Лабораторная работа № 1

Киреев Андрей ИУ5-63Б ¶

В качестве датасета использован https://www.kaggle.com/harlfoxem/housesalesprediction (https://www.kaggle.com/harlfoxem/housesalesprediction)

This dataset contains house sale prices for King County, which includes Seattle. It includes homes sold between May 2014 and May 2015.

Импорт библиотек

In [6]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
%matplotlib inline
```

In [7]:
```
data = pd.read_csv('kc_house_data.csv', sep=",")
```

Основные характеристики датасета

Первые пять строк датасета

In [9]:
```
data.head()
```

Out[9]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	w
0	7129300520	20141013T000000	221900.0	3	1.00	1180	5650	1.0	
1	6414100192	20141209T000000	538000.0	3	2.25	2570	7242	2.0	
2	5631500400	20150225T000000	180000.0	2	1.00	770	10000	1.0	
3	2487200875	20141209T000000	604000.0	4	3.00	1960	5000	1.0	
4	1954400510	20150218T000000	510000.0	3	2.00	1680	8080	1.0	

5 rows × 21 columns

Размер

In [12]:
```
data.shape
```

Out[12]:
```
(21613, 21)
```

Колонки их их типы данных

In [13]:
```
data.dtypes
```

Out[13]:
```
id                int64
date             object
price           float64
bedrooms          int64
bathrooms       float64
sqft_living       int64
sqft_lot          int64
floors          float64
waterfront        int64
view              int64
condition         int64
grade             int64
sqft_above        int64
sqft_basement     int64
yr_built          int64
yr_renovated      int64
zipcode           int64
lat             float64
long            float64
sqft_living15     int64
sqft_lot15        int64
dtype: object
```

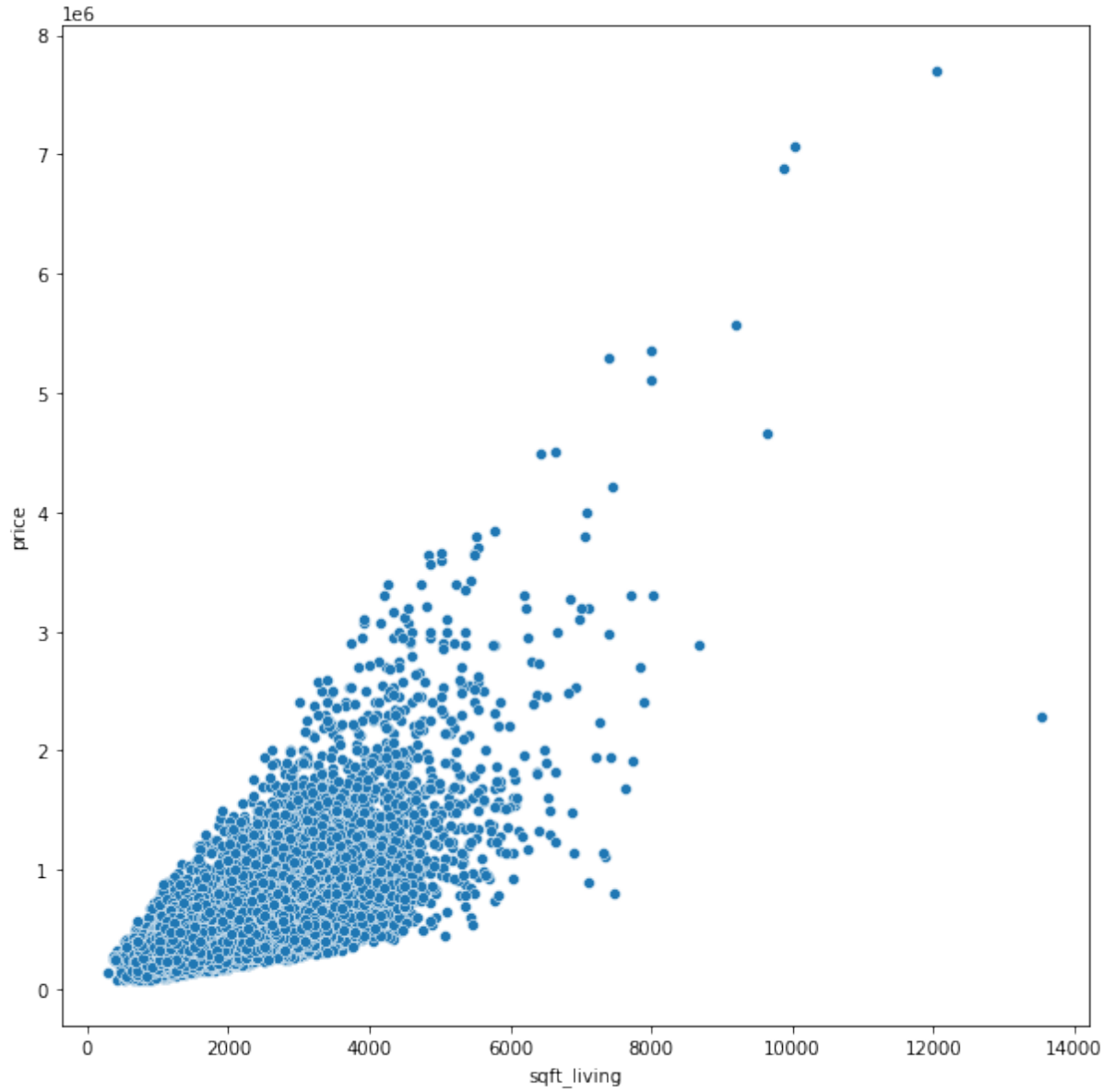
In [14]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21613 entries, 0 to 21612
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   id             21613 non-null  int64  
 1   date           21613 non-null  object 
 2   price          21613 non-null  float64
 3   bedrooms       21613 non-null  int64  
 4   bathrooms      21613 non-null  float64
 5   sqft_living    21613 non-null  int64  
 6   sqft_lot       21613 non-null  int64  
 7   floors         21613 non-null  float64
 8   waterfront     21613 non-null  int64  
 9   view           21613 non-null  int64  
 10  condition      21613 non-null  int64  
 11  grade          21613 non-null  int64  
 12  sqft_above     21613 non-null  int64  
 13  sqft_basement  21613 non-null  int64  
 14  yr_built       21613 non-null  int64  
 15  yr_renovated   21613 non-null  int64  
 16  zipcode        21613 non-null  int64  
 17  lat            21613 non-null  float64
 18  long           21613 non-null  float64
 19  sqft_living15  21613 non-null  int64  
 20  sqft_lot15     21613 non-null  int64  
dtypes: float64(5), int64(15), object(1)
memory usage: 3.5+ MB
```

Визуальное исследование датасета

In [17]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='sqft_living', y='price', data=data)
```

Out[17]: <AxesSubplot:xlabel='sqft_living', ylabel='price'>

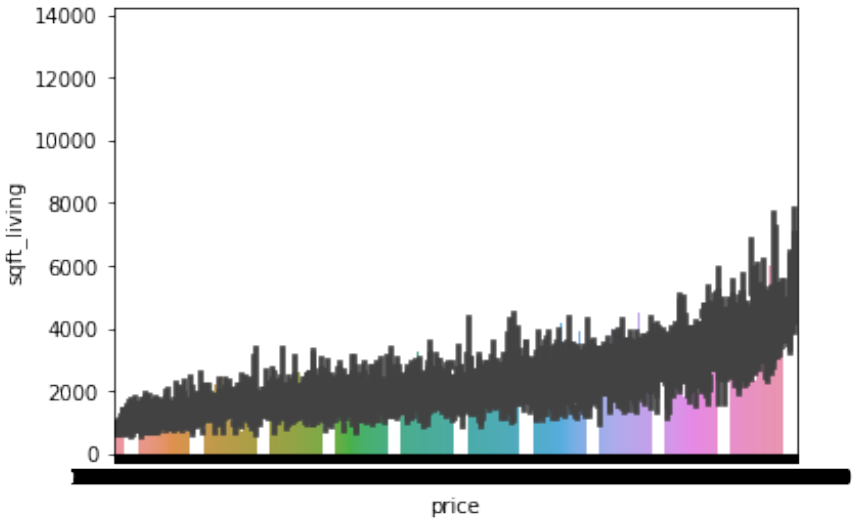


Можно увидеть,что между полями sqft_living и price присутствует корреляция: чем больше жилаю площадь,чем выше цена дома (что очень логично))

Гистограмма

In [19]:
```
sns.barplot(x=data['price'], y='sqft_living', data=data)
```

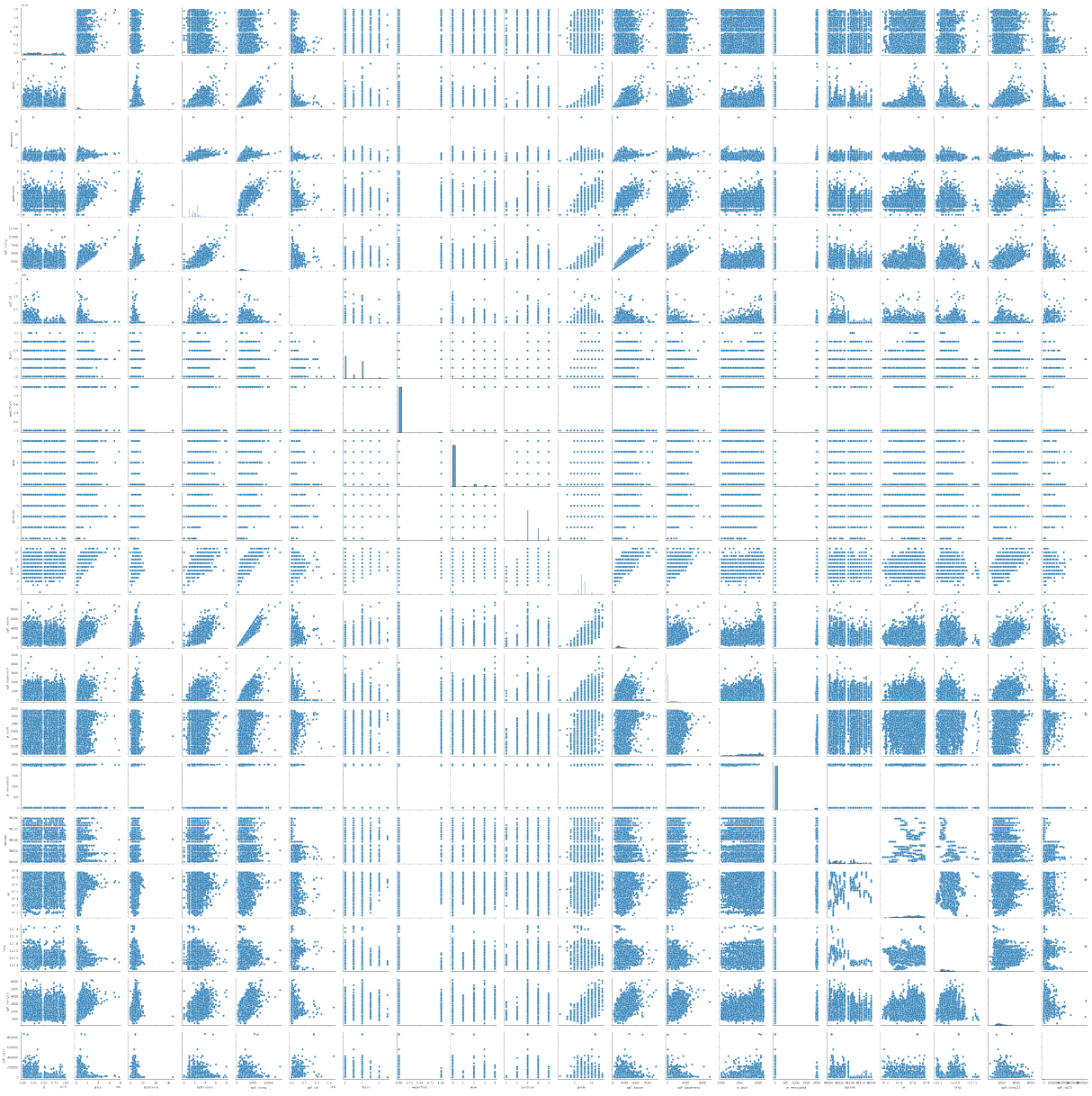
Out[19]:
```
<AxesSubplot:xlabel='price', ylabel='sqft_living'>
```



Парные диаграммы

In [21]:
```
sns.pairplot(data)
```

Out[21]:
```
<seaborn.axisgrid.PairGrid at 0x137bb1790>
```

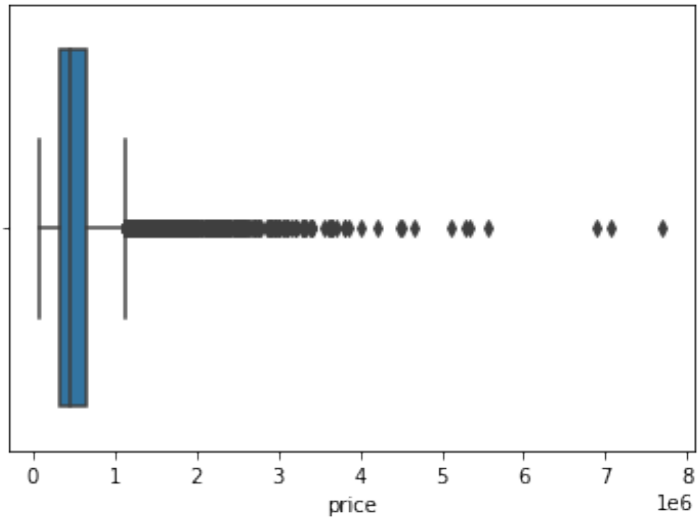


Ящик с усами

Отображает одномерное распределение вероятности.

In [22]:
```
sns.boxplot(x=data['price'])
```

Out[22]:
```
<AxesSubplot:xlabel='price'>
```

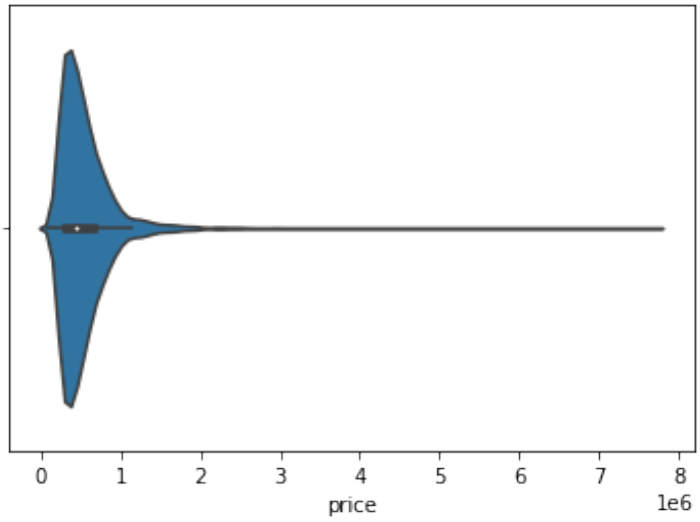


Violin plot

Похоже на предыдущую диаграмму, но по краям отображаются распределения плотности:

In [23]:
```
sns.violinplot(x=data['price'])
```

Out[23]:
```
<AxesSubplot:xlabel='price'>
```



Информация о корреляция признаков

Проверка корреляции признаков позволяет решить две задачи:

- Понять какие признаки (колонки датасета) наиболее сильно коррелируют с целевым признаком (колонка "price"). Именно эти признаки будут наиболее информативными для моделей машинного обучения. Признаки, которые слабо коррелируют с целевым признаком, можно попробовать исключить из построения модели, иногда это повышает качество модели.
- Понять какие нецелевые признаки линейно зависимы между собой. Линейно зависимые признаки, как правило, очень плохо влияют на качество моделей. Поэтому если несколько признаков линейно зависимы, то для построения модели из них выбирают какой-то один признак.

In [24]:
```
data.corr()
```

Out[24]:

	id	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	wa
id	1.000000	-0.016762	0.001286	0.005160	-0.012258	-0.132109	0.018525	-0
price	-0.016762	1.000000	0.308350	0.525138	0.702035	0.089661	0.256794	0
bedrooms	0.001286	0.308350	1.000000	0.515884	0.576671	0.031703	0.175429	-0
bathrooms	0.005160	0.525138	0.515884	1.000000	0.754665	0.087740	0.500653	0
sqft_living	-0.012258	0.702035	0.576671	0.754665	1.000000	0.172826	0.353949	0
sqft_lot	-0.132109	0.089661	0.031703	0.087740	0.172826	1.000000	-0.005201	0
floors	0.018525	0.256794	0.175429	0.500653	0.353949	-0.005201	1.000000	0
waterfront	-0.002721	0.266369	-0.006582	0.063744	0.103818	0.021604	0.023698	1
view	0.011592	0.397293	0.079532	0.187737	0.284611	0.074710	0.029444	0
condition	-0.023783	0.036362	0.028472	-0.124982	-0.058753	-0.008958	-0.263768	0
grade	0.008130	0.667434	0.356967	0.664983	0.762704	0.113621	0.458183	0
sqft_above	-0.010842	0.605567	0.477600	0.685342	0.876597	0.183512	0.523885	0
sqft_basement	-0.005151	0.323816	0.303093	0.283770	0.435043	0.015286	-0.245705	0
yr_built	0.021380	0.054012	0.154178	0.506019	0.318049	0.053080	0.489319	-0
yr_renovated	-0.016907	0.126434	0.018841	0.050739	0.055363	0.007644	0.006338	0
zipcode	-0.008224	-0.053203	-0.152668	-0.203866	-0.199430	-0.129574	-0.059121	0
lat	-0.001891	0.307003	-0.008931	0.024573	0.052529	-0.085683	0.049614	-0
long	0.020799	0.021626	0.129473	0.223042	0.240223	0.229521	0.125419	-0
sqft_living15	-0.002901	0.585379	0.391638	0.568634	0.756420	0.144608	0.279885	0
sqft_lot15	-0.138798	0.082447	0.029244	0.087175	0.183286	0.718557	-0.011269	0

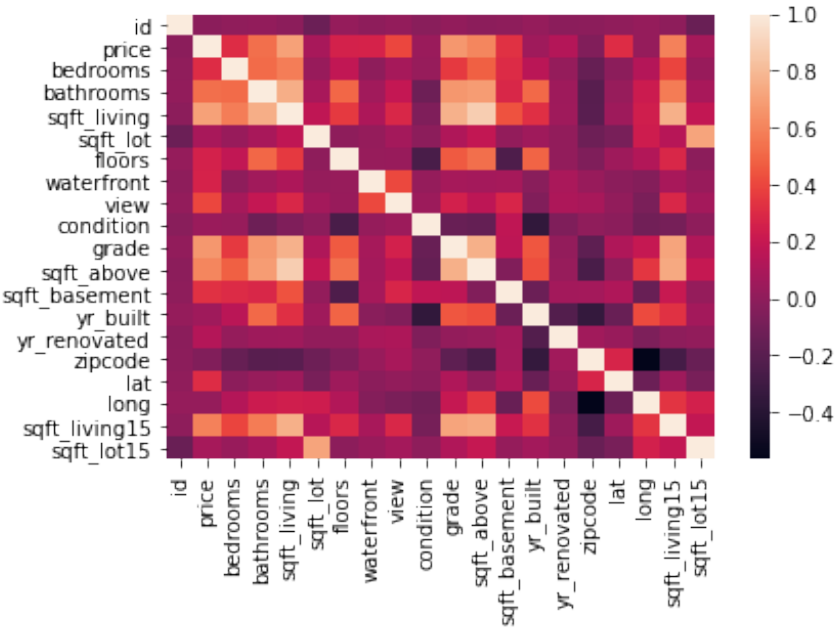
Корреляционная матрица содержит коэффициенты корреляции между всеми парами признаков. На основе корреляционной матрицы можно сделать следующие выводы:

- Целевой признак наиболее сильно коррелирует с жилой площадь/sqft_living (0.702035). Этот признак обязательно следует оставить в модели.
- Целевой признак слабо коррелирует с состоянием/condition (0.036362), количеством этажей/floors (0.256794). Скорее всего эти признаки стоит исключить из модели, возможно они только ухудшат качество модели.

В случае большого количества признаков анализ числовой корреляционной матрицы становится неудобен. Для визуализации корреляционной матрицы будем использовать "тепловую карту" heatmap которая показывает степень корреляции различными цветами.

In [29]:
```
sns.heatmap(data.corr())
```

Out[29]:
```
<AxesSubplot:>
```



In [ ]:
```

```