CHƯƠNG 2:
DỰ ÁN MÁY HỌC

Khoa Khoa học và Kỹ thuật thông tin
Bộ môn Khoa học dữ liệu

NỘI DUNG

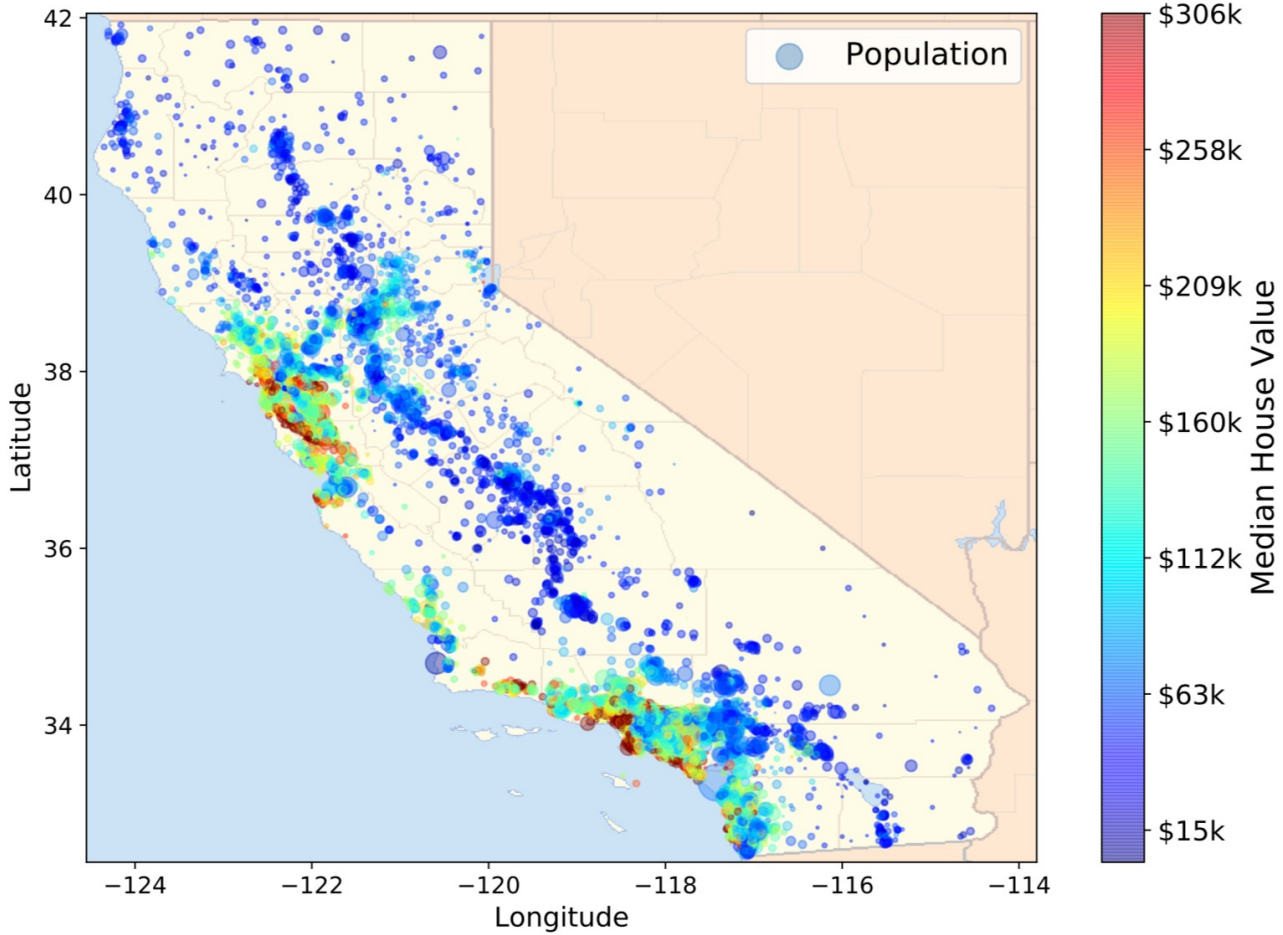
1. Xác định bối cảnh và định nghĩa bài toán.
2. Thu thập dữ liệu.
3. Khám phá dữ liệu.
4. Chuẩn bị dữ liệu.
5. Huấn luyện mô hình.
6. Tinh chỉnh mô hình.
7. Đưa ra giải pháp.
8. Vận hành, theo dõi, và bảo trì hệ thống

Làm việc với dữ liệu thực tế

Khi học máy học nên sử dụng dữ liệu thực tế, không nên dùng dữ liệu nhân tạo. Một vài nguồn cung cấp dữ liệu *miễn phí*:

- Một số kho dữ liệu mở phổ biến
    - UC Irvine Machine Learning Repository
    - Kaggle datasets
    - Amazon's AWS datasets
- Một vài siêu cổng dữ liệu (liệt kê các kho dữ liệu mở khác)
    - Data Portals
    - OpenDataMonitor
    - Quandl
- Một số trang khác liệt kê các kho dữ liệu mở
    - Wikipedia's list of Machine Learning datasets
    - Quora.com
    - The datasets subreddit

Dữ liệu sử dụng trong buổi học: *California Housing Prices dataset* [1]



[1] Pace and Barry. “Sparse spatial autoregressions”. 1997.

1. XÁC ĐỊNH BỐI CẢNH VÀ ĐỊNH NGHĨA BÀI TOÁN

Xác định bài toán

- **Mục tiêu của dự án máy học**: Xây dựng mô hình dự đoán **giá nhà** tại bang California dựa trên **dữ liệu điều tra dân số** của bang này.
- Dữ liệu này có các số liệu (thuộc tính) sau: **dân số**, **thu nhập trung bình**, **giá nhà trung bình**, v.v... cho mỗi *khu vực [2]* tại bang California.
- Bài toán máy học:
  - Đầu vào: khu vực bất kỳ và các dữ kiện khác đi kèm.
  - Đầu ra: **giá nhà trung bình** của khu vực đó.

[2] Khu vực: đơn vị địa lí nhỏ nhất theo Cục điều tra dân số Hoa Kỳ. Mỗi khu vực thường có dân số từ 600 đến 3,000 người.

Bối cảnh sử dụng

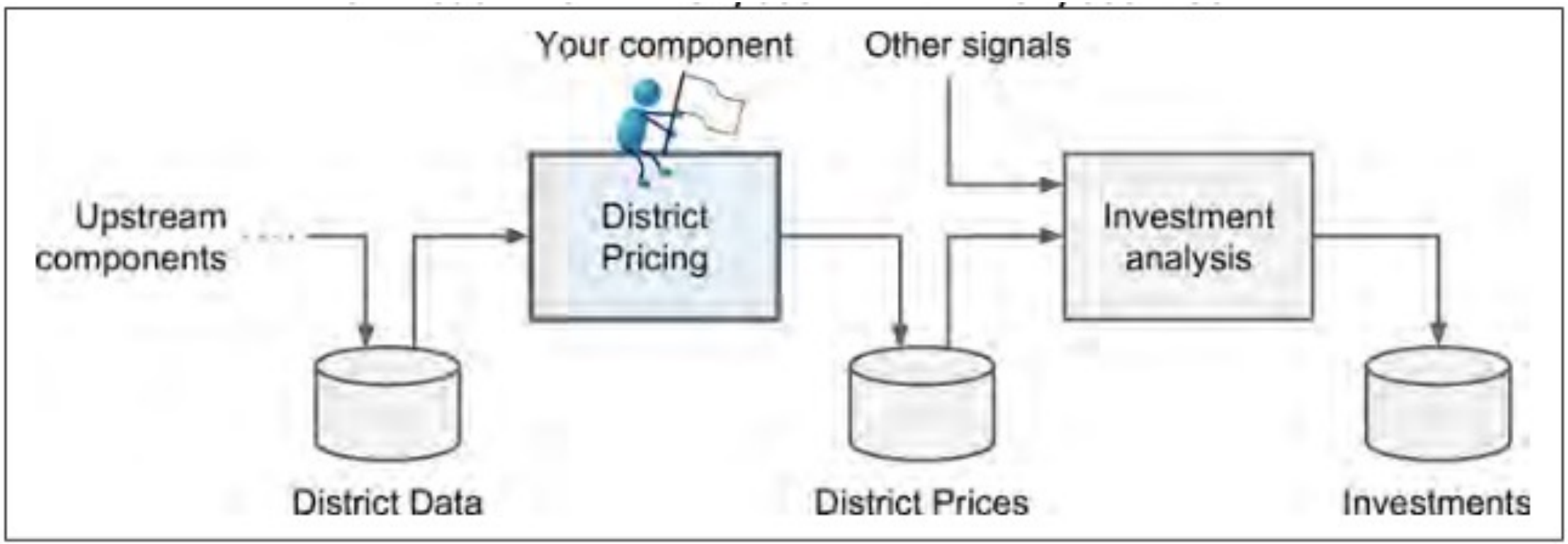
- Hệ thống này sẽ được dùng như thế nào?
- Đầu ra sẽ thế nào?
- Phương pháp thực hiện là gì ?
- Hệ thống tin cậy tới mức nào ?

Bối cảnh sử dụng

- Hệ thống này sẽ được dùng như thế nào?
  + Trả lời: dùng để đưa ra dự đoán giá nhà trung bình dựa trên các dữ kiện đầu vào ➔ **hỗ trợ** cho các nhà đầu tư.
- Đầu ra sẽ thế nào?
  + Trả lời: Đầu ra là một giá trị thể hiện giá nhà trung bình.
- Phương pháp thực hiện là gì ?
  + Trả lời: **Phương pháp hồi quy (regression).**
- Hệ thống tin cậy tới mức nào ?
  + Trả lời: Sẽ chứng minh dựa trên dữ liệu kiểm thử bằng một độ đo cụ thể là RMSE.

  → ***Mỗi bài toán phải có ít nhất độ đo đánh giá***.

Pipeline for ML process



Kiểm tra các giả định

- Các bạn (người phát triển hệ thống máy học) có những giả định gì trong đầu, phải làm sáng tỏ trước khi bắt tay xây dựng hệ thống.
- Ví dụ: Bạn nghĩ là đây nên là bài toán hồi quy nhưng hệ thống sử dụng kết quả của bạn lại chuyển đổi giá nhà dự đoán sang các giá trị phân loại (“rẻ”, “vừa”, hoặc “đắt”) thay vì giá nhà cụ thể. Như vậy, bài toán lúc này nên định hình thành phân lớp (classification) hơn là hồi quy (regression).
- Nếu hệ thống sau đó cần giá trị cụ thể thì bài toán hồi quy là phù hợp.

2. THU THẬP DỮ LIỆU

MỤC TIÊU

- Thu thập dữ liệu phục vụ cho việc huấn luyện và kiểm tra mô hình máy học.
- Yêu cầu:
  + Nguồn dữ liệu có tin cậy hay không?
  + Dữ liệu phù hợp với bài toán:
    - Miền (domain) của dữ liệu.
    - Biến target của dữ liệu (đối với bài toán học có giám sát) --> **Nhãn**.

Ví dụ

- Bài toán: Xây dựng mô hình dự đoán giá nhà tại bang California dựa trên dữ liệu điều tra dân số của bang này.
- Bộ dữ liệu: **California Housing**
- Nguồn tải: https://github.com/ageron/handson-ml/tree/master/datasets/housing
- Định dạng: CSV.
- Publication: Pace, R. K., & Barry, R. (1997). Sparse spatial autoregressions. *Statistics & Probability Letters*, *33*(3), 291-297.

Ý nghĩa từng thuộc tính

1. longitude: kinh độ.
2. latitude: vĩ độ.
3. housingMedianAge: tuổi của ngôi nhà.
4. totalRooms: Tổng số phòng.
5. totalBedrooms: Số phòng ngủ.
6. population: tổng dân số cư trú.
7. households: tổng số hộ gia đình.
8. medianIncome: Thu nhập trung bình của từng hộ gia đình.
9. **medianHouseValue: Giá nhà.**
10. ocean_proximity: Vị trí nhà (trong đất liền, giáp biển, giáp vịnh, trên đảo).

3. KHÁM PHÁ DỮ LIỆU

Thông tin tổng quan

Phân tích trực quan

→ Mục đích để làm gì?

Thông tin tổng quan

- Hiểu và có cái nhìn tổng quan về bộ dữ liệu.
- Các thông tin cần phải nắm rõ:
  + Kích thước dữ liệu: số thuộc tính, số dòng dữ liệu.
  + Thuộc tính nào là thuộc tính dự đoán?
  + Đặc điểm của giá trị của các thuộc tính:
    - Loại giá trị: *numerical, categorical, text, character, ...*
    - Đặc điểm của giá trị: *phân bố, miền giá trị, độ dài, ...*

Tổng quan dữ liệu

- Bộ dữ liệu: **California Housing.**
- Số lượng thuộc tính: 10.
- Số lượng dữ liệu: 20,640.

Xem thông tin 10 dòng dữ liệu đầu tiên

– Sử dụng lệnh head():

```
data.head(10)
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	-122.23	37.88	41.0	880.0	129.0	322.0	126.0	8.3252	452600.0
1	-122.22	37.86	21.0	7099.0	1106.0	2401.0	1138.0	8.3014	358500.0
2	-122.24	37.85	52.0	1467.0	190.0	496.0	177.0	7.2574	352100.0
3	-122.25	37.85	52.0	1274.0	235.0	558.0	219.0	5.6431	341300.0
4	-122.25	37.85	52.0	1627.0	280.0	565.0	259.0	3.8462	342200.0
5	-122.25	37.85	52.0	919.0	213.0	413.0	193.0	4.0368	269700.0
6	-122.25	37.84	52.0	2535.0	489.0	1094.0	514.0	3.6591	299200.0
7	-122.25	37.84	52.0	3104.0	687.0	1157.0	647.0	3.1200	241400.0
8	-122.26	37.84	42.0	2555.0	665.0	1206.0	595.0	2.0804	226700.0
9	-122.25	37.84	52.0	3549.0	707.0	1551.0	714.0	3.6912	261100.0

Xem thông tin các thuộc tính

– Sử dụng lệnh: info()

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
```

Thông số cho các thuộc tính số

- Có 2 dạng giá trị đối với dữ liệu:
  + Giá trị categorical (tạm dịch: giá trị phân loại)
  + Giá trị số (numerical).
- Các thông số cần quan tâm đối với thuộc tính số:
  + Giá trị trung bình (mean).
  + Giá trị lớn nhất (max).
  + Giá trị nhỏ nhất (min).
  + Giá trị độ lệch chuẩn (std).
  + Tứ phân vị (quantile).

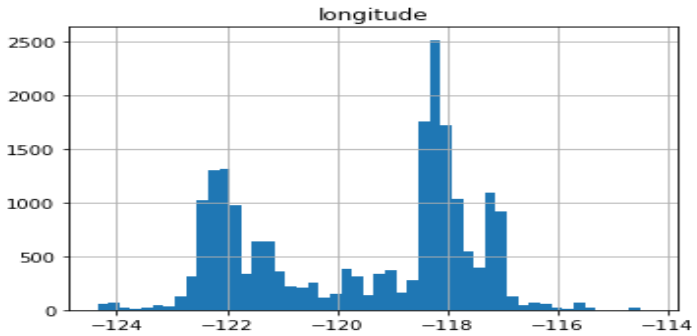
Xem thông tin đối với các thuộc tính số

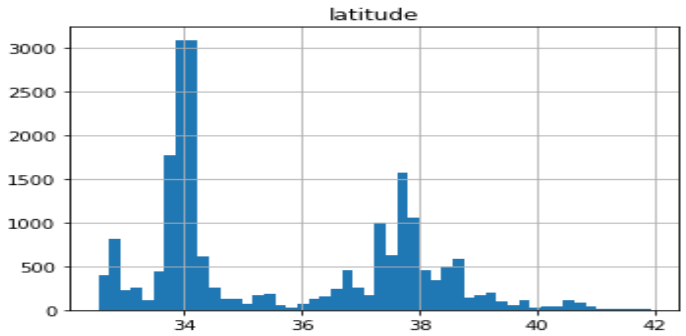
– Sử dụng hàm describe()

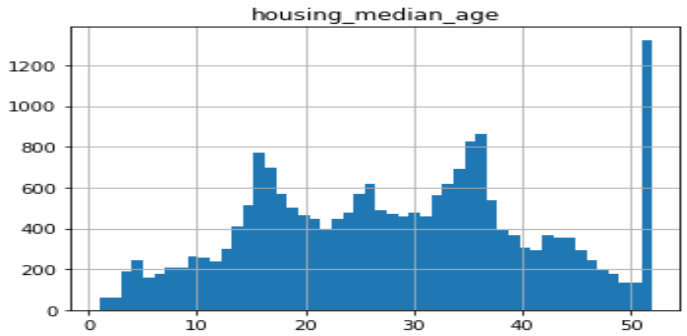
```
data.describe()
```

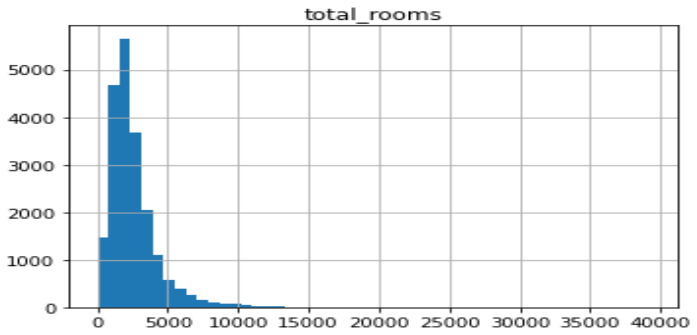
	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income
count	20640.000000	20640.000000	20640.000000	20640.000000	20433.000000	20640.000000	20640.000000	20640.000000
mean	-119.569704	35.631861	28.639486	2635.763081	537.870553	1425.476744	499.539680	3.870671
std	2.003532	2.135952	12.585558	2181.615252	421.385070	1132.462122	382.329753	1.899822
min	-124.350000	32.540000	1.000000	2.000000	1.000000	3.000000	1.000000	0.499900
25%	-121.800000	33.930000	18.000000	1447.750000	296.000000	787.000000	280.000000	2.563400
50%	-118.490000	34.260000	29.000000	2127.000000	435.000000	1166.000000	409.000000	3.534800
75%	-118.010000	37.710000	37.000000	3148.000000	647.000000	1725.000000	605.000000	4.743250
max	-114.310000	41.950000	52.000000	39320.000000	6445.000000	35682.000000	6082.000000	15.000100

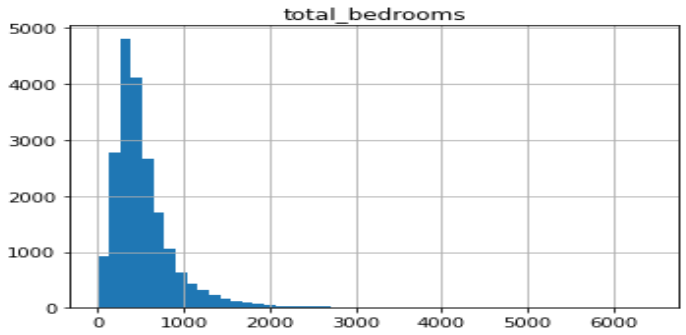
Trực quan hoá dữ liệu đơn giản

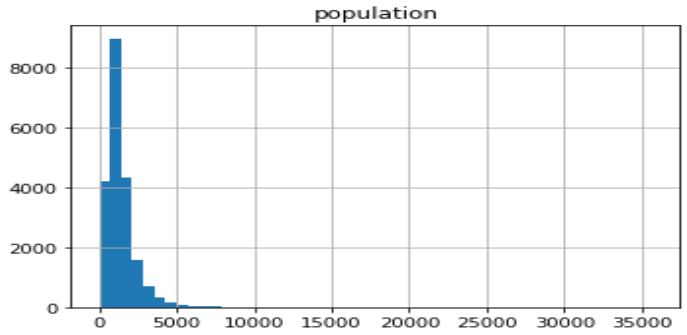


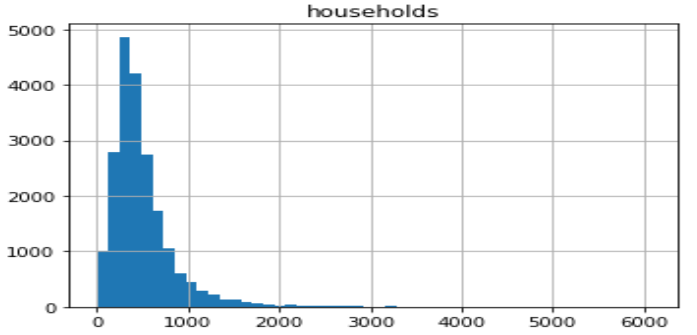


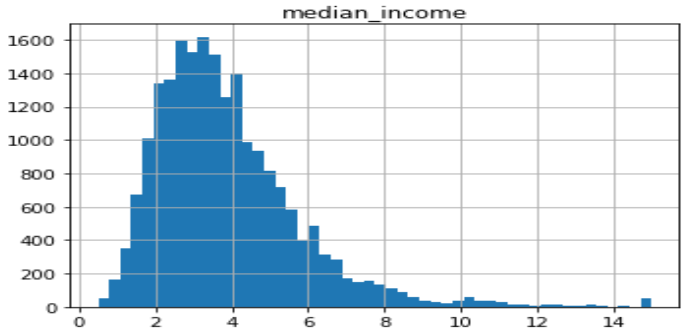


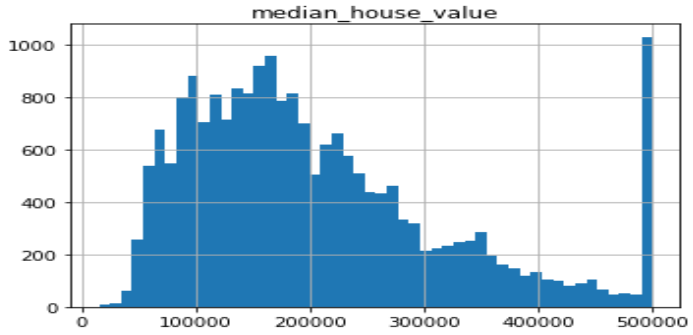












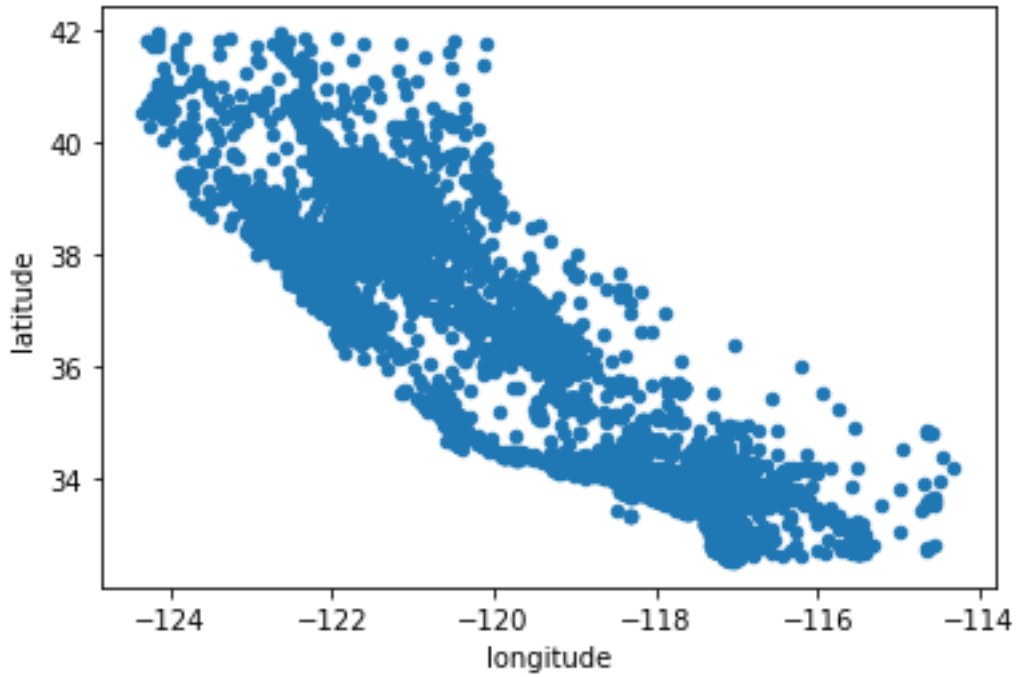
Một số nhận xét

- Phân bố dữ liệu: các thuộc tính có dạng phân bố chuẩn (dạng hình chuông) hay không?
- Các thuộc tính nào bị lệch? Lệch về phía nào?
- Khoảng giá trị (scale) của các thuộc tính như thế nào? Có phải các thuộc tính đều có cùng khoảng giá trị?
- Có dữ liệu dạng NA hay không?

Lưu ý: Sinh viên dựa vào thông tin về các thuộc tính số và đồ thị trực quan để trả lời các câu hỏi trên.

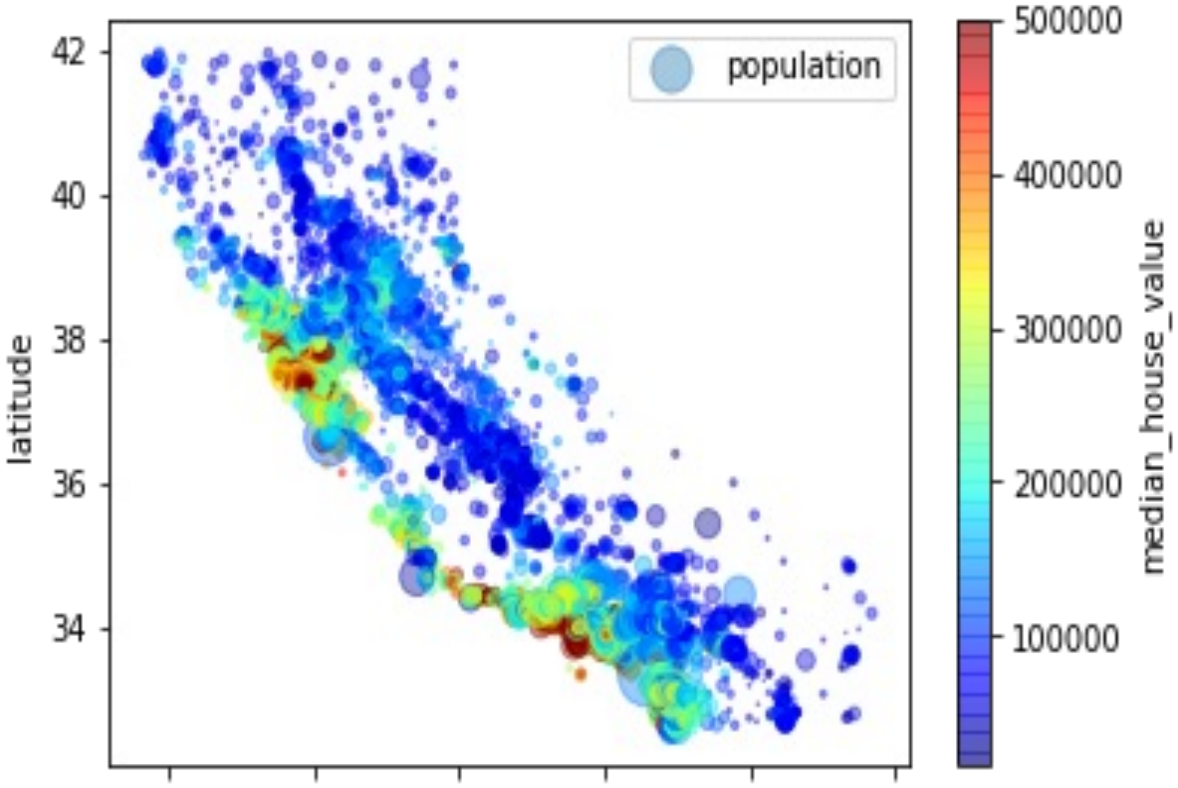
Trực quan hoá dữ liệu địa lý

Trực quan hoá từ dữ liệu



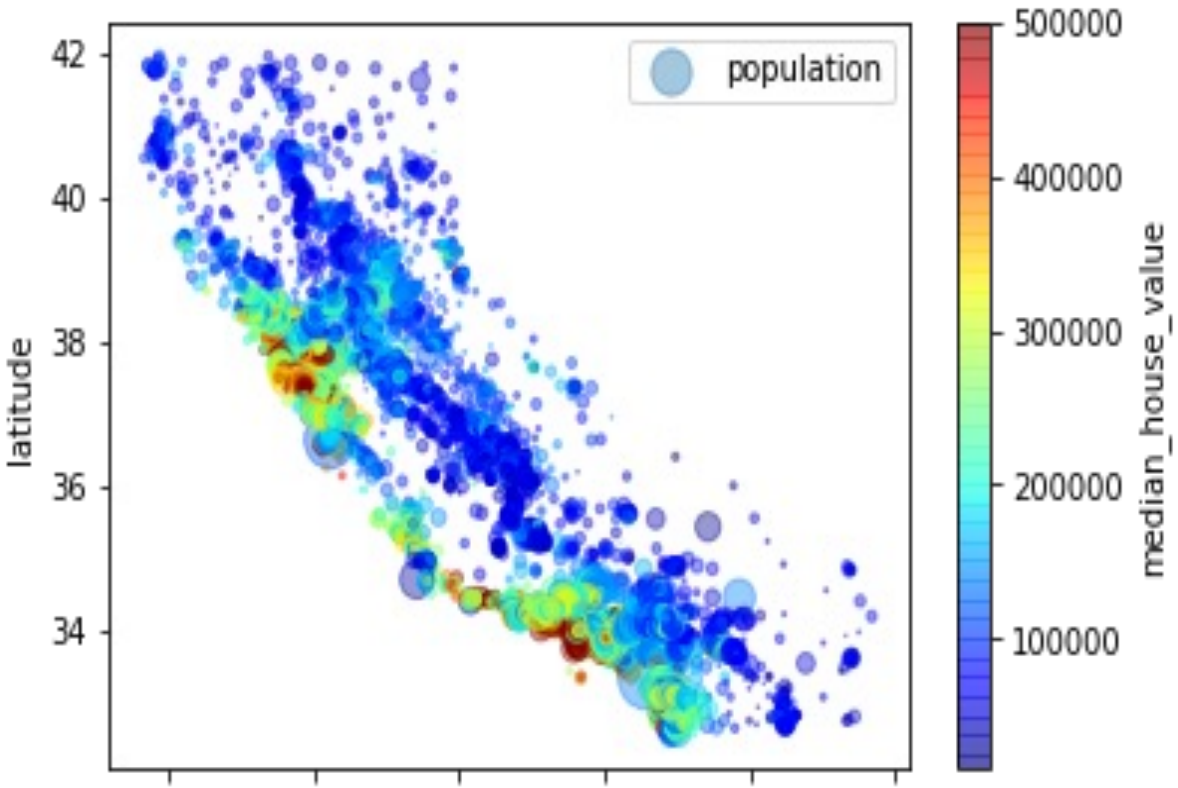
Dữ liệu thực tế của bang California

Trực quan hoá dữ liệu giá nhà theo khu vực



1. Giá nhà có phụ thuộc vào vị trí không?
2. Vị trí nào giá nhà đắt nhất ?
3. Giá nhà và dân số có mối quan hệ nào không?

Kết quả phân tích



1. Giá nhà có phụ thuộc vào vị trí không?
   + Vị trí nhà ảnh hưởng tới giá nhà.
2. Vị trí nào giá nhà đắt nhất ?
   + Gần biển và trên đảo.
3. Giá nhà và dân số có mối quan hệ nào không?
   + Khu vực đông dân thì giá nhà cao.

4. CHUẨN BỊ DỮ LIỆU

Giới thiệu

- Chuẩn bị dữ liệu còn gọi là tiền xử lý dữ liệu
- Các thao tác chuẩn bị dữ liệu:
  - Làm sạch dữ liệu (data cleaning)
  - Xử lý thuộc tính dạng văn bản và loại rời rạc
  - Co giãn thuộc tính (feature scaling)
  - Xử lý tùy biến
- Tất cả những thao tác xử lý dữ liệu ở bước này phải viết thành chương trình để tự động hóa về sau.
- Những hàm xử lý dữ liệu này có thể tập hợp thành thư viện để tái sử dụng.
- Có thể kết hợp các thao tác xử lý dữ liệu thành 1 pipeline (TH-t70).

Làm sạch dữ liệu

- Xử lý những thuộc tính thiếu giá trị (VD: thuộc tính total_bedrooms (TH-t63). Có 3 cách:
  - Bỏ những mẫu dữ liệu thiếu giá trị
  - Bỏ thuộc tính
  - Điền giá trị cho những mẫu bị thiếu giá trị (điền 0 hoặc giá trị trung bình hoặc giá trị trung vị,...)

Xử lý thuộc tính dạng văn bản và loại rời rạc

- Một số thuộc tính có giá trị dạng chuỗi như:
- Hầu hết thuật toán máy học chỉ làm việc tốt với thuộc tính dạng số, do đó phải chuyển chuỗi thành số (TH-t66,67).

```
>>> housing_cat = housing[["ocean_proximity"]]
>>> housing_cat.head(10)
       ocean_proximity
17606        <1H OCEAN
18632        <1H OCEAN
14650       NEAR OCEAN
3230            INLAND
3555         <1H OCEAN
19480           INLAND
8879         <1H OCEAN
13685           INLAND
4937         <1H OCEAN
4861         <1H OCEAN
```

Co giãn thuộc tính

- Các thuật toán máy học thường không hoạt động tốt khi thuộc tính có khoảng giá trị quá khác biệt.
- Co giãn thuộc tính giúp giải quyết vấn đề. Có 2 cách:
  - **Co giãn min-max** (normalization): Chuyển giá trị về trong khoảng [0,1] (Dùng MinMaxScaler của Scikit-Learn).
  - **Chuẩn hóa**: Đầu tiên nó trừ giá trị thuộc tính cho giá trị trung bình, sau đó chia cho độ lệch chuẩn để có phân phối có phương sai đơn vị. (Dùng StandardScaler của Scikit-Learn).

Xử lý tùy biến

– Xem TH-t68

Dữ liệu/Bộ dữ liệu chưa có?

????

5. HUẤN LUYỆN MÔ HÌNH

MỤC TIÊU

- Huấn luyện mô hình dự đoán cho bài toán.
- Các công việc cần làm:
  + Xác định loại bài toán: hồi quy, phân lớp.
  + Chọn mô hình phù hợp cho từng loại bài toán.
  + Phân chia dữ liệu huấn luyện.
  + Kiểm tra hiệu năng của mô hình.

Bài toán dự đoán giá nhà

- Loại bài toán áp dụng: Hồi quy (regression).
- Mô hình áp dụng: Hồi quy tuyến tính (Linear regression).
  + Input: Các thông số về giá nhà.
  + Output: Giá nhà dự kiến.
- Chia dữ liệu thành 2 biến như sau:
  + Biến X: các thuộc tính trong bộ dữ liệu.
  + Biến y: thuộc tính đích (target) – thuộc tính median_house_value

Huấn luyện mô hình

- Huấn luyện mô hình với mô hình Linear Regression đánh giá mô hình bằng cách đo RMSE của mô hình Linear Regression.
- Lỗi cao cho thấy mô hình đang chưa khớp dữ liệu. Có 2 cách giải quyết:
  + Chọn mô hình khác tốt hơn
  + Thêm thuộc tính
  + Giảm ràng buộc mô hình (đối với mô hình được chính quy hóa)
- Huấn luyện thêm mô hình Decision Tree, so sánh với mô hình Linear Regression.
- Đánh giá bằng phương pháp thẩm định chéo sử dụng k phần dữ liệu (k-fold cross validation).
  + Huấn luyện trên k-1 folds dữ liệu và đánh giá trên 1 fold còn lại, sau đó tính giá trị trung bình và độ lệch chuẩn để so sánh các mô hình.

6. TINH CHỈNH MÔ HÌNH

TINH CHỈNH MÔ HÌNH

- Tinh chỉnh mô hình (fine tuning) là phương pháp điều chỉnh giá trị của các siêu tham số (hyper parameter) của mô hình sao cho mô hình tối ưu nhất.
- Các phương pháp tinh chỉnh tham số cho mô hình:
  + Grid Search.
  + Randomized search.
  + Ensemble method.

Phân tích lỗi của mô hình tốt nhất

- Phân tích lỗi của những mô hình tốt nhất giúp cho chúng ta hiểu sâu thêm về bài toán: những điểm khó, ...
- Một số cách thực hiện:
  - Xem mức độ quan trọng của các thuộc tính
  - Quan sát chi tiết các lỗi sai của mô hình, suy nghĩ xem lý do còn lỗi và tìm cách cải tiến (thêm thuộc tính, bỏ những thuộc tính không hữu ích, bỏ các giá trị ngoại biên,...).

Đánh giá hệ thống trên tập dữ liệu thử nghiệm

- Sau khi lựa chọn được mô hình tốt nhất và chuẩn bị đưa vào sử dụng, bước cuối cùng là chạy cả pipeline của hệ thống để đánh giá hệ thống trên tập dữ liệu thử nghiệm.
- Kết quả thường sẽ thấp hơn khi chạy trên dữ liệu thử nghiệm, nhưng KHÔNG ĐƯỢC cải tiến gì nữa vì cải tiến lúc này không đảm bảo hệ thống sẽ chạy tốt trên dữ liệu mới hoàn toàn.

7. TRÌNH BÀY GIẢI PHÁP

Viết báo cáo, trình bày hệ thống

- Báo cáo phải nêu bật được:
  - Những khám phá của bạn (người xây dựng hệ thống máy học) sau quá trình xây dựng hệ thống.
  - Cái gì đã thử thấy hiệu quả và không hiệu quả.
  - Những giả định đã sử dụng trong hệ thống.
  - Những hạn chế của hệ thống.
- Viết tài liệu hướng dẫn cho tất cả mọi thứ, sử dụng nhiều hình ảnh, viết câu văn súc tích/dễ nhớ.
- Trong nhiều trường hợp, hệ thống máy học làm không tốt bằng các chuyên gia con người, tuy nhiên nó cũng giúp tiết kiệm thời gian để các chuyên gia có thể làm việc khác thú vị hơn.

8. Vận hành, theo dõi, và bảo trì hệ thống

Tổng kết

- Có tổng cộng 8 bước nhằm xây dựng.
- Đối với bất kỳ dự án hay đề tài nào về máy học, xác định được bài toán hay tác vụ đang làm là quan trọng nhất.
- Dữ liệu đóng vai trò quan trọng đối với một dự án máy học. Quá trình thu thập, xây dựng, tiền xử lý và khám phá dữ liệu chiếm phần lớn thời gian trong quá trình thực hiện.

Tài liệu tham khảo

- Chương 2 của sách: *Hands-on Machine Learning with ScikitLearn, Keras & TensorFlow.*