №1
по дисциплине
«Технологии машинного обучения»
на тему
«Разведочный анализ данных. Исследование и визуализация данных»

Выполнил:
студент группы ИУ5-63Б
Кривцов Н. А.

_____

Москва — 2020 г.

0.1. # Лабораторная работа №1. Разведочный анализ данных. Исследование и визуализация данных.

0.2. 1) Текстовое описание набора данных

Используется набор данных о жилье Калифорнии. Датасет доступен в библиотеке scikit-learn.
Каждая запись представляет собой сведения о некотором территориальном блоке, определенном переписью населения США 1990 года.

Признаки датасета: * MedInc - медианный годовой доход населения блока (в сотнях тысяч долларов). * HouseAge - средний возраст домов в блоке. * AveRooms - среднее число комнат в доме. * AveBedrooms - среднее число *спальных* комнат в доме. * Population - население блока. * AveOccup - средняя число жильцов в доме. * Latitude - географическая широта блока. * Longitude - географическая долгота блока. * MedValue - медианная стоимость домов в блоке (в сотнях тысяч долларов). **Целевой признак.** — ### Импорт библиотек

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn.datasets
%matplotlib inline
sns.set(style="darkgrid")
```

0.2.1. Загрузка набора данных

Метод `sklearn.datasets.fetch_california_housing()` возвращает объект со следующими полями: * `data` - NumPy-матрица значений исходных признаков. * `target` - NumPy-вектор значений целевого признака. * `feature_names` - упорядоченный массив названий признаков датасета (а именно - поля `data`).

Для удобства анализа исходные и целевые признаки объединяются в один `DataFrame`.

[2]:
```
cal_housing = sklearn.datasets.fetch_california_housing()
data = pd.DataFrame(cal_housing.data, columns=cal_housing.feature_names)
target = pd.DataFrame(cal_housing.target, columns=["MedValue"])
data = data.join(target)
```

0.3. 2) Основные характеристики датасета

[3]:
```
# Типы признаков датасета
data.dtypes
```

[3]:
```
MedInc        float64
HouseAge      float64
AveRooms      float64
AveBedrms     float64
Population    float64
AveOccup      float64
Latitude      float64
Longitude     float64
MedValue      float64
```

```
dtype: object
```

[4]:
```
# Первые пять записей в датасете
data.head()
```

[4]:
```
   MedInc  HouseAge  AveRooms  AveBedrms  Population  AveOccup  Latitude  \
0  8.3252      41.0  6.984127   1.023810       322.0  2.555556     37.88
1  8.3014      21.0  6.238137   0.971880      2401.0  2.109842     37.86
2  7.2574      52.0  8.288136   1.073446       496.0  2.802260     37.85
3  5.6431      52.0  5.817352   1.073059       558.0  2.547945     37.85
4  3.8462      52.0  6.281853   1.081081       565.0  2.181467     37.85

   Longitude  MedValue
0    -122.23     4.526
1    -122.22     3.585
2    -122.24     3.521
3    -122.25     3.413
4    -122.25     3.422
```

[5]:
```
# Статистические характеристики признаков датасета
data.describe()
```

[5]:
```
             MedInc      HouseAge      AveRooms     AveBedrms  Population  \
count  20640.000000  20640.000000  20640.000000  20640.000000  20640.000000
mean       3.870671     28.639486      5.429000      1.096675   1425.476744
std        1.899822     12.585558      2.474173      0.473911   1132.462122
min        0.499900      1.000000      0.846154      0.333333      3.000000
25%        2.563400     18.000000      4.440716      1.006079    787.000000
50%        3.534800     29.000000      5.229129      1.048780   1166.000000
75%        4.743250     37.000000      6.052381      1.099526   1725.000000
max       15.000100     52.000000    141.909091     34.066667  35682.000000

            AveOccup      Latitude     Longitude      MedValue
count  20640.000000  20640.000000  20640.000000  20640.000000
mean       3.070655     35.631861   -119.569704      2.068558
std       10.386050      2.135952      2.003532      1.153956
min        0.692308     32.540000   -124.350000      0.149990
25%        2.429741     33.930000   -121.800000      1.196000
50%        2.818116     34.260000   -118.490000      1.797000
75%        3.282261     37.710000   -118.010000      2.647250
```

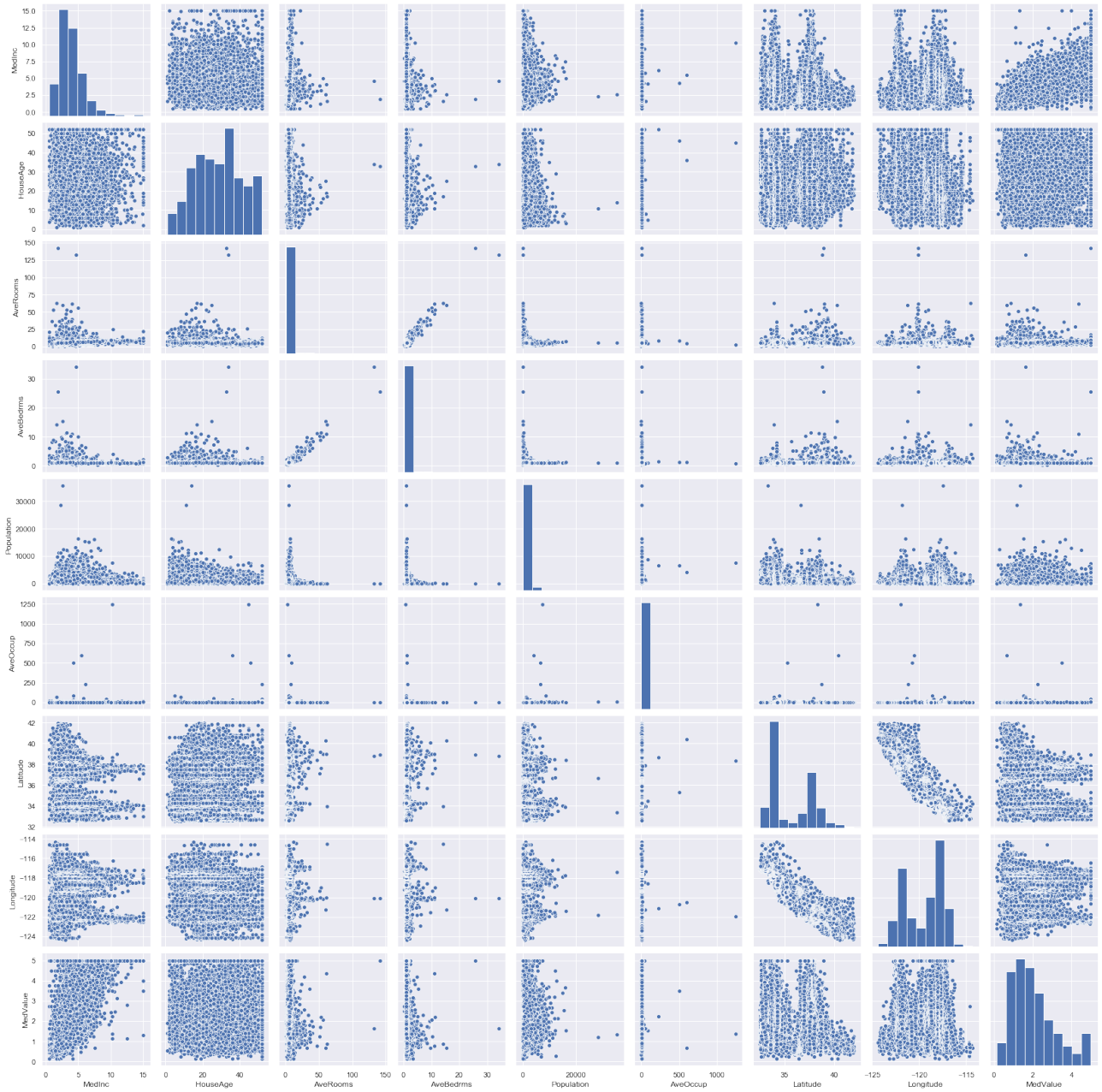
```
max      1243.333333      41.950000    -114.310000       5.000010
```

0.4. 3) Визуальное исследование датасета

0.4.1. Диаграммы рассеяния и гистограммы

[6]:
```
sns.pairplot(data)
```

[6]:
```
<seaborn.axisgrid.PairGrid at 0x13021130>
```



Между признаками `AveRooms` и `AveBedrms` наблюдается линейная зависимость (что очевидно из “физического смысла” самих признаков).

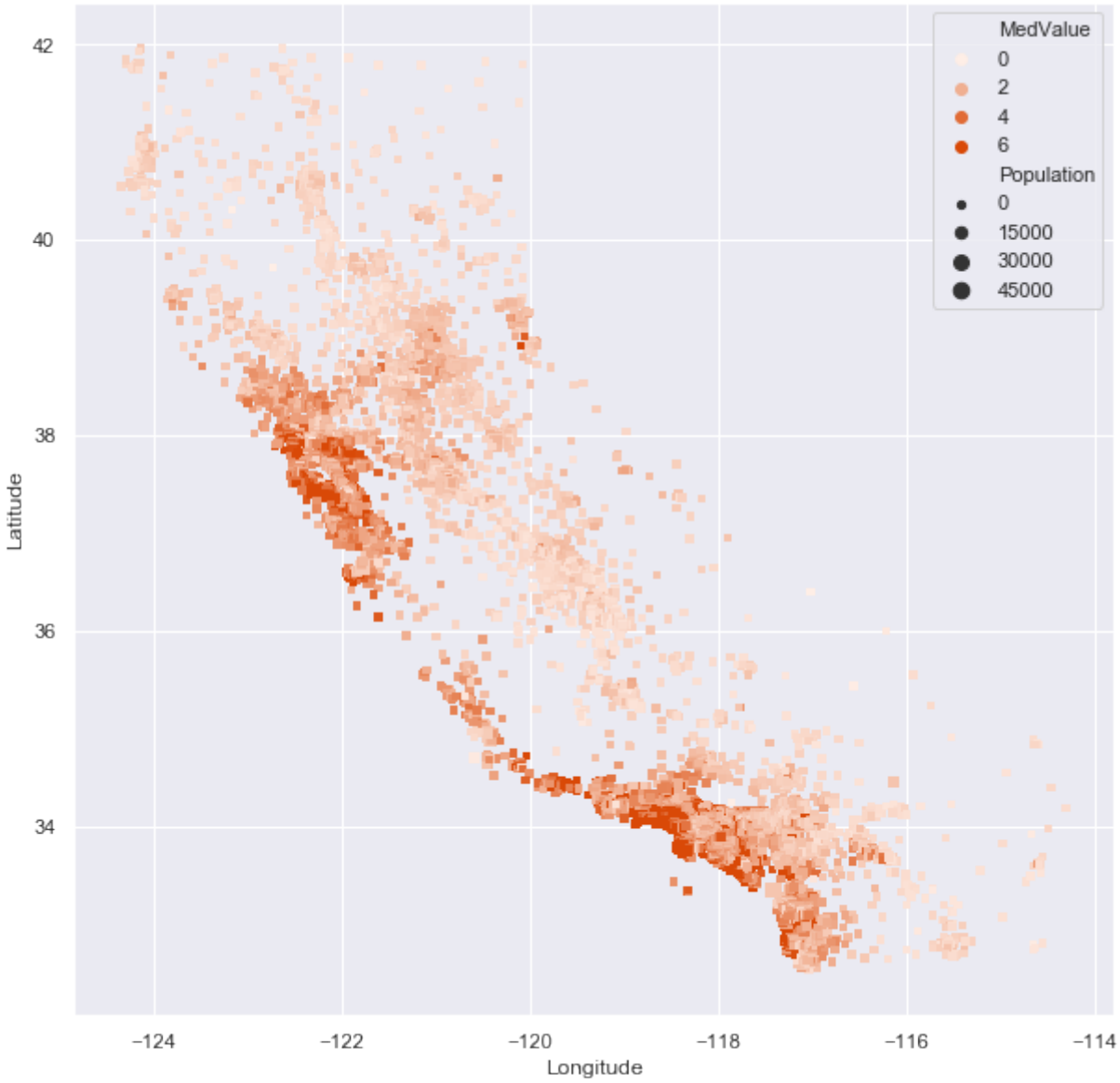
Судя по гистограмме, целевой признак имеет “почти” нормальное распределение - исключением является второй пик графика около значения `5.0`.

0.4.2. Зависимость стоимости от географических координат

По оси абсцисс откладывается долгота, по оси ординат - широта блока. Цвет точки характеризуется медианной стоимостью домов - более насыщенному цвету соотвествует более высокая стоимость.

[7]:
```
colors = sns.light_palette((217/255, 73/255, 7/255), as_cmap=True)
fig, ax = plt.subplots(figsize=(10, 10))
sns.scatterplot(x="Longitude", y='Latitude', data=data, hue="MedValue",
 ↪ax=ax, palette=colors, linewidth=0, marker='s', size='Population')
```

[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x15979fd0>
```

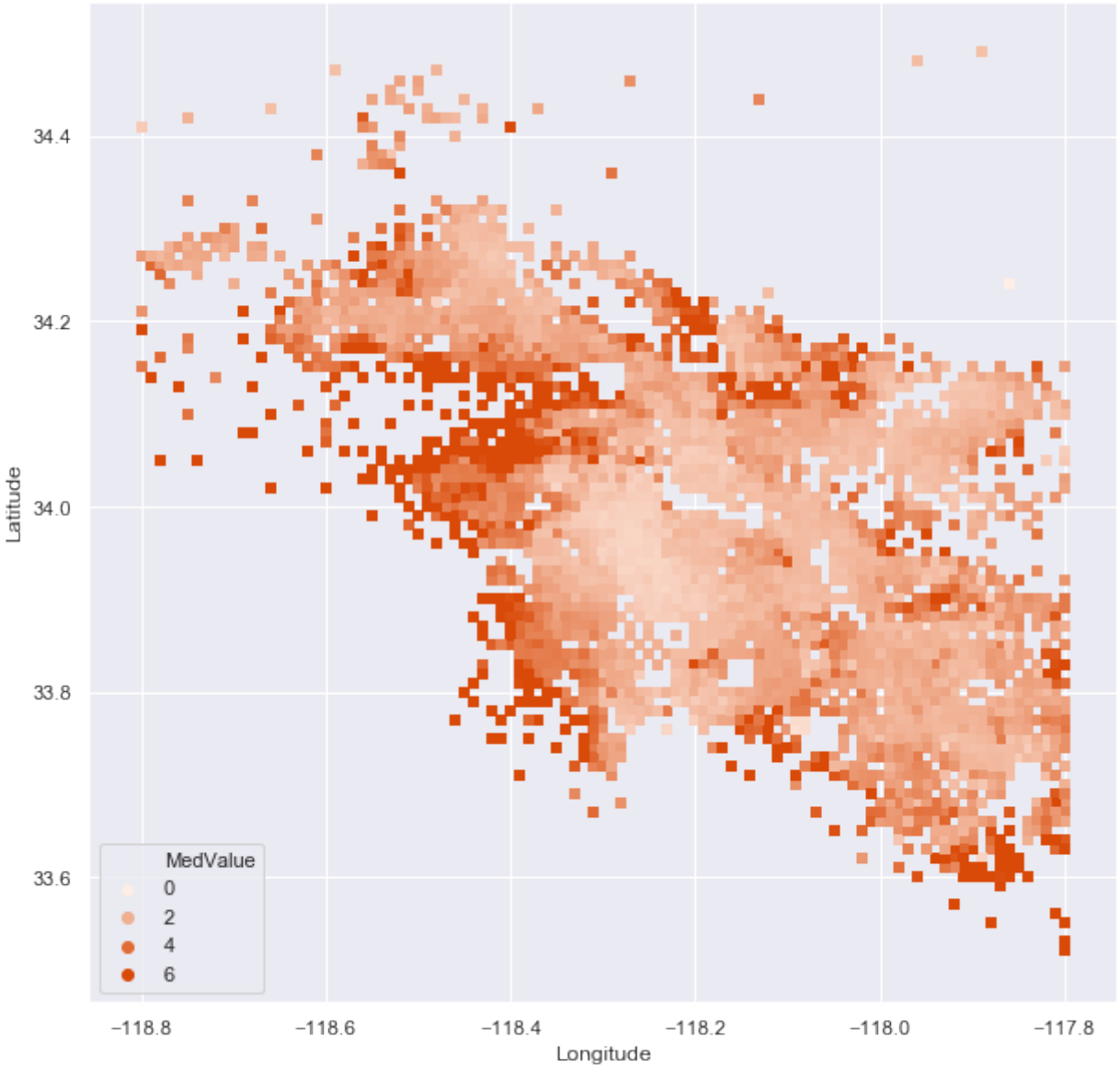


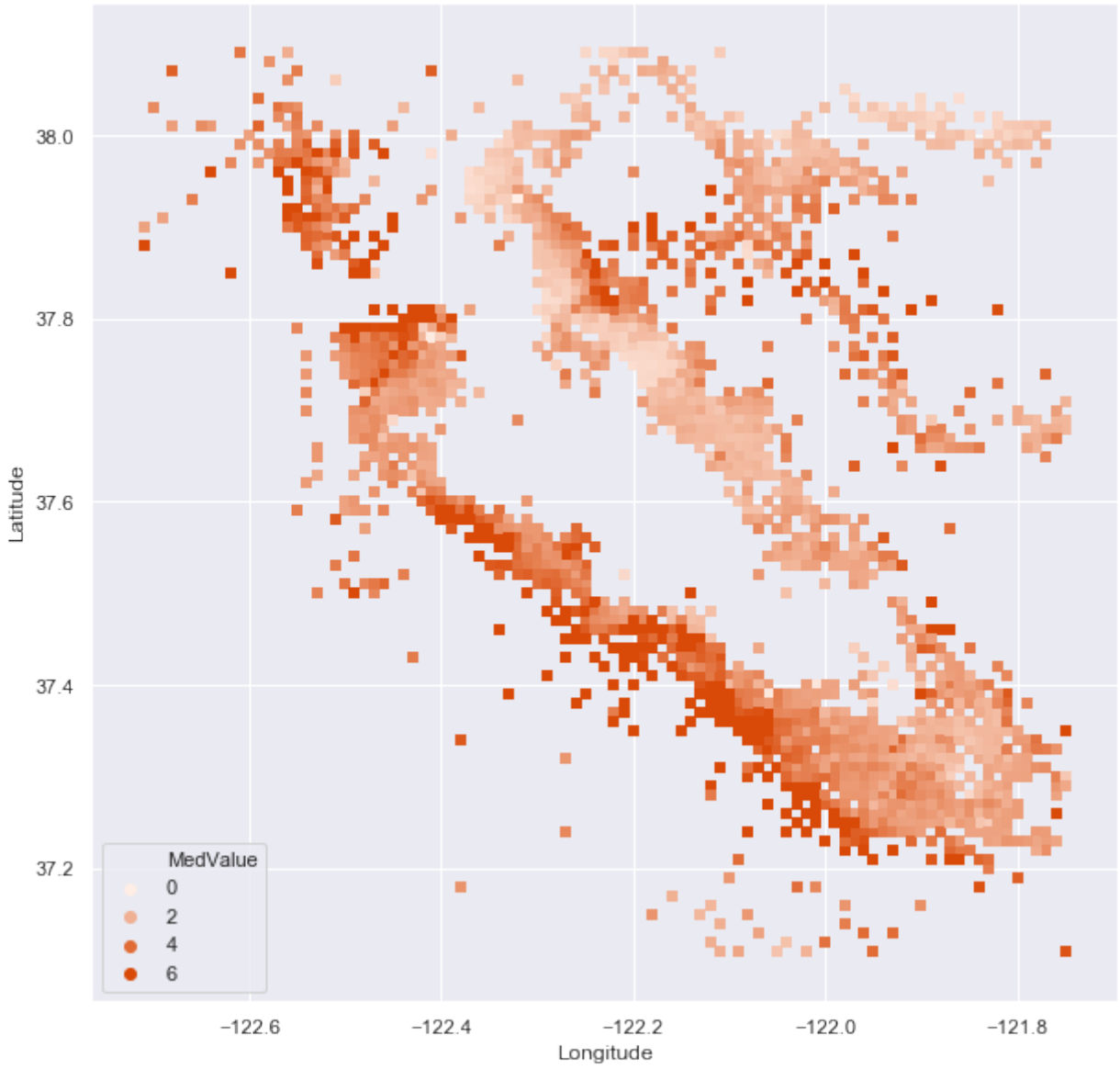
По диаграмме можно выделить следующие закономерности: * Дома на тихоокеанском побережье, в общем случае, оказываются дороже, нежели дома вдали от берега. * Выделяются два района с высокой стоимостью домов и крайне плотным расположением исследуемых блоков.

В самом деле: в Калифорнии находятся две крупных агломерации - округ Лос-Анджелес и бухта Сан-Франциско, что объясняет группировку точек на диаграмме.
Рассмотрим оба региона в крупном масштабе.

[8]:
```
city = data.loc[(data['Longitude'] >= -118.8) & (data['Longitude'] <=
 ↪-117.8) & (data['Latitude'] < 34.5) & (data['Latitude'] > 33.5)]
fig, ax = plt.subplots(figsize=(10, 10))
sns.scatterplot(x="Longitude", y='Latitude', data=city, hue="MedValue",
 ↪ax=ax, palette=colors, linewidth=0, marker='s')
city = data.loc[(data['Longitude'] >= -122.75) & (data['Longitude'] <=
 ↪-121.75) & (data['Latitude'] < 38.1) & (data['Latitude'] > 37.1)]
fig, ax = plt.subplots(figsize=(10, 10))
sns.scatterplot(x="Longitude", y='Latitude', data=city, hue="MedValue",
 ↪ax=ax, palette=colors, linewidth=0, marker='s')
```

[8]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1014190>
```



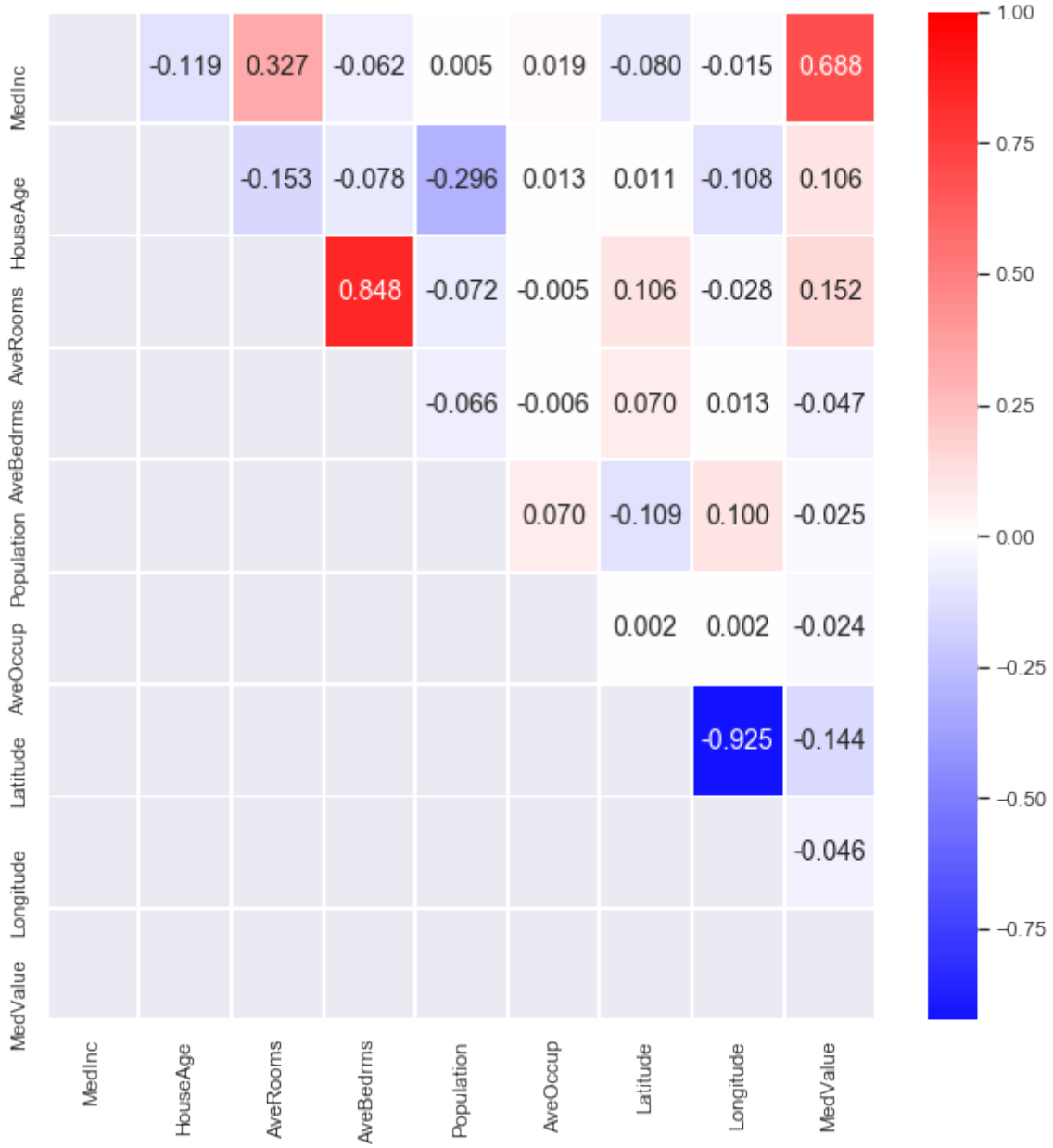


0.5. 4) Корреляция признаков

0.5.1. Тепловая карта

[14]:
```
mask = np.zeros_like(data.corr(), dtype=np.bool)
mask[np.tril_indices_from(mask)] = True
fig, ax = plt.subplots(figsize=(10, 10))
sns.heatmap(data.corr(), cmap='bwr', annot=True, fmt='.3f',
 ↪linewidths=1, center=0, annot_kws={'size': 14}, mask=mask)
```

[14]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x19213050>
```



Как было отмечено раньше, между признаками `AveRooms` и `AveBedrms` наблюдается значительная корреляция, которая так же заметна между координатами широты и долготы.

0.5.2. Признаки, коррелирующие с целевым

[11]:
```
data.corr()["MedValue"].sort_values(ascending=False)
```

[11]:
```
MedValue      1.000000
MedInc        0.688075
AveRooms      0.151948
HouseAge      0.105623
AveOccup     -0.023737
Population   -0.024650
Longitude    -0.045967
AveBedrms    -0.046701
```

```
Latitude     -0.144160
Name: MedValue, dtype: float64
```