King County Dataset
Exploratory Data Analysis (EDA)

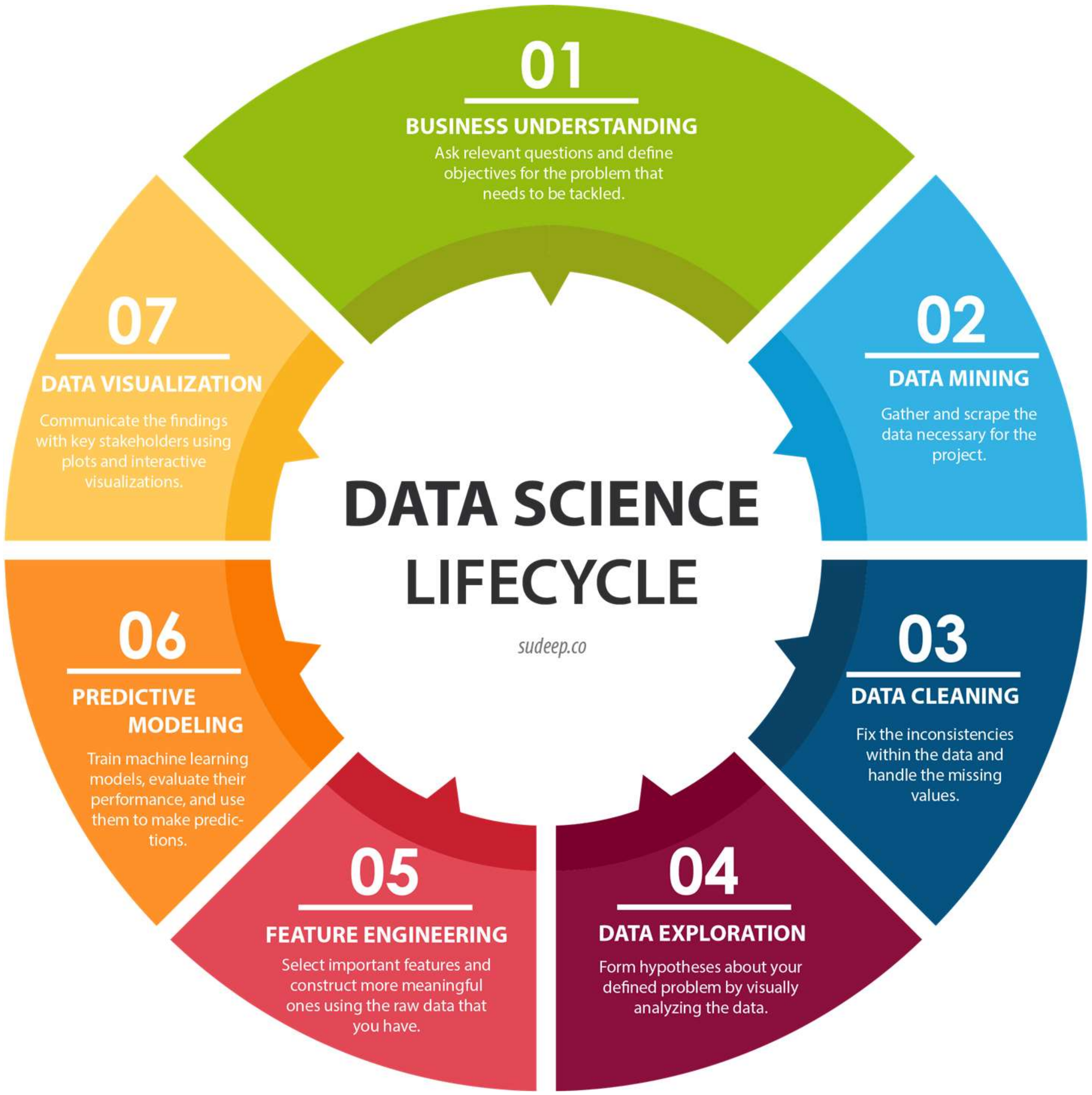
Last Week

- Introduction to Machine Learning
- Unsupervised / Supervised Learning
- Batch / Online Learning
- Instance-based / Model-based learning
- Underfitting and Overfitting
- Training, validation and test sets

End-to-End Machine Learning

Aurélien Géron, ***Hands-on-Machine Learning***

1. Look at the big picture
2. Get the data and set aside a test set
3. Discover and visualise the data to gain insights
4. Prepare the data for Machine Learning algorithms
5. Identify a suitable metric for evaluating the task
6. Select a model and train it
7. Fine-tune your model
8. Present your solution
9. Launch, monitor and maintain your system



DATA SCIENCE LIFECYCLE

sudeep.co

01

BUSINESS UNDERSTANDING

Ask relevant questions and define objectives for the problem that needs to be tackled.

02

DATA MINING

Gather and scrape the data necessary for the project.

03

DATA CLEANING

Fix the inconsistencies within the data and handle the missing values.

04

DATA EXPLORATION

Form hypotheses about your defined problem by visually analyzing the data.

05

FEATURE ENGINEERING

Select important features and construct more meaningful ones using the raw data that you have.

06

PREDICTIVE MODELING

Train machine learning models, evaluate their performance, and use them to make predictions.

07

DATA VISUALIZATION

Communicate the findings with key stakeholders using plots and interactive visualizations.

End-to-End Machine Learning

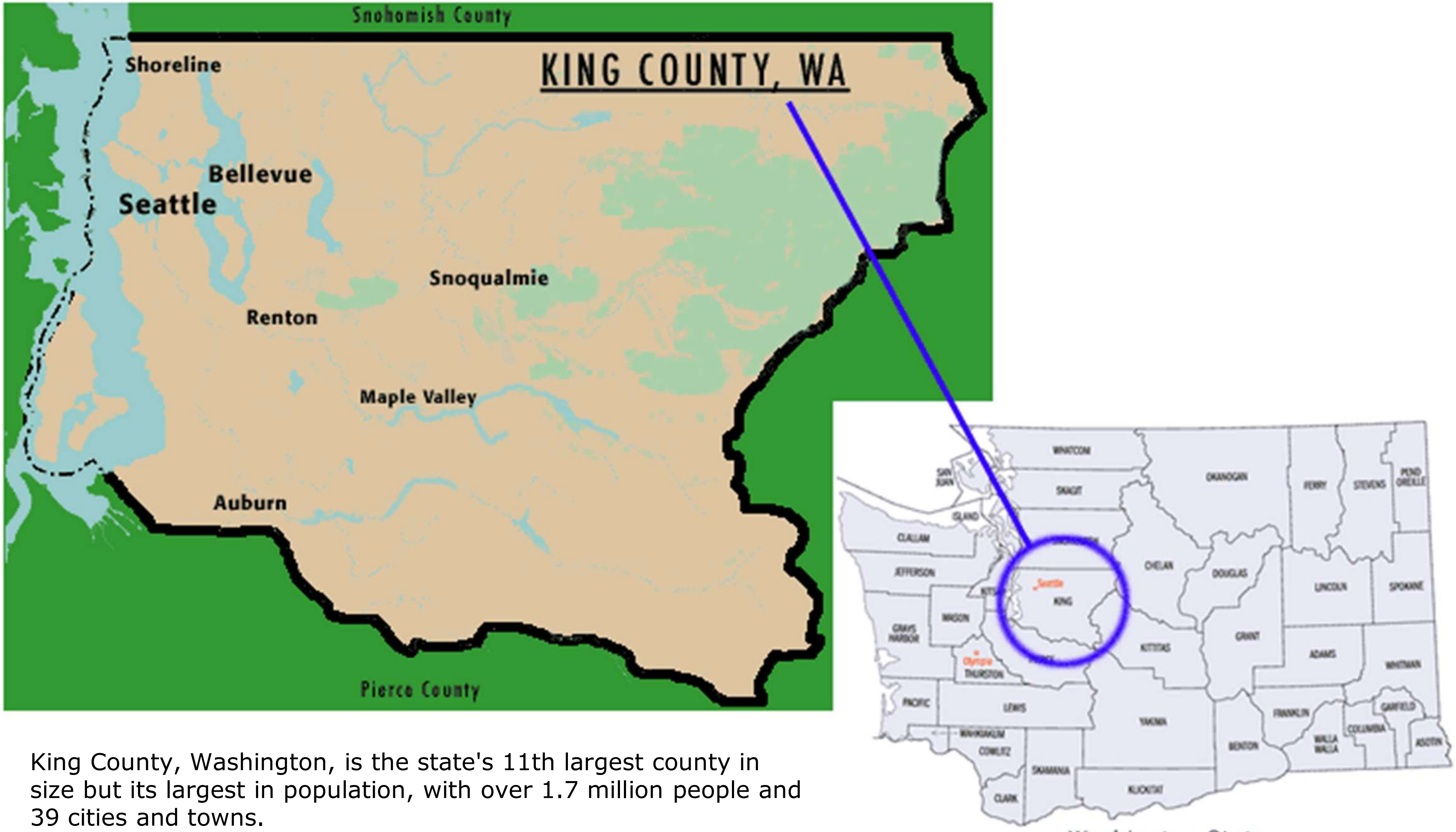
Aurélien Géron, ***Hands-on-Machine Learning***

1. Look at the big picture
2. Get the data and set aside a test set
3. Discover and visualise the data to gain insights
4. Prepare the data for Machine Learning algorithms
5. Identify a suitable metric for evaluating the task
6. Select a model and train it
7. Fine-tune your model
8. Present your solution
9. Launch, monitor and maintain your system

1. Frame the Problem

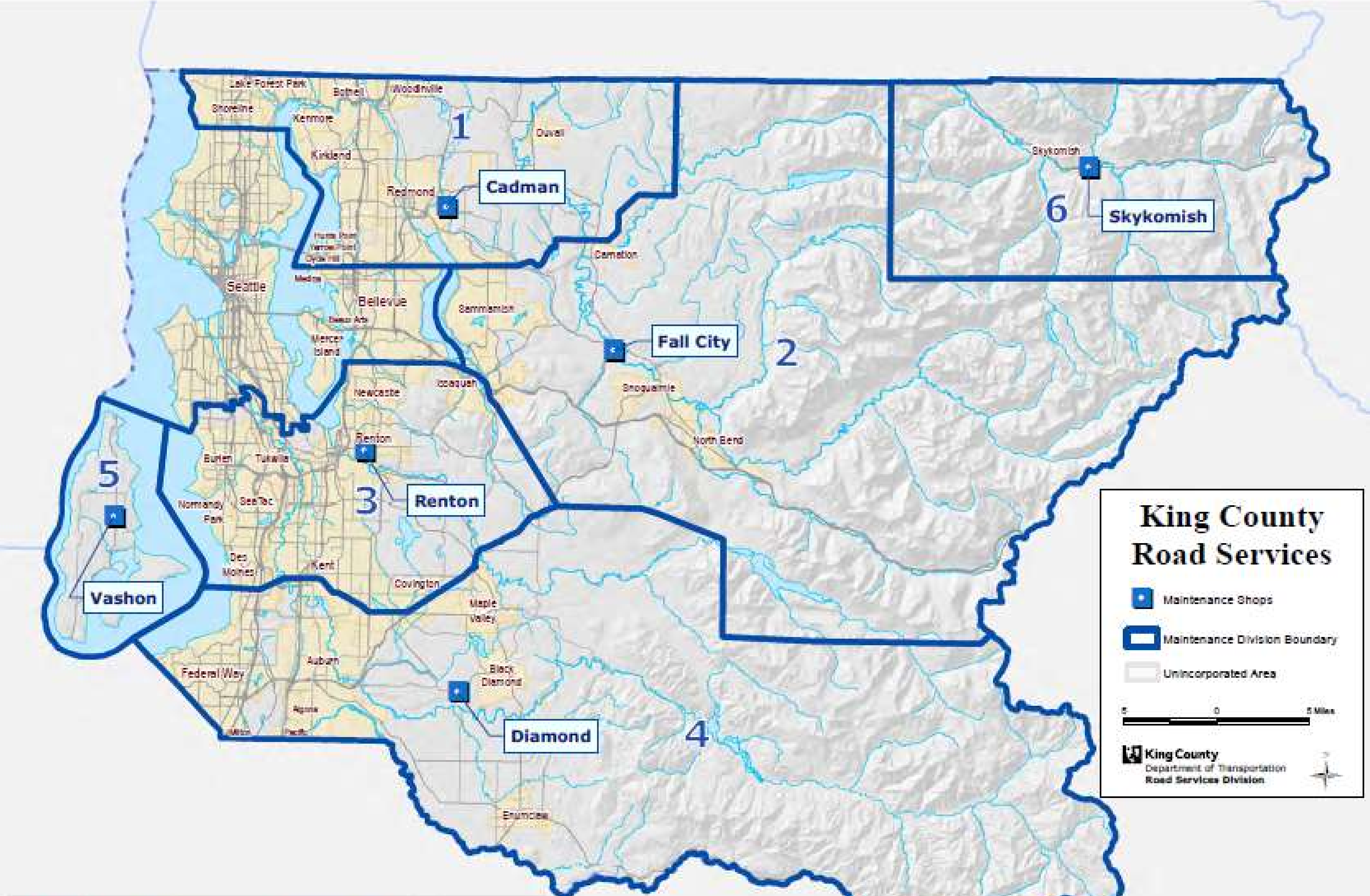
- We want to be able to **predict the price of houses** in King County, Washington, US.

- Questions for you to consider:
    - Is it supervised, unsupervised, or reinforcement learning?
    - Is it a classification task, a regression task or something else?
    - Should you use batch learning or online learning techniques?



King County, Washington, is the state's 11th largest county in size but its largest in population, with over 1.7 million people and 39 cities and towns.

Washington State
map courtesy US Census Bureau



King County Road Services

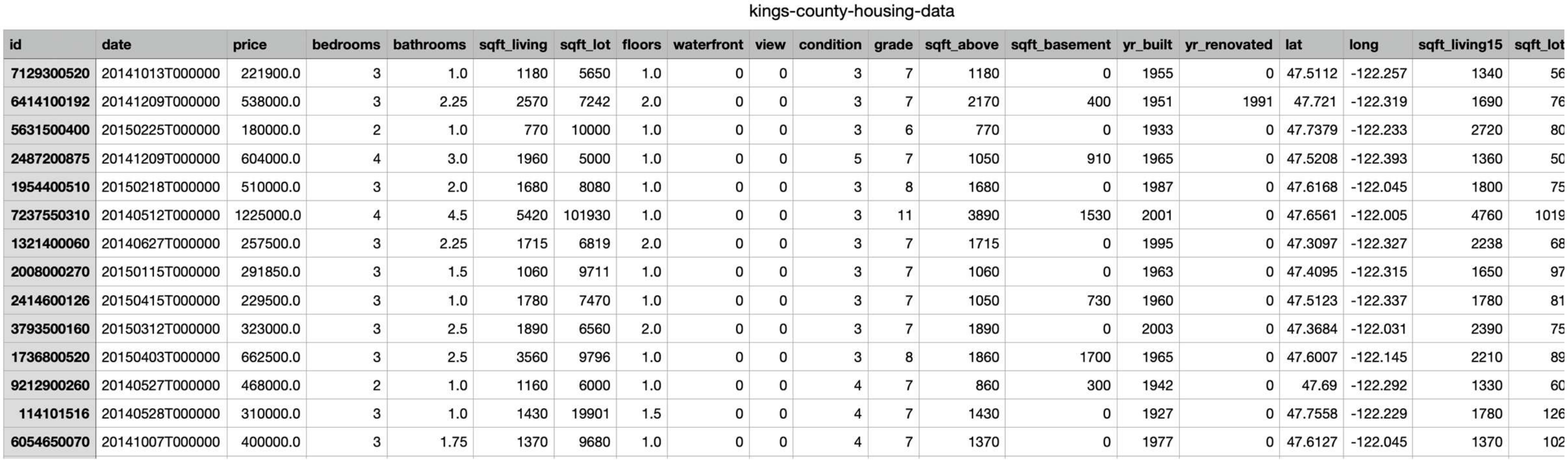
The information included on this map has been compiled by King County staff from a variety of sources and is subject to change without notice. King County makes no representations or warranties, express or implied, as to accuracy, completeness, timeliness, or rights to the use of such information. This document is not intended for use as a survey product. King County shall not be liable for any general, special, indirect, incidental, or consequential damages including, but not limited to, lost revenues or lost profits resulting from the use or misuse of the information contained on this map. Any sale of this map or information on this map is prohibited except by written permission of King County.

Date Saved: 10/14/2014
\dot\roads\SBO62\GIS\Projects\RoadMaintenanceDivisionBoundaries\MXD\RSD_MaintDiv-6hp_Sml.mxd

2. Get the data

- Create a workspace (with enough storage space).
- Get the data
- Convert the data to a format you can easily manipulate (without changing the data itself).
- Ensure sensitive information is deleted or protected (e.g. anonymised)
- Check the size and type of data (time series, sample, geographical)
- **Sample a test set, put it aside,** and never look at it (no data snooping!)

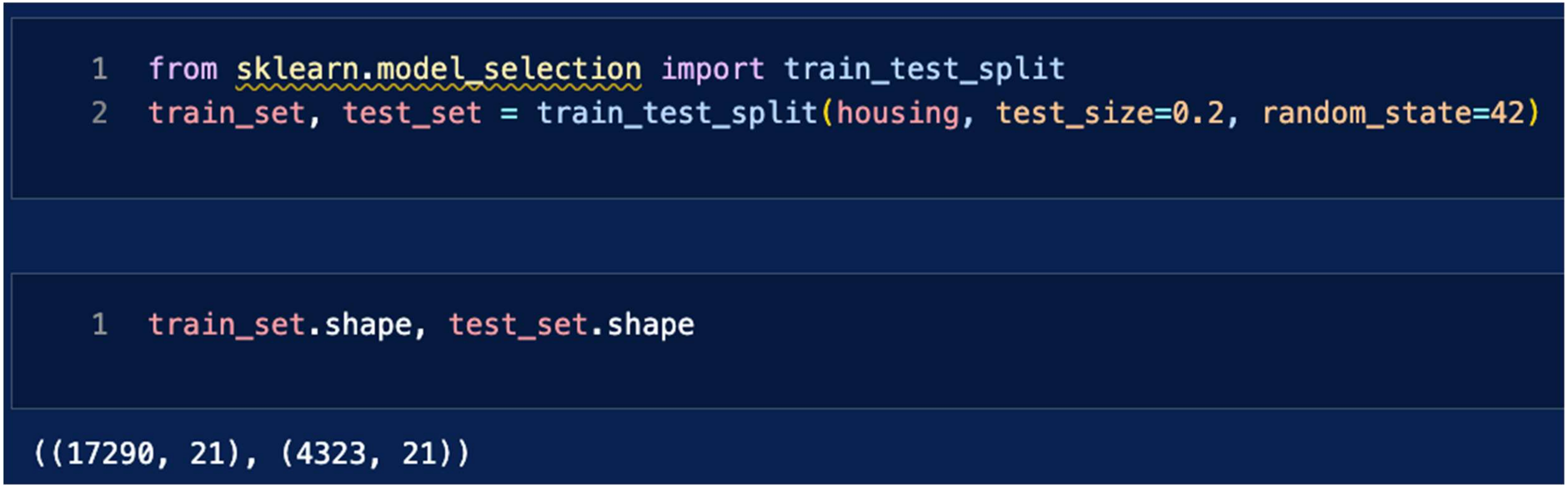
2. Get the data



kings-county-housing-data

id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	sqft_above	sqft_basement	yr_built	yr_renovated	lat	long	sqft_living15	sqft_lot
7129300520	20141013T000000	221900.0	3	1.0	1180	5650	1.0	0	0	3	7	1180	0	1955	0	47.5112	-122.257	1340	56
6414100192	20141209T000000	538000.0	3	2.25	2570	7242	2.0	0	0	3	7	2170	400	1951	1991	47.721	-122.319	1690	76
5631500400	20150225T000000	180000.0	2	1.0	770	10000	1.0	0	0	3	6	770	0	1933	0	47.7379	-122.233	2720	80
2487200875	20141209T000000	604000.0	4	3.0	1960	5000	1.0	0	0	5	7	1050	910	1965	0	47.5208	-122.393	1360	50
1954400510	20150218T000000	510000.0	3	2.0	1680	8080	1.0	0	0	3	8	1680	0	1987	0	47.6168	-122.045	1800	75
7237550310	20140512T000000	1225000.0	4	4.5	5420	101930	1.0	0	0	3	11	3890	1530	2001	0	47.6561	-122.005	4760	1019
1321400060	20140627T000000	257500.0	3	2.25	1715	6819	2.0	0	0	3	7	1715	0	1995	0	47.3097	-122.327	2238	68
2008000270	20150115T000000	291850.0	3	1.5	1060	9711	1.0	0	0	3	7	1060	0	1963	0	47.4095	-122.315	1650	97
2414600126	20150415T000000	229500.0	3	1.0	1780	7470	1.0	0	0	3	7	1050	730	1960	0	47.5123	-122.337	1780	81
3793500160	20150312T000000	323000.0	3	2.5	1890	6560	2.0	0	0	3	7	1890	0	2003	0	47.3684	-122.031	2390	75
1736800520	20150403T000000	662500.0	3	2.5	3560	9796	1.0	0	0	3	8	1860	1700	1965	0	47.6007	-122.145	2210	89
9212900260	20140527T000000	468000.0	2	1.0	1160	6000	1.0	0	0	4	7	860	300	1942	0	47.69	-122.292	1330	60
114101516	20140528T000000	310000.0	3	1.0	1430	19901	1.5	0	0	4	7	1430	0	1927	0	47.7558	-122.229	1780	126
6054650070	20141007T000000	400000.0	3	1.75	1370	9680	1.0	0	0	4	7	1370	0	1977	0	47.6127	-122.045	1370	102

2. train_test_split ahead of EDA



```python
from sklearn.model_selection import train_test_split
train_set, test_set = train_test_split(housing, test_size=0.2, random_state=42)
```

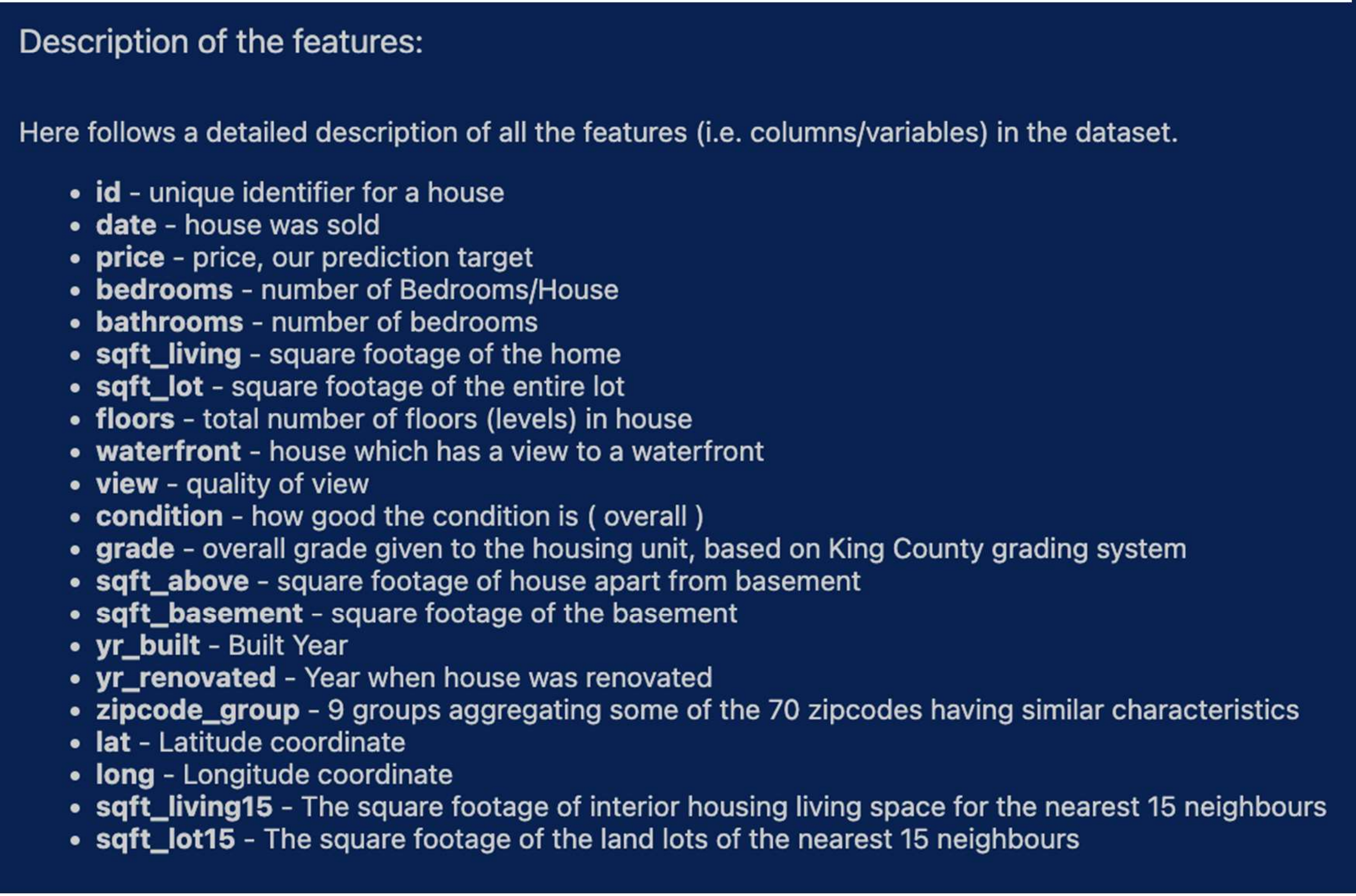
```python
train_set.shape, test_set.shape
```

```
((17290, 21), (4323, 21))
```

3. Inspect the data to gain insights

- Study each attribute and its characteristics
    - Name
    - Type (categorical, int/float, bounded/unbounded, text, structured etc)
    - % of missing values
    - Noisiness and type of noise (stochastic, outliers, rounding errors etc)
    - Usefulness for the task
    - Type of distribution (Gaussian, uniform, logarithmic etc)
- For supervised learning tasks, identify the target attribute(s)

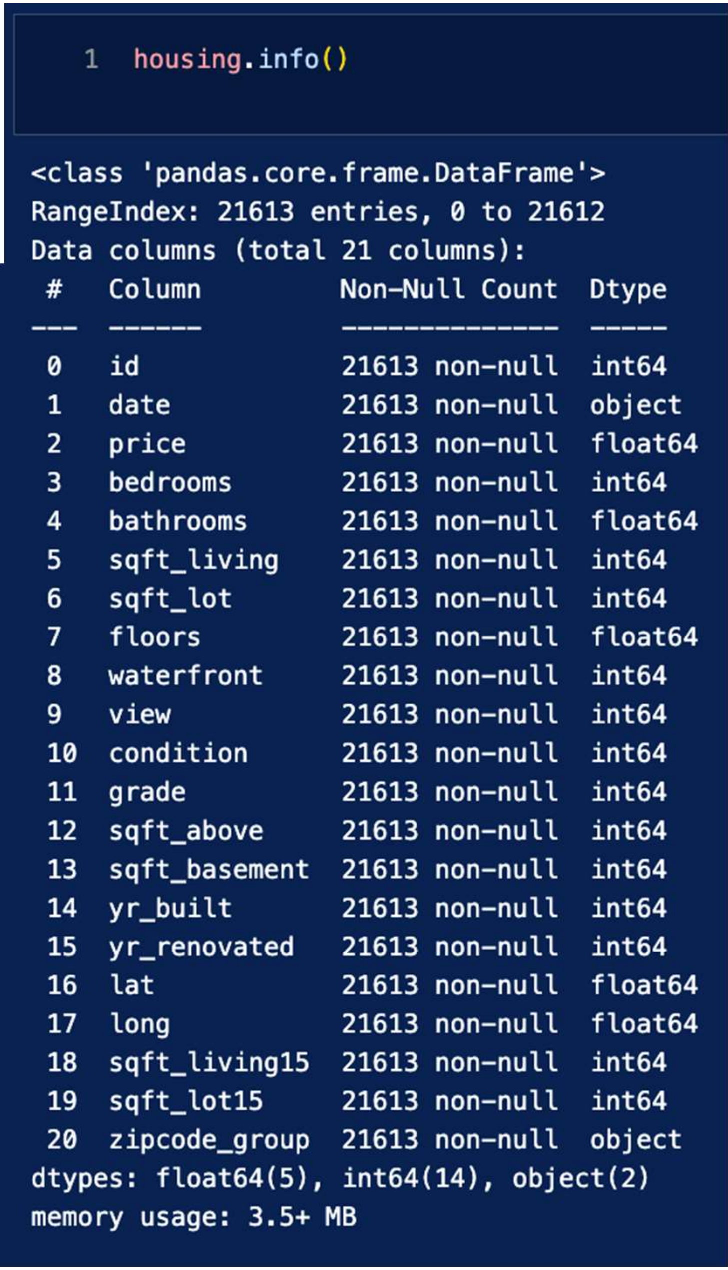
3. Inspect the data



Description of the features:

Here follows a detailed description of all the features (i.e. columns/variables) in the dataset.

- **id** - unique identifier for a house
- **date** - house was sold
- **price** - price, our prediction target
- **bedrooms** - number of Bedrooms/House
- **bathrooms** - number of bedrooms
- **sqft_living** - square footage of the home
- **sqft_lot** - square footage of the entire lot
- **floors** - total number of floors (levels) in house
- **waterfront** - house which has a view to a waterfront
- **view** - quality of view
- **condition** - how good the condition is (overall)
- **grade** - overall grade given to the housing unit, based on King County grading system
- **sqft_above** - square footage of house apart from basement
- **sqft_basement** - square footage of the basement
- **yr_built** - Built Year
- **yr_renovated** - Year when house was renovated
- **zipcode_group** - 9 groups aggregating some of the 70 zipcodes having similar characteristics
- **lat** - Latitude coordinate
- **long** - Longitude coordinate
- **sqft_living15** - The square footage of interior housing living space for the nearest 15 neighbours
- **sqft_lot15** - The square footage of the land lots of the nearest 15 neighbours



```
housing.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  lat             21613 non-null  float64
 17  long            21613 non-null  float64
 18  sqft_living15   21613 non-null  int64
 19  sqft_lot15      21613 non-null  int64
 20  zipcode_group   21613 non-null  object
dtypes: float64(5), int64(14), object(2)
memory usage: 3.5+ MB
```

3. Exploratory Data Analysis (EDA)

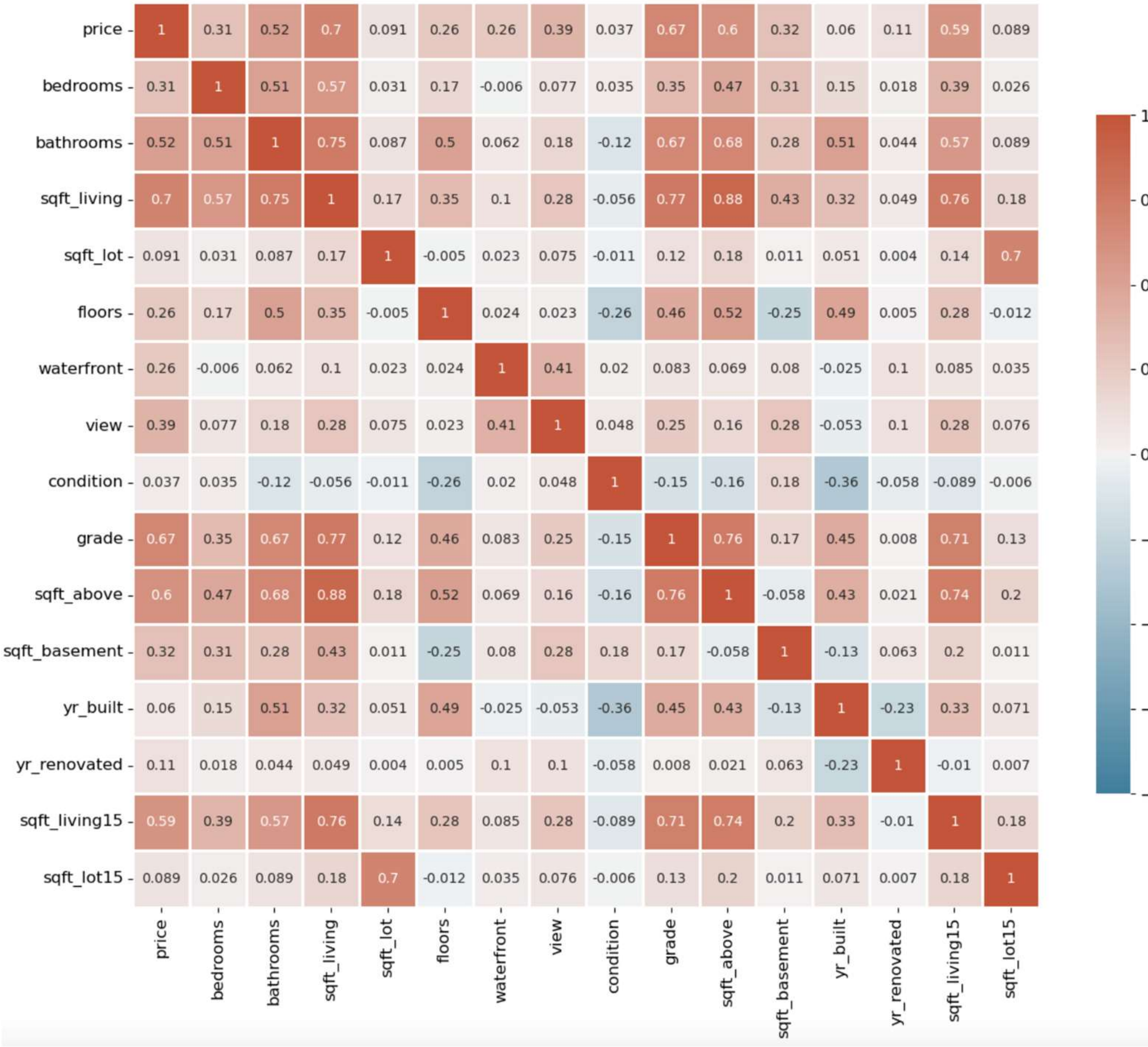
- Visualise the data
- Study the correlations between attributes
- Study how you would solve the problem manually
- Identify the promising transformations you may want to apply
- Identify extra data that would be useful to the investigation

Seaborn

- High level data visualisation library based on matplotlib.
- Ideal for drawing informative statistical graphics
- Dataset-oriented API for examining relationships between multiple variables
- Convenient views onto the overall structure of complex datasets
- High-level abstractions for structuring multi-plot grids that let you easily build complex visualisations
- Concise control over matplotlib figure styling with several built-in themes

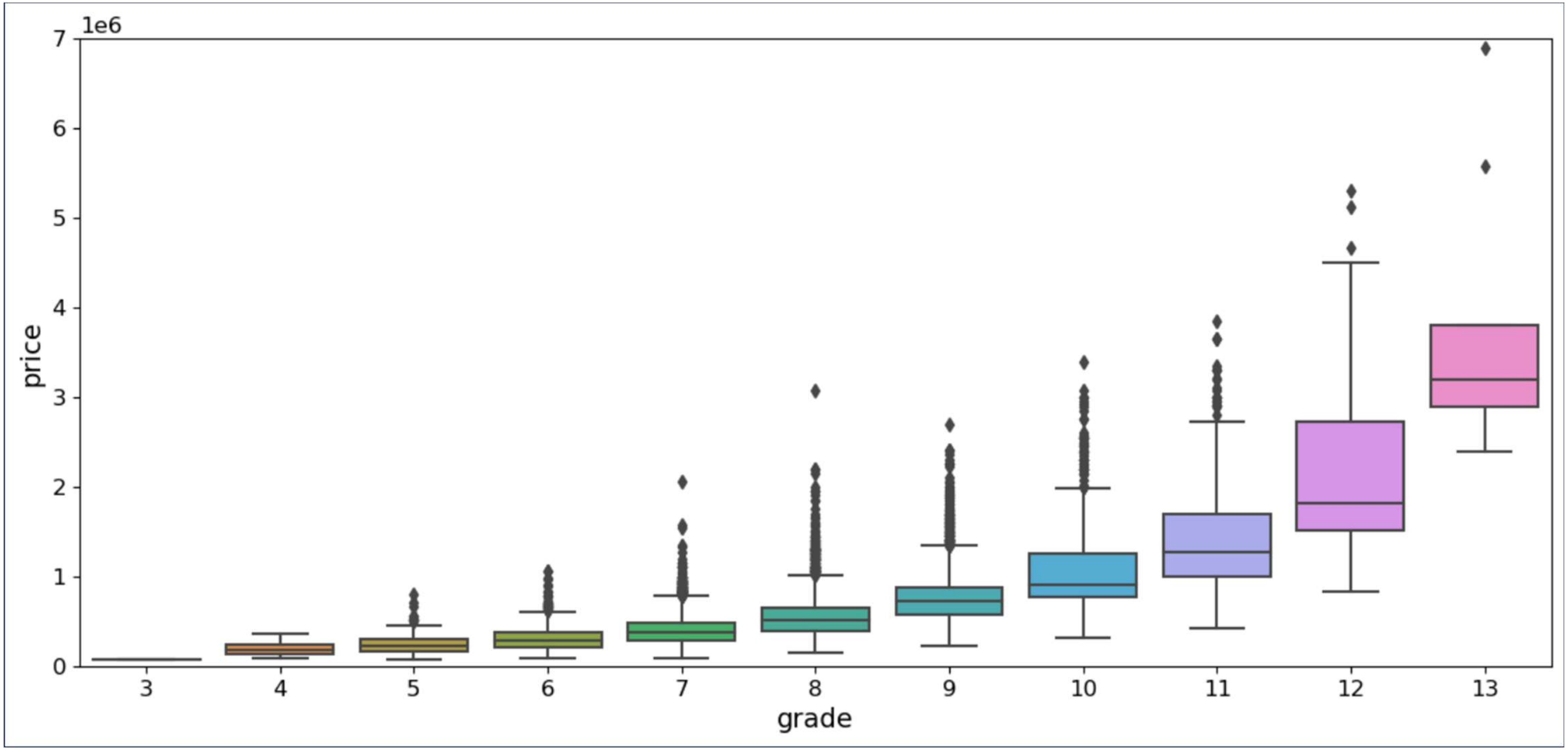
sns.heatmap

- Pearson's correlation coefficient drawn as a 'heat' map
- Colour coded for accessibility
- +1 = perfect positive correlation – hotter!
- -1 = perfect negative correlation – cooler!

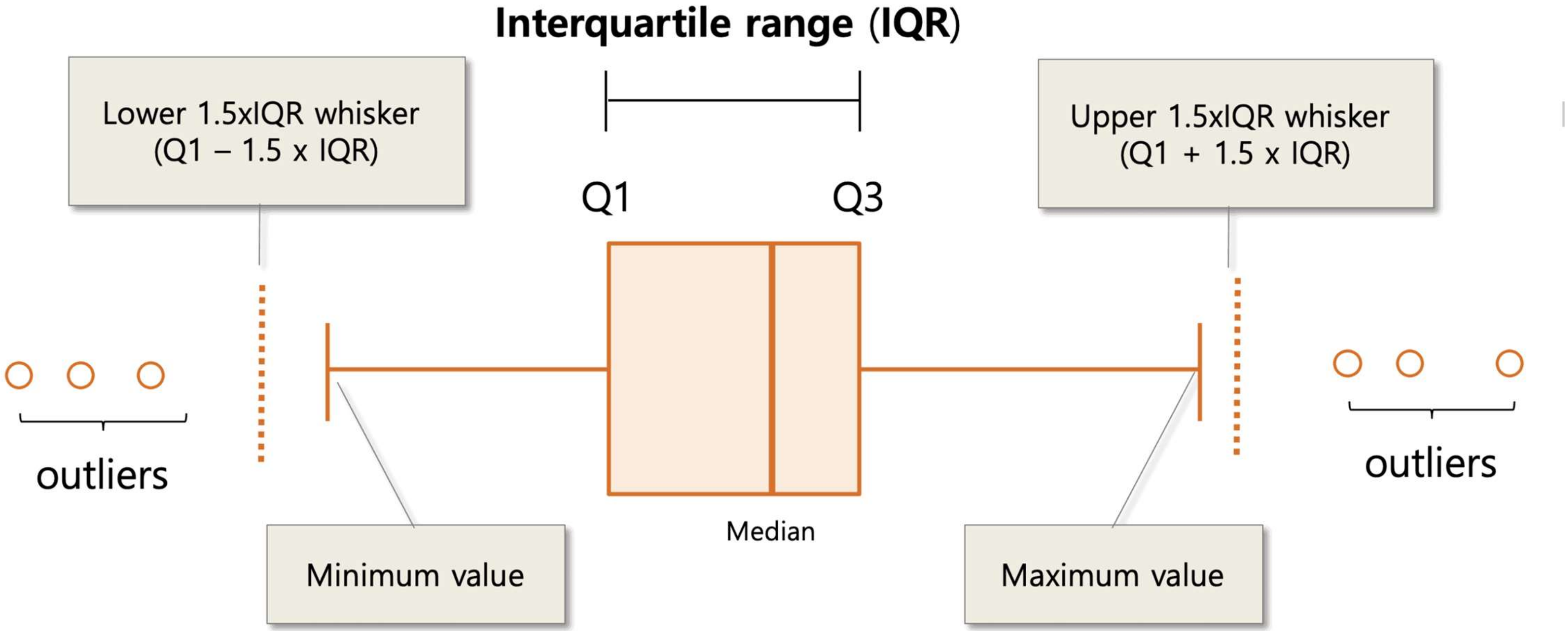


	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	sqft_above	sqft_basement	yr_built	yr_renovated	sqft_living15	sqft_lot15
price	1	0.31	0.52	0.7	0.091	0.26	0.26	0.39	0.037	0.67	0.6	0.32	0.06	0.11	0.59	0.089
bedrooms	0.31	1	0.51	0.57	0.031	0.17	-0.006	0.077	0.035	0.35	0.47	0.31	0.15	0.018	0.39	0.026
bathrooms	0.52	0.51	1	0.75	0.087	0.5	0.062	0.18	-0.12	0.67	0.68	0.28	0.51	0.044	0.57	0.089
sqft_living	0.7	0.57	0.75	1	0.17	0.35	0.1	0.28	-0.056	0.77	0.88	0.43	0.32	0.049	0.76	0.18
sqft_lot	0.091	0.031	0.087	0.17	1	-0.005	0.023	0.075	-0.011	0.12	0.18	0.011	0.051	0.004	0.14	0.7
floors	0.26	0.17	0.5	0.35	-0.005	1	0.024	0.023	-0.26	0.46	0.52	-0.25	0.49	0.005	0.28	-0.012
waterfront	0.26	-0.006	0.062	0.1	0.023	0.024	1	0.41	0.02	0.083	0.069	0.08	-0.025	0.1	0.085	0.035
view	0.39	0.077	0.18	0.28	0.075	0.023	0.41	1	0.048	0.25	0.16	0.28	-0.053	0.1	0.28	0.076
condition	0.037	0.035	-0.12	-0.056	-0.011	-0.26	0.02	0.048	1	-0.15	-0.16	0.18	-0.36	-0.058	-0.089	-0.006
grade	0.67	0.35	0.67	0.77	0.12	0.46	0.083	0.25	-0.15	1	0.76	0.17	0.45	0.008	0.71	0.13
sqft_above	0.6	0.47	0.68	0.88	0.18	0.52	0.069	0.16	-0.16	0.76	1	-0.058	0.43	0.021	0.74	0.2
sqft_basement	0.32	0.31	0.28	0.43	0.011	-0.25	0.08	0.28	0.18	0.17	-0.058	1	-0.13	0.063	0.2	0.011
yr_built	0.06	0.15	0.51	0.32	0.051	0.49	-0.025	-0.053	-0.36	0.45	0.43	-0.13	1	-0.23	0.33	0.071
yr_renovated	0.11	0.018	0.044	0.049	0.004	0.005	0.1	0.1	-0.058	0.008	0.021	0.063	-0.23	1	-0.01	0.007
sqft_living15	0.59	0.39	0.57	0.76	0.14	0.28	0.085	0.28	-0.089	0.71	0.74	0.2	0.33	-0.01	1	0.18
sqft_lot15	0.089	0.026	0.089	0.18	0.7	-0.012	0.035	0.076	-0.006	0.13	0.2	0.011	0.071	0.007	0.18	1

sns.boxplot

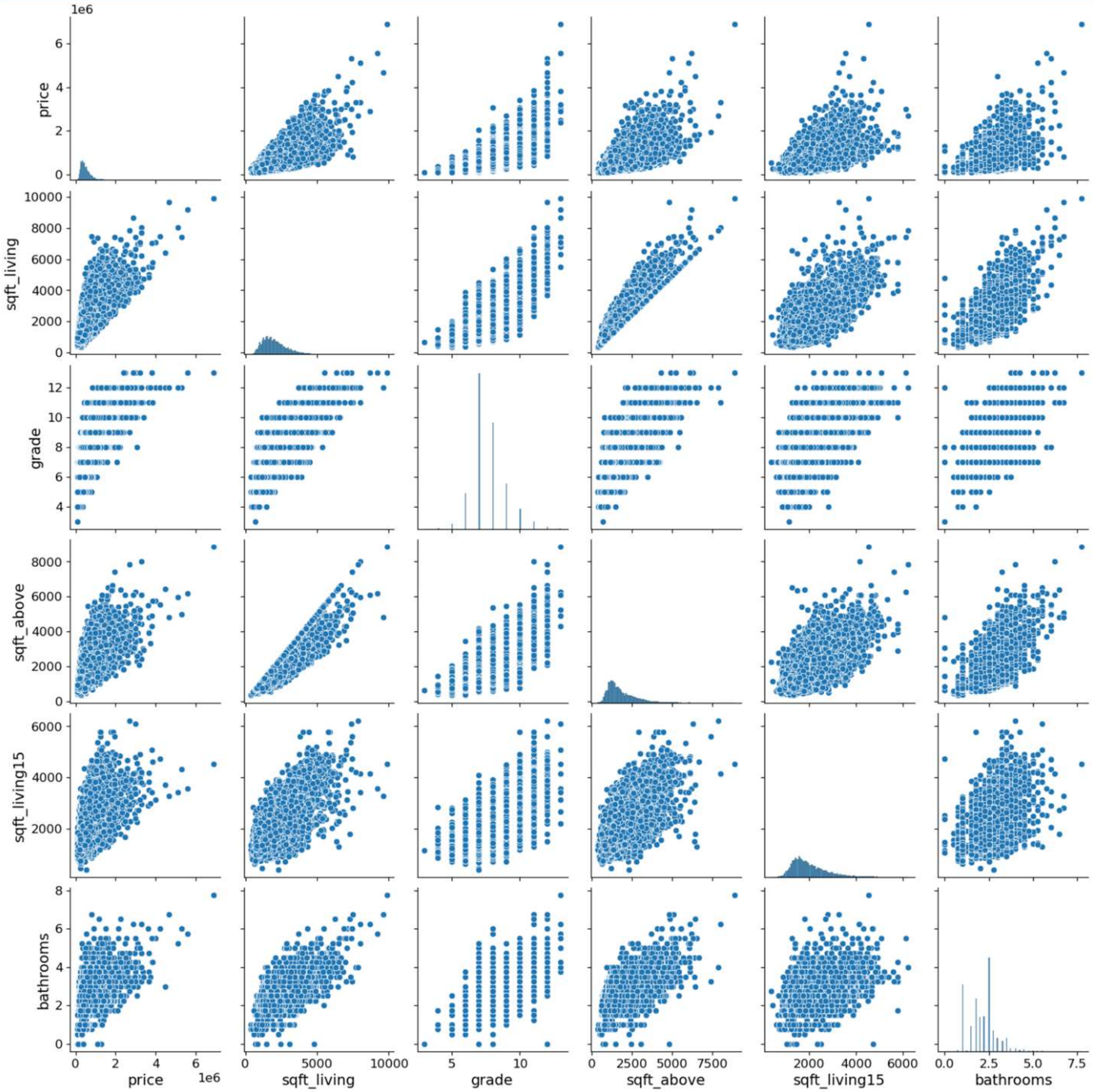


Boxplots and IQRs

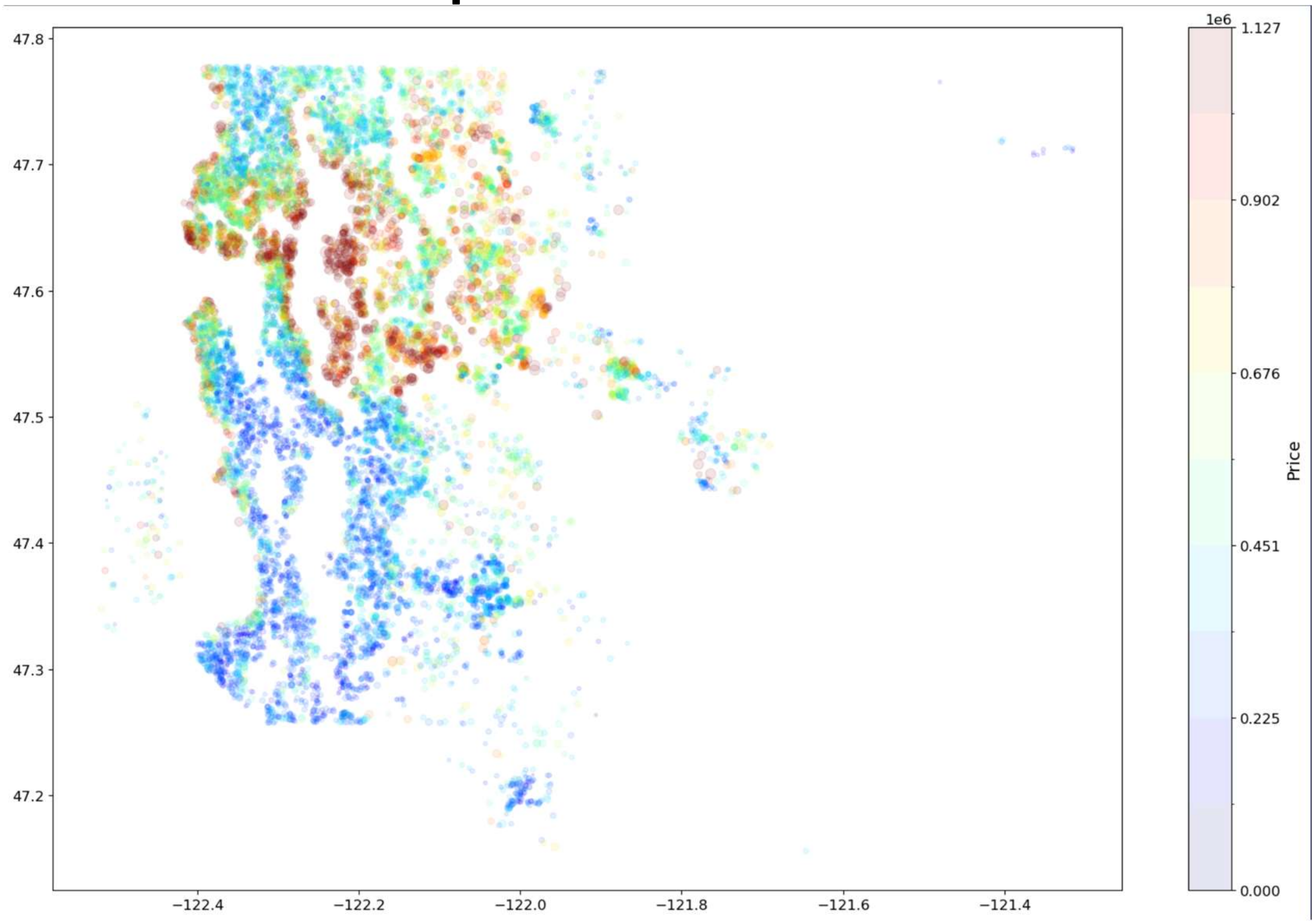


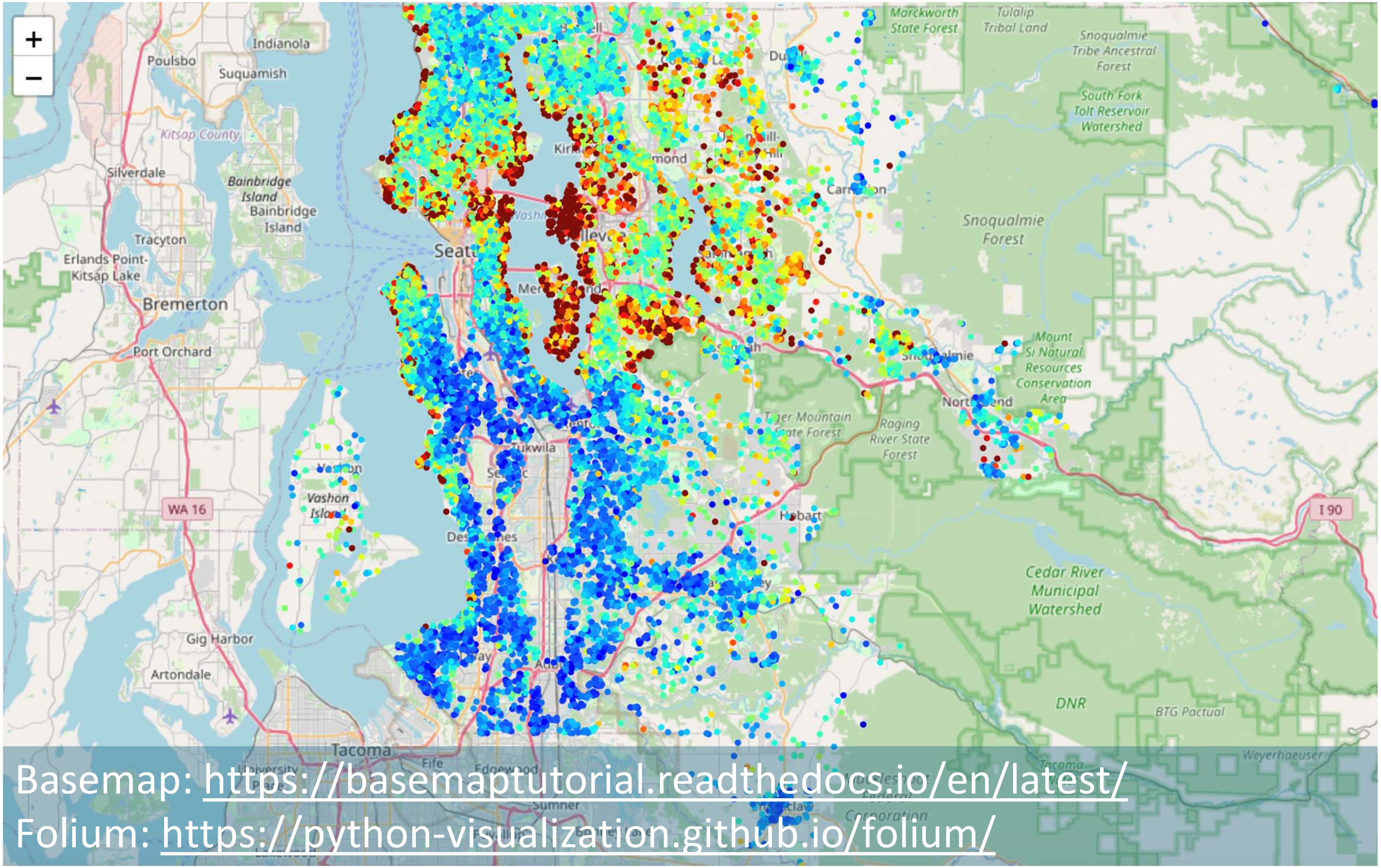
sns.pairplot

- We can get identify the distribution of data
- Linear relationships between variables



Example: colormap





Basemap: https://basemaptutorial.readthedocs.io/en/latest/

Folium: https://python-visualization.github.io/folium/

1. Frame the Problem

- We want to be able to **predict the price of houses** in King County, Washington, US.

- Questions for you to consider:
  - Is it supervised, unsupervised, or reinforcement learning?
  - Is it a classification task, a regression task or something else?
  - Should you use batch learning or online learning techniques?