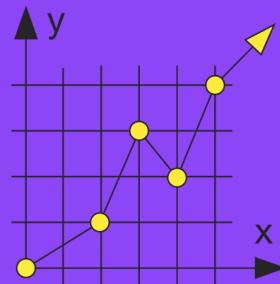


Визуализация данных с Matplotlib и Seaborn

Урок 4





Что будет на уроке сегодня

- 📌 Виды графиков
- 📌 Интерпретация графиков
- 📌 Визуальный анализ данных
- 📌 Анализ геоданных





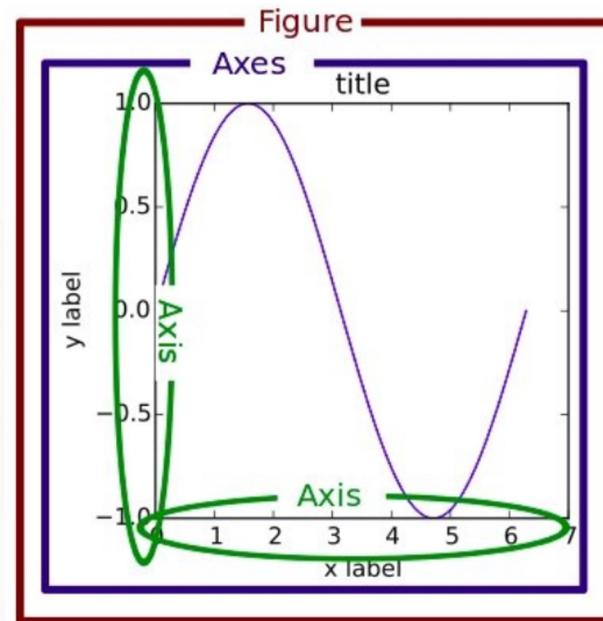
Устройство графиков



Устройство графика

Есть объект `Figure` (фигура), в которой можно класть любое количество `Axes` (осей)

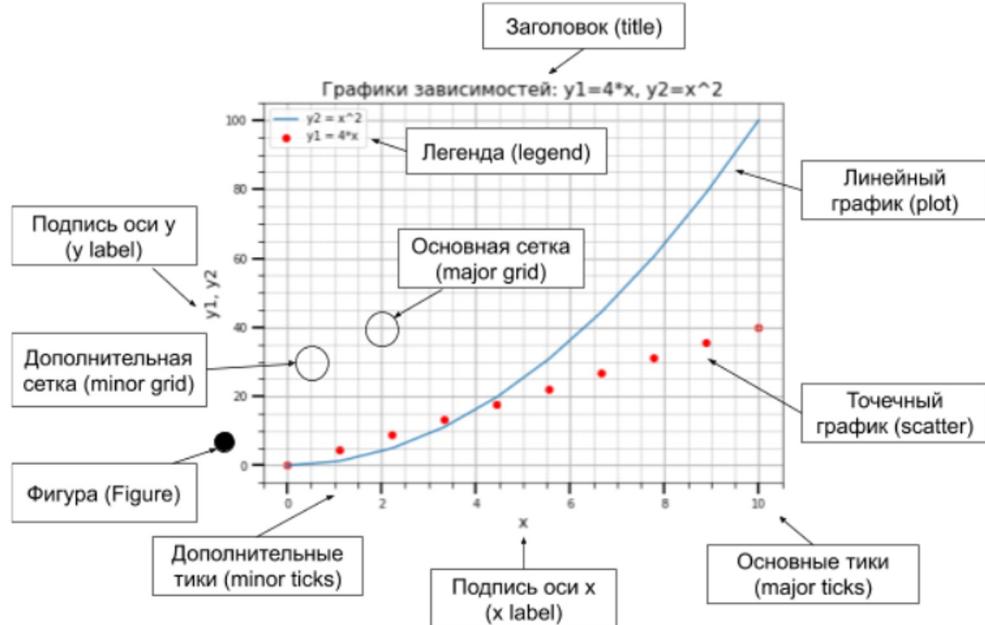
А уже в осях можно рисовать много графиков





Устройство графика

- 📌 title. Заголовок графика
- 📌 y label/x label. Именования осей x и y
- 📌 legend. Легенда к графикам
- 📌 major/minor ticks. Основной и дополнительный масштаб осей
- 📌 major/minor grid. Основная и дополнительная сетка на графике



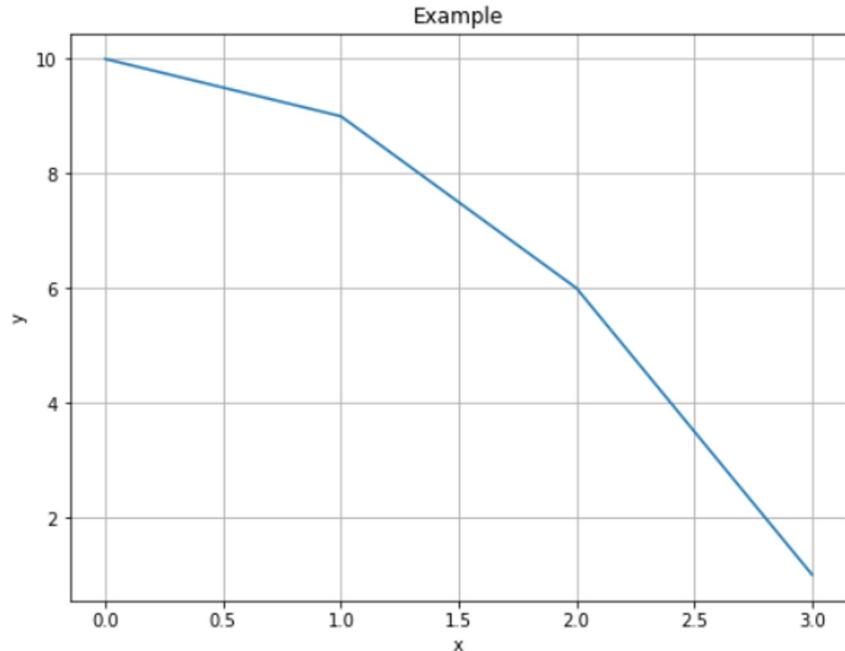


Устройство графика

Давайте создадим простой график, используя стандартный синтаксис:

- 📌 Вначале нужно создать объект фигуры и указать его размер plt.figure(figsize=(8, 6))
- 📌 Затем нарисовать график. Мы нарисуем 4 точки. Первые значения расположены по оси x, а вторые по оси y. plt.plot([0, 1, 2, 3], [10, 9, 6, 1])
- 📌 Можно добавить наименования осей plt.xlabel('x')
- 📌 Можно добавить заголовок plt.title('Example')
- 📌 Можно добавить сетку plt.grid()

```
plt.figure(figsize=(8, 6))
plt.plot([0, 1, 2, 3], [10, 9, 6, 1])
plt.xlabel('x')
plt.ylabel('y')
plt.title('Example')
plt.grid();
```





Виды графиков



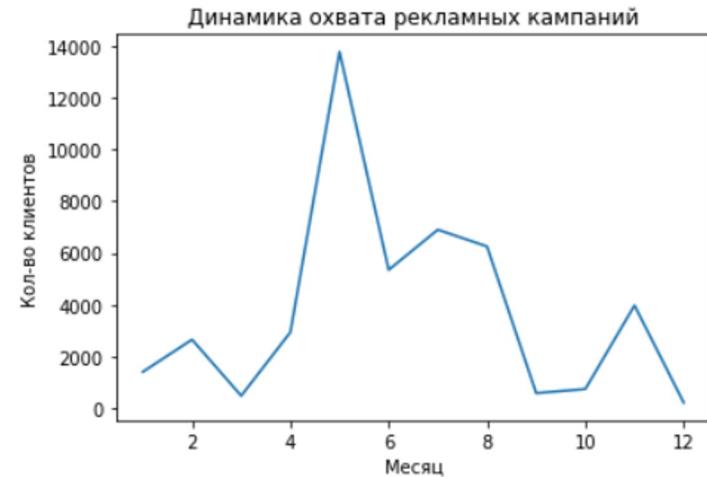
Линейный график

- 📌 Matplotlib - plt.plot()
- 📌 Seaborn - sns.lineplot()

```
plt.figure(figsize=(6, 4))

plt.plot(data['month'], data['count'])

plt.title('Динамика охвата рекламных кампаний')
plt.xlabel('Месяц')
plt.ylabel('Кол-во клиентов');
```

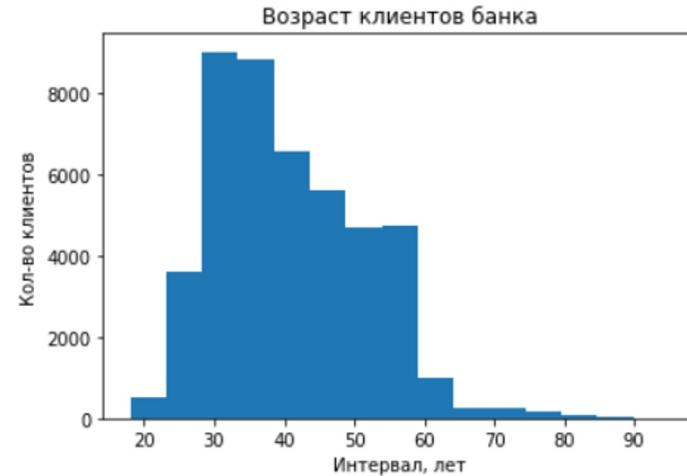




Гистограмма

- 📌 Matplotlib - plt.hist()
- 📌 Seaborn - sns.histplot()

```
plt.figure(figsize=(6, 4))  
plt.hist(df['age'], bins=15)  
plt.title('Возраст клиентов банка')  
plt.xlabel('Интервал, лет')  
plt.ylabel('Кол-во клиентов')|
```





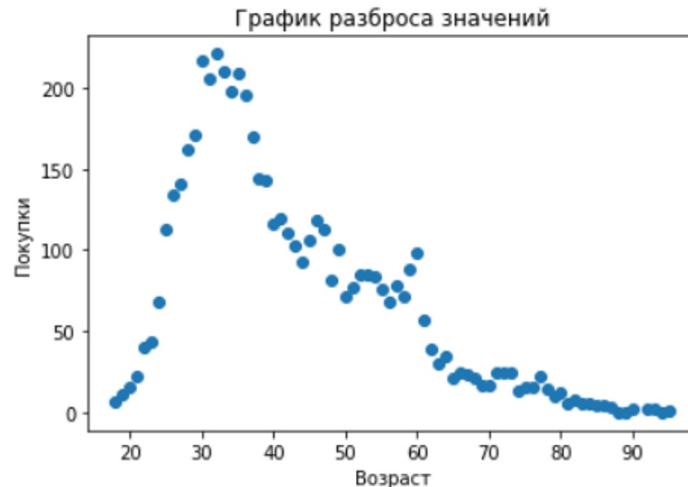
Точечный график

- 📌 Matplotlib - plt.scatter()
- 📌 Seaborn - sns.scatterplot()

```
plt.figure(figsize=(6, 4))

plt.scatter(data['age'], data['convert'])

plt.title('График разброса значений')
plt.xlabel('Возраст')
plt.ylabel('Покупки');
```

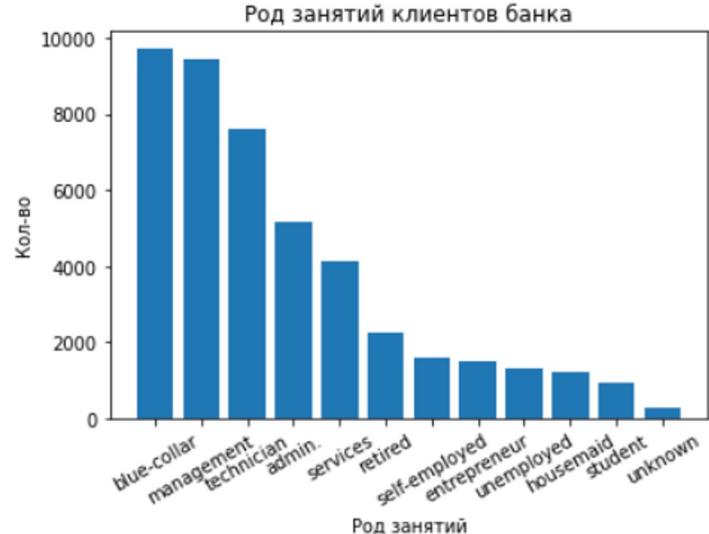




Столбчатая диаграмма

- 📌 Matplotlib - plt.bar()
- 📌 Seaborn - sns.barplot()

```
plt.figure(figsize=(6, 4))  
  
plt.bar(data['index'], data['job'])  
  
plt.title('Род занятых клиентов банка')  
plt.xlabel('Род занятых')  
plt.ylabel('Кол-во')  
plt.xticks(rotation=30);
```





Ящик с усами

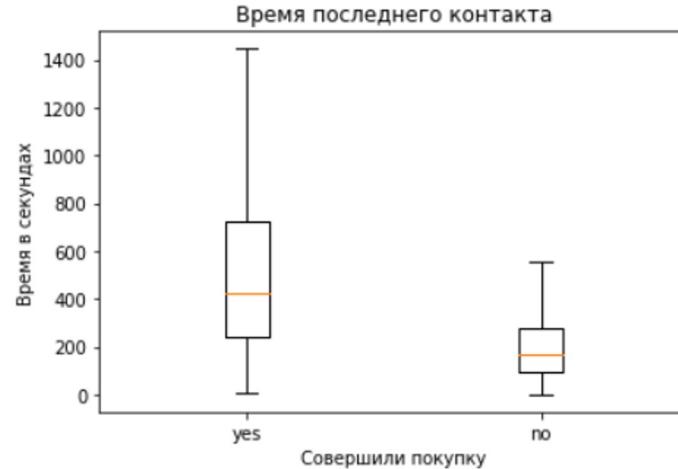
- 📌 Matplotlib - plt.boxplot()
- 📌 Seaborn - sns.boxplot()

```
# Строим боксплот без выбросов
plt.figure(figsize=(6, 4))

plt.boxplot([data1, data2], showfliers=False)

plt.title('Время последнего контакта')
plt.xlabel('Совершили покупку')

plt.ylabel('Время в секундах')
plt.xticks([1, 2], ['yes', 'no']);
```



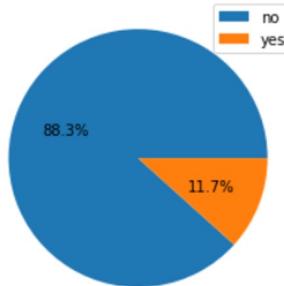


Круговая диаграмма

📌 Matplotlib - plt.pie()

```
plt.figure(figsize=(6, 4))  
plt.pie(data, autopct='%1.1f%%')  
plt.title('Доля клиентов банка, открывших депозит')  
plt.legend(data.index);
```

Доля клиентов банка, открывших депозит





Визуальный анализ данных



Визуальный анализ данных

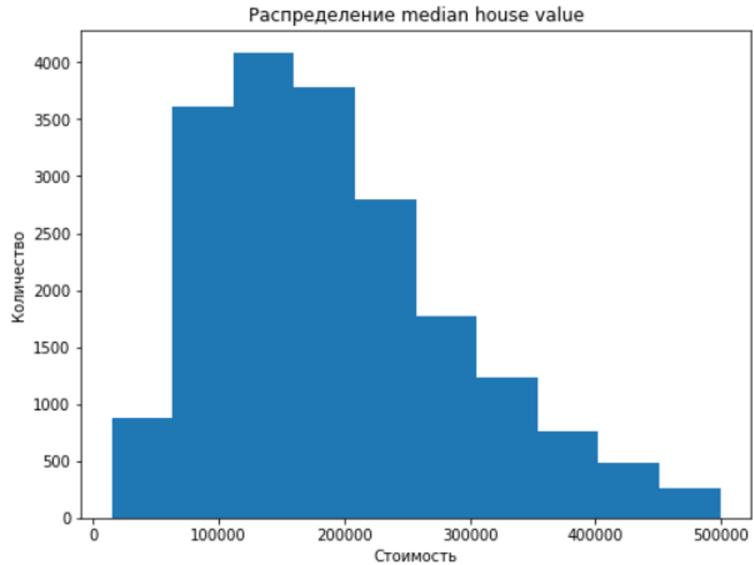
Три шага в EDA:

1. Анализ вещественных характеристик
2. Анализ категориальных признаков
3. Анализ взаимного распределения признаков



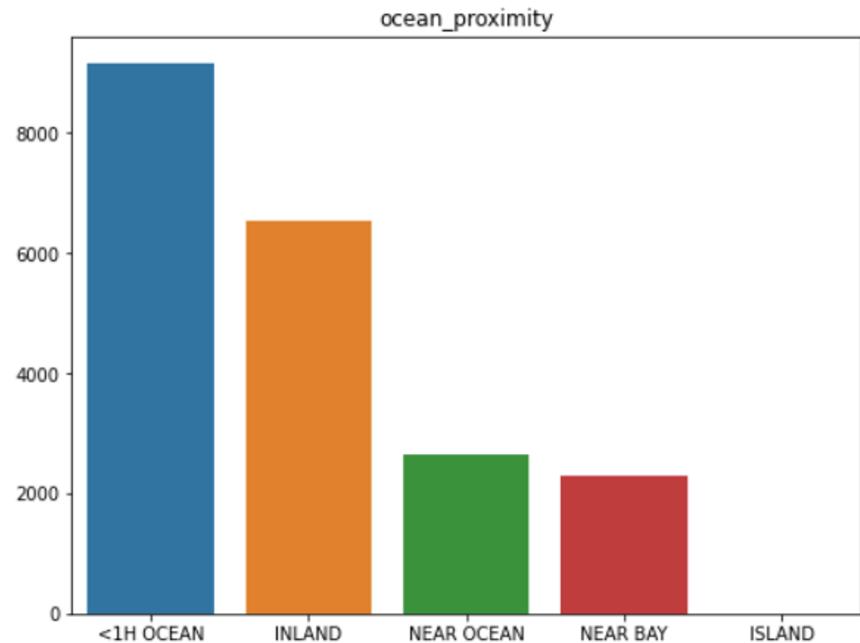
Анализ вещественных характеристик

Для таких признаков стоит изучить распределение, а это можно сделать построив гистограмму



Анализ категориальных характеристик

Для таких признаков тоже стоит изучить распределение, но уже используя столбчатые диаграммы

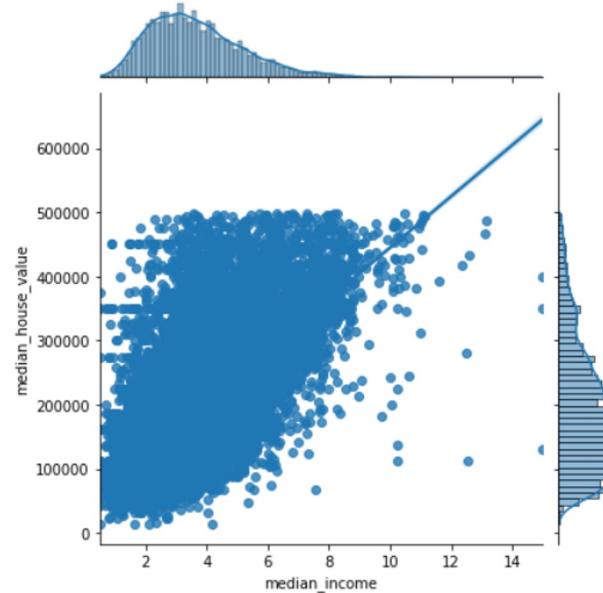




Анализ взаимного распределения (вещественные)

Для анализа подойдут графики взаимного распределения

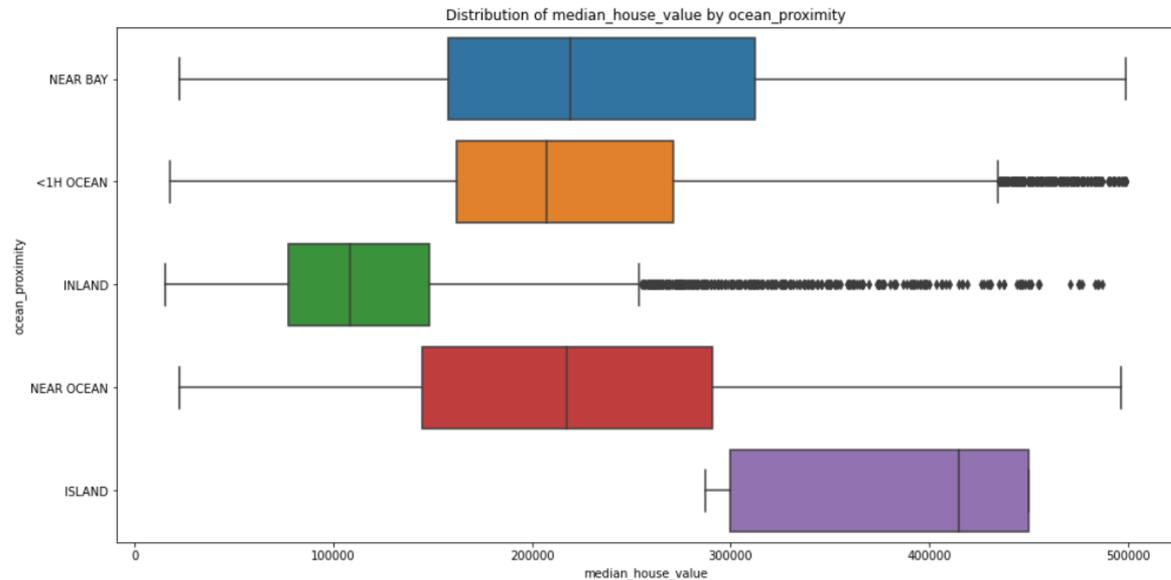
```
sns.jointplot(x=df['median_income'], y=df['median_house_value'], kind='reg');
```





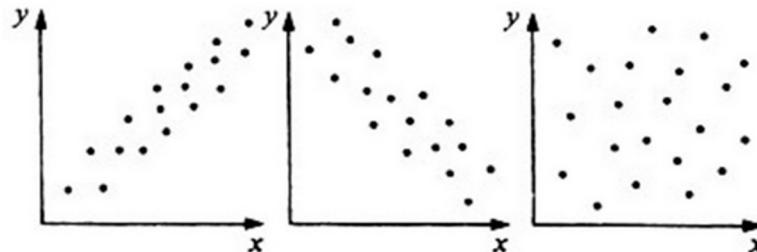
Анализ взаимного распределения (категориальные)

Для анализа подойдут графики боксплот





Анализ матрицы корреляций



A. Положительная B. Отрицательная C. Корреляция отсутствует

Величина коэффициента корреляции	0.1 - 0.3	0.3 - 0.5	0.5 – 0.7	0.7 – 0.9	0.9 – 1.0
Характеристика силы связи	слабая	умеренная	заметная	высокая	очень высокая



Визуализация геоданных

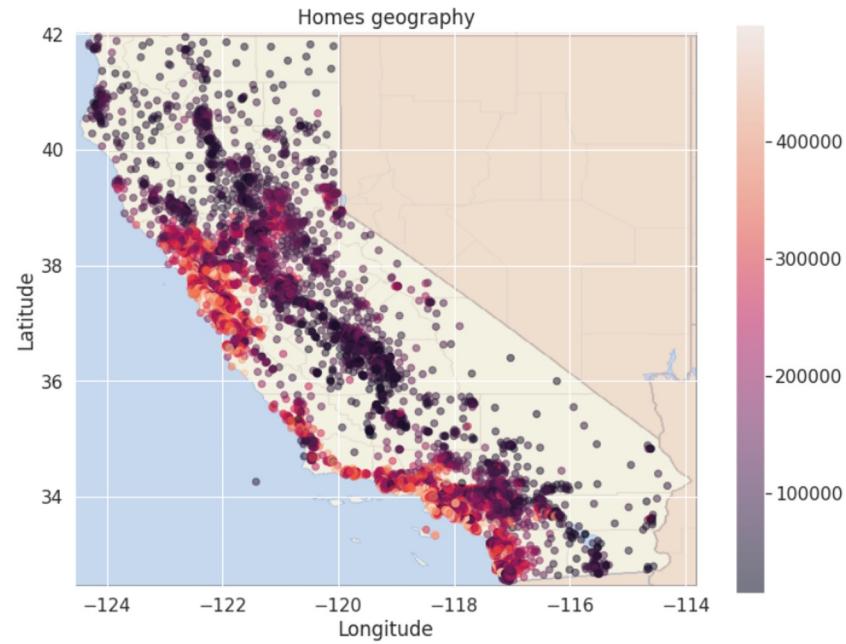


Matplotlib

```
import matplotlib.image as img
california_map = img.imread('California_Map.png')

plt.figure(figsize=(12, 9))
plt.imshow(california_map,
           extent=[min_long, max_long, min_lat, max_lat], alpha=0.5);

sc = plt.scatter(df['longitude'], df['latitude'], alpha=0.5, c=df['median_house_value'])
```





Folium

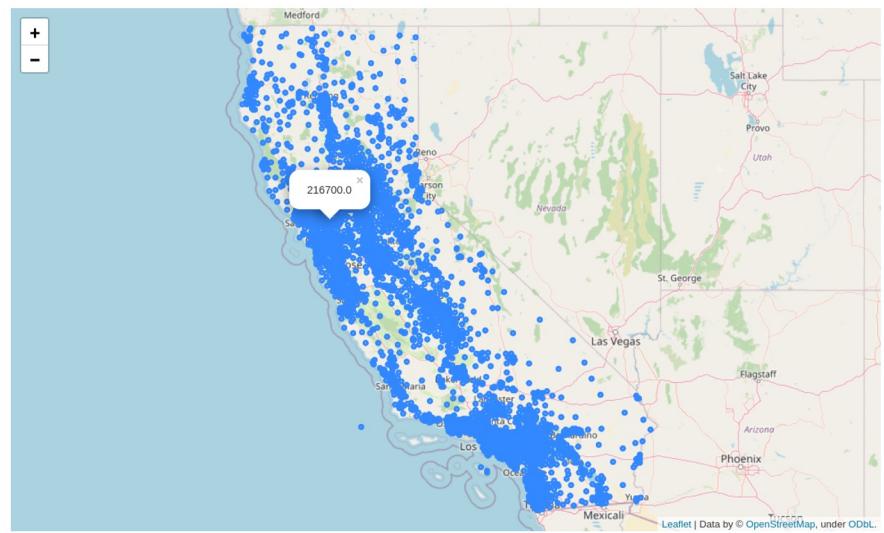
```
import folium
this_map = folium.Map(prefer_canvas=True)

def plotDot(point):
    folium.CircleMarker(
        location=[point.latitude, point.longitude],
        radius=2,
        popup=point.median_house_value
    ).add_to(this_map)

df.apply(plotDot, axis=1)

this_map.fit_bounds(this_map.get_bounds())

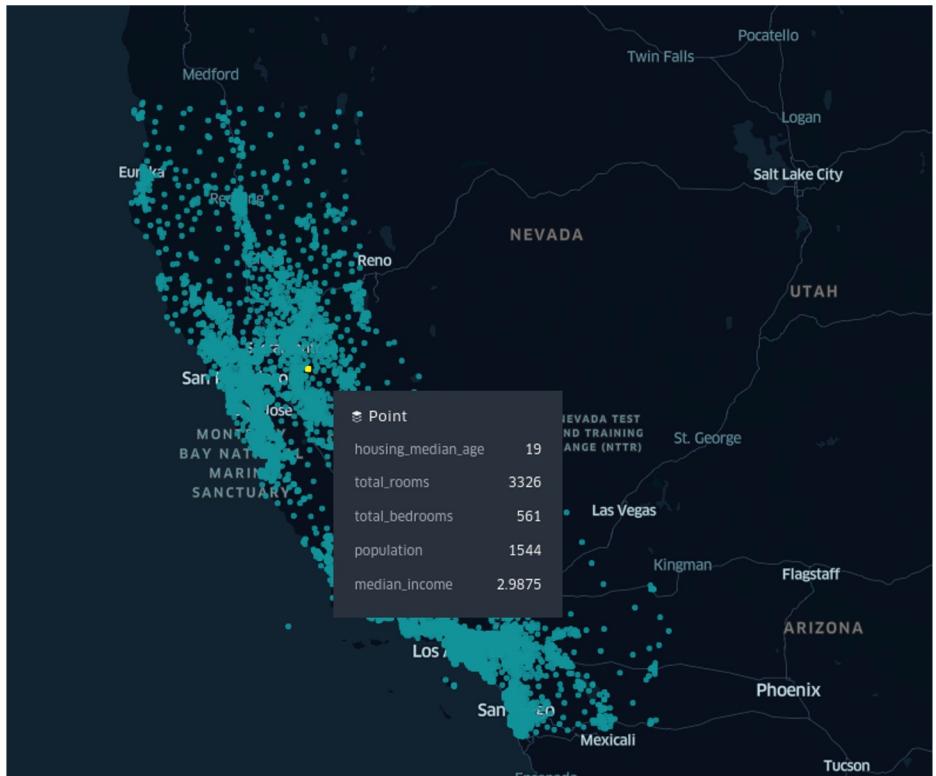
this_map
```





KeplerGL

```
from keplergl import KeplerGl  
  
map_ = KeplerGl(height=700)  
map_.add_data(df, 'Data')  
map_.save_to_html(file_name='./california.html')
```





Итоги урока

- 📌 Познакомились с видами графиков
- 📌 Научились строить и интерпретировать графики
- 📌 Рассмотрели визуальный анализ данных
- 📌 Поняли как можно анализировать геоданные





Спасибо за урок!

