VIET NAM NATIONAL UNIVERSITY HO CHI MINH CITY

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHTN
TP. HỒ CHÍ MINH

Faculty of Information Technology

Môn học : Khoa Học Dữ Liệu
Lớp CQ2016/2 – Học kỳ I/2019-2020
GV: Trần Trung Kiên

BÁO CÁO ĐỒ ÁN MÔN HỌC CUỐI KỲ
-- ĐỀ TÀI DỰ ĐOÁN GIÁ NHÀ --

<Vấn đáp với giảng viên ngày 09-01-2020>

Nhóm 14: *Đặng Phương Nam – Lê Minh Nghĩa*

TP. Hồ Chí Minh, ngày 09 tháng 01 năm 2020

NỘI DUNG CHÍNH

Câu hỏi đặt ra?

Thu thập dữ liệu

Tiền xử lý dữ liệu

Mô hình hóa dữ liệu

Tổng kết

Cho các thông tin về một căn nhà
(đường, khu vực, diện tích, loại nhà, số phòng, tình hình thuế và tội phạm…)

Căn nhà này có giá là bao nhiêu tiền?

CÂU HỎI ĐẶT RA?

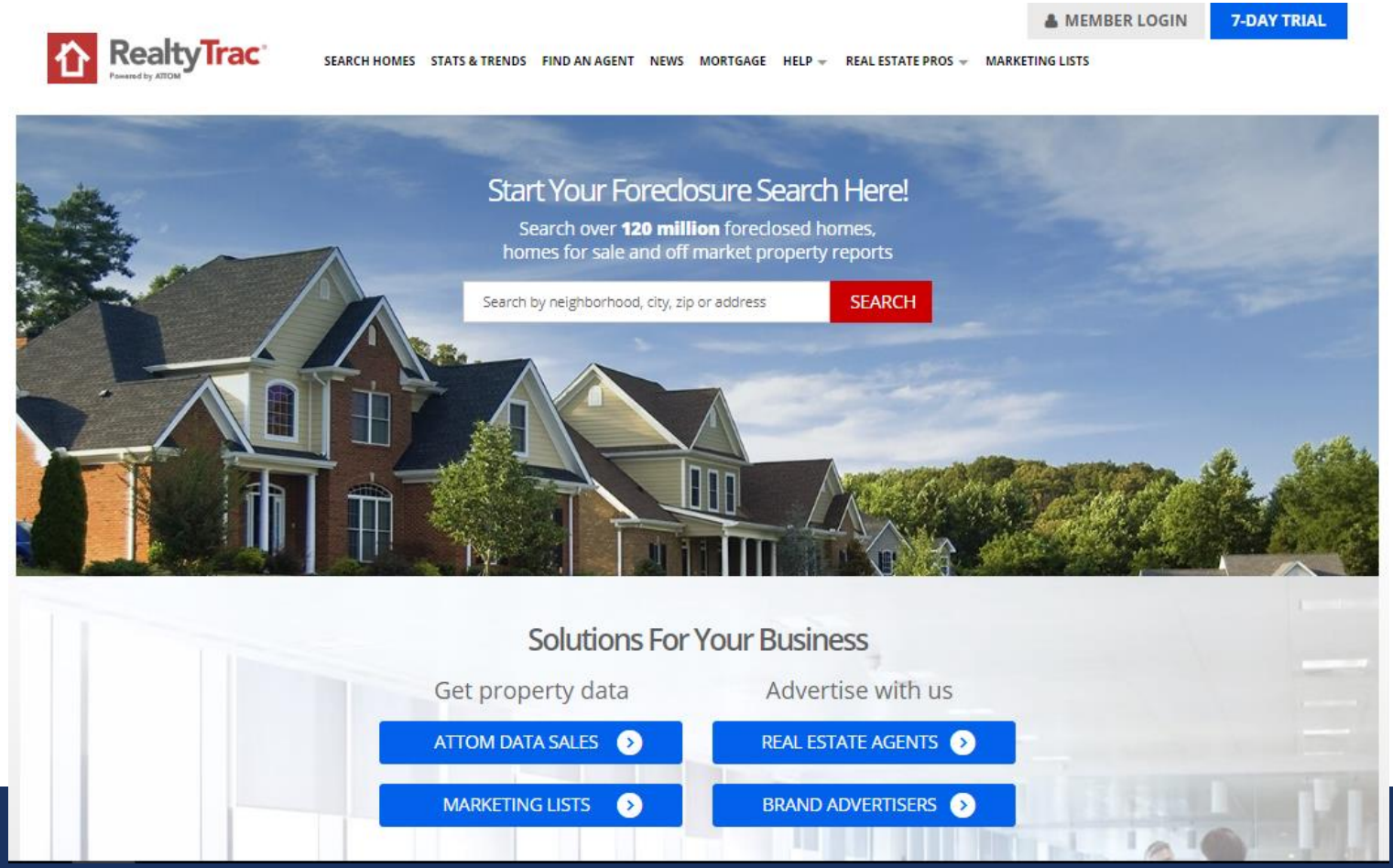
Người bán có thể dự đoán được giá trị căn nhà mà mình muốn bán.

Người mua có thể ước lượng được căn nhà mình muốn mua có giá cả hợp lý hay không?

Nguồn gốc câu hỏi: nhóm tự nghĩ ra.

LỢI ÍCH CỦA CÂU HỎI

Trang web thu thập dữ liệu: https://www.realtytrac.com/



THU THẬP DỮ LIỆU – PARSE HTML

Thu thập dữ liệu về danh sách "Các căn nhà đã được bán thành công tại quận Cam bang California của Mỹ"

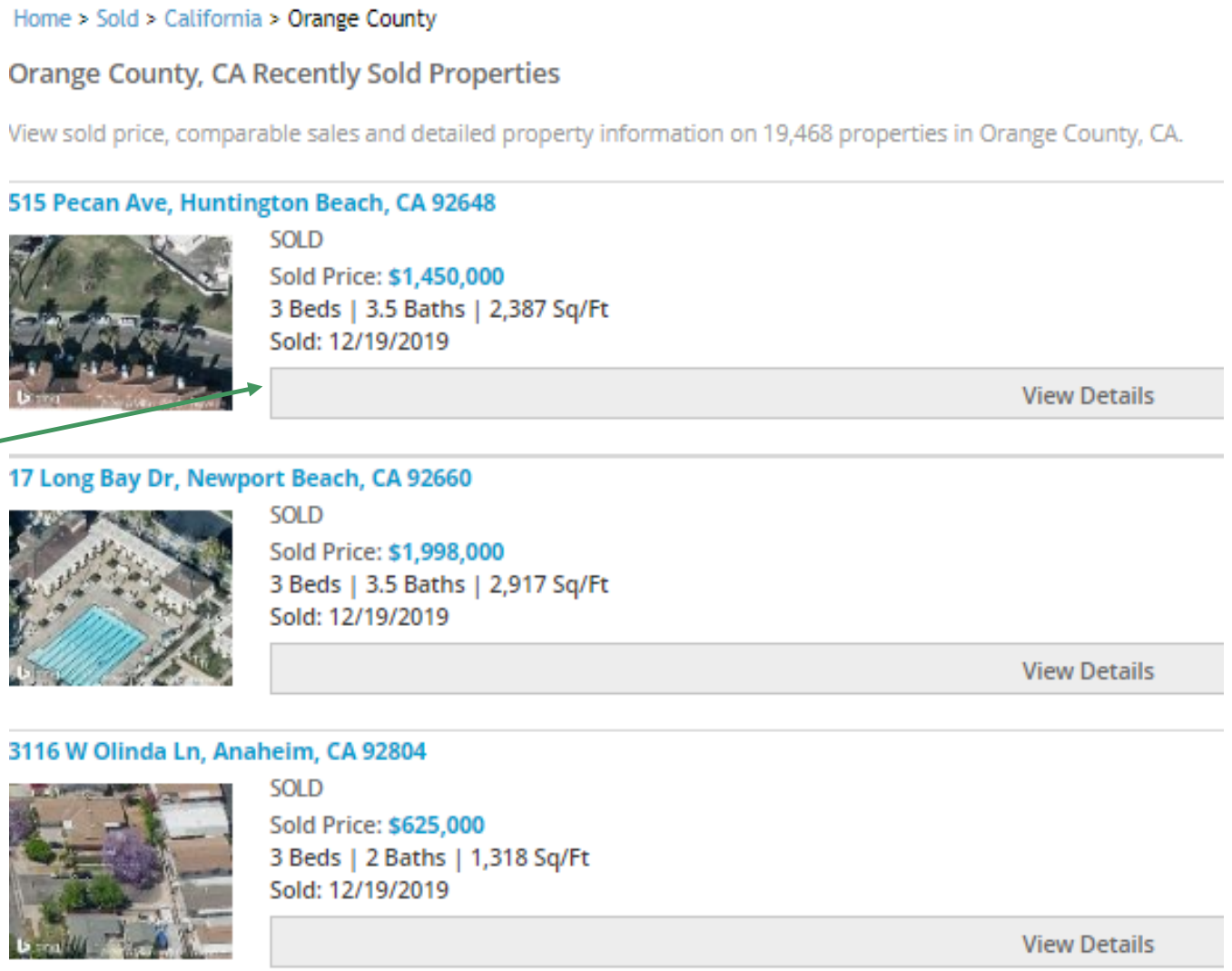
Lưu ý: Dữ liệu mà nhóm thu thập là hợp pháp (parse HTML có kiểm tra file robots.txt) và giá nhà là correct output (do căn nhà đã bán thành công).

Parse HTML để lấy:

+ Detail url.

Ở mỗi lần parse, nhóm đều thực kiểm tra việc lấy dữ liệu có hợp pháp hay không tại file robots.txt của trang web:

https://m.realtytrac.com/robots.txt



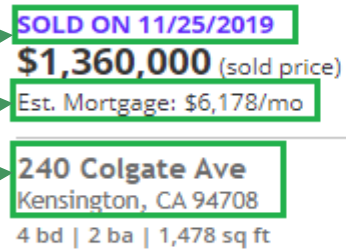
Ở mỗi bản tin bán nhà thành công, nhóm muốn lấy thêm một số thông tin chi tiết:

Parse HTML để lấy:

+ Date Sold.
+ Mortgage.
+ Address.
+ Description.
+ Property Infor:
  - Bedroom.
  - Bathroom.
  - Size.
  - ….

Ở mỗi lần parse, nhóm đều thực kiểm tra việc lấy dữ liệu có hợp pháp hay không tại file robots.txt của trang web:

https://m.realtytrac.com/robots.txt



8101 Barrington Dr is a single family residence located in La Mirada, CA 90638. Built in 1960, this property features 3 bedrooms, 3 bathrooms, 7,200 sq ft lot, and 1,627 sq ft of living space. This property recently sold for $605,000 on 12/19/2019.

For the surrounding community of La Mirada, CA 90638, the average sale price for similar homes to 8101 Barrington Dr is $628,885. The nearby schools are excellent and include Charles G Emery Elementary, Sunny Hills High and Buena Park Junior High. The overall crime risk for this area is low with 3 criminal anc sex offenders residing within 1 mile. The natural disaster risk for this area includes very high earthquake risk, low tornado risk, and minimal flood risk.

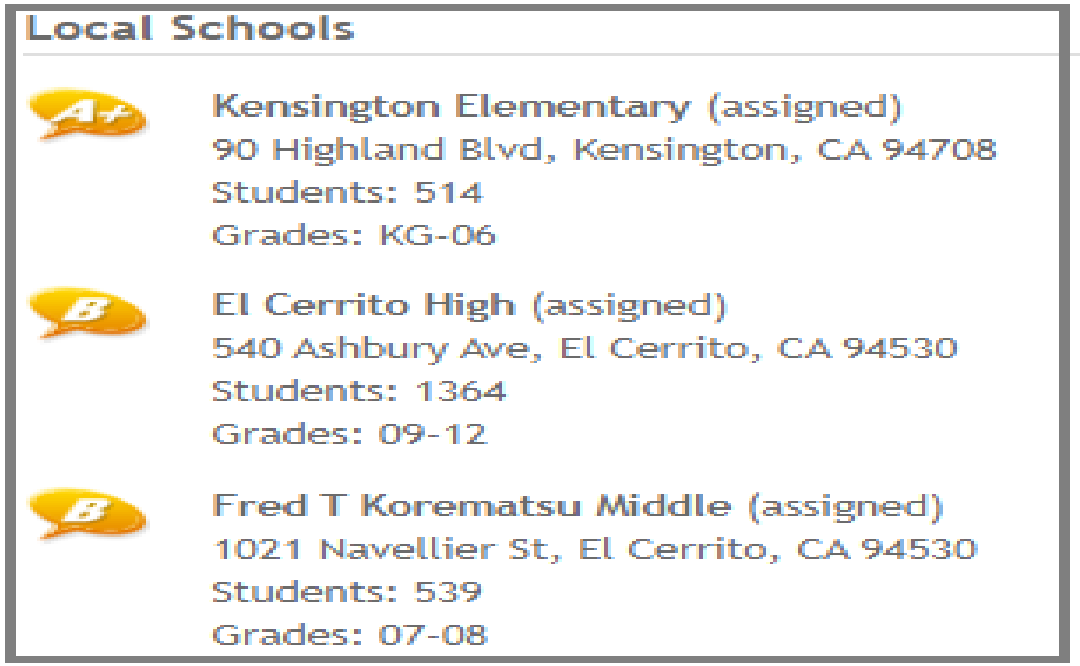
Property Info

Type	Single Family Residence
Bedrooms	4
Bathrooms	2
Size	1,478 sqft
Lot Size	5,000 sqft
Year Built	1938
Sold Price	$1,360,000
Property ID	1102358090
County	Contra Costa County, CA
Parcel Number	5701800079

Ở mỗi bản tin bán nhà thành công, nhóm muốn lấy thêm một số thông tin chi tiết:

Parse HTML để lấy:

+ Taxes.
+ Price History.
+ Number school.



Ở mỗi lần parse, nhóm đều thực kiểm tra việc lấy dữ liệu có hợp pháp hay không tại file robots.txt của trang web:

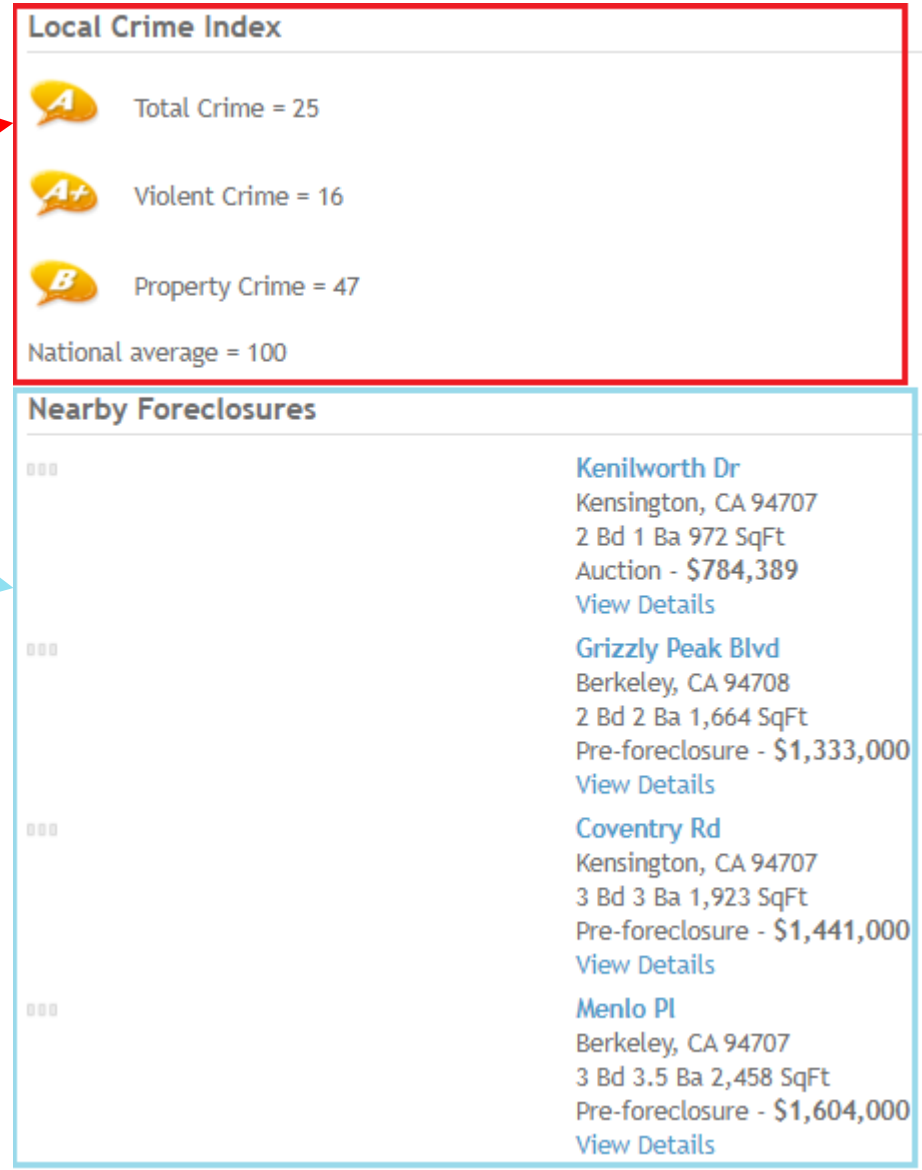
https://m.realtytrac.com/robots.txt

Ở mỗi bản tin bán nhà thành công, nhóm muốn lấy thêm một số thông tin chi tiết:

Parse HTML để lấy:

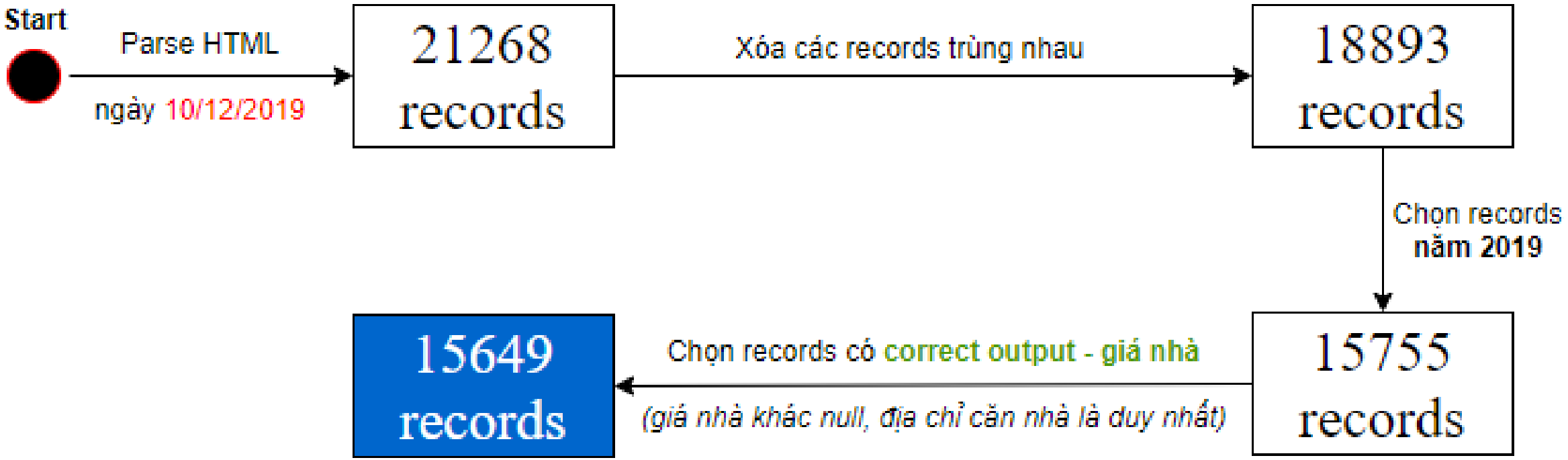
+ Info crime.

+ Number near foreclosures.



Ở mỗi lần parse HTML ở mỗi trang nhóm đều thực kiểm tra việc lấy dữ liệu có hợp pháp hay không tại file robots.txt của trang web:

https://m.realtytrac.com/robots.txt



THU THẬP DỮ LIỆU – LỰA CHỌN DỮ LIỆU NĂM 2019

Dữ liệu có 15649 dòng và 26 cột

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15649 entries, 0 to 15648
Data columns (total 26 columns):
address_street          15603 non-null object
address_locality        15649 non-null object
address_region          15649 non-null object
address_code            15649 non-null int64
date_sold               15649 non-null object
mortgage                15649 non-null float64
info_type               15649 non-null object
info_bedrooms           14222 non-null float64
info_bathrooms          14232 non-null float64
info_size               14937 non-null float64
info_lot_size           10605 non-null float64
info_year_built         14461 non-null float64
info_sold_price         15649 non-null float64
info_property_id        15649 non-null int64
info_county             15649 non-null object
info_parcel_number      15649 non-null int64
taxes_land              15408 non-null float64
taxes_improvements      15408 non-null float64
taxes_total             15408 non-null float64
taxes_taxes             15408 non-null object
school                  15649 non-null int64
total_crime             15166 non-null float64
violent_crime           15166 non-null float64
property_crime          15166 non-null float64
foreclosures            15649 non-null int64
year_sold               15649 non-null int64
dtypes: float64(13), int64(6), object(7)
memory usage: 3.1+ MB
```

1. address_street: tên đường.
2. address_locality: tên địa phương.
3. address_region: tên vùng.
4. address_code: mã bưu điện.
5. date_sold: ngày bán thành công ngôi nhà.
6. mortgage: định giá ngôi nhà cho thuê theo tháng trước khi bán đi.
7. info_type: loại nhà.
8. info_bedrooms: số lượng phòng ngủ.
9. info_bathrooms: số lượng phòng tắm.
10. info_size: kích thước ngôi nhà (sqft).
11. info_lot_size: kích thước lô đất (sqft).
12. info_year_build: năm ngôi nhà được xây dựng.
13. **info_sold_price: giá bán thành công.**
14. info_property_id: id căn nhà.
15. info_county: tên quận.
16. info_parcel_number: số bưu kiện của căn nhà.

17. taxes_land: tiền đất.
18. taxes_improvements: tiền các tiện ích có trên đất đó (hàng rào, đường đi từ cổng vô nhà, ...).
19. taxes_total: tổng của tiền land và tiền improvements.
20. taxes_taxes: tiền thuế phải đóng ứng với taxes_total.
21. school: số lượng trường học gần đó.
22. total_crime: tỉ lệ tội phạm (%) so với tội phạm cả nước.
23. violent_crime: tỉ lệ tội phạm bạo lực (%) so với tội phạm cả nước.
24. property_crime: tỉ lệ tội phạm về tài sản (%) so với tội phạm cả nước.
25. foreclosures: số lượng các căn nhà bị tịch thu gần đó.
26. year_sold: năm bán nhà.

Có 15 cột có giá trị thiếu là:

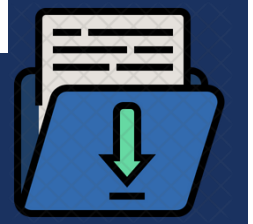
- address_street thiếu 46 giá trị.
- info_bedrooms thiếu 1427 giá trị.
- info_bathrooms thiếu 1417 giá trị.
- info_size thiếu 712 giá trị.
- info_lot_size thiếu 1334 giá trị.
- info_year_built thiếu 5044 giá trị.
- info_sold_price thiếu 934 giá trị.
- info_property_id thiếu 151 giá trị.
- taxes_land, taxes_improvements, taxes_total và taxes_taxes đều thiếu 241 giá trị.
- total_crime, violent_crime và property_crime đều thiếu 483 giá trị.

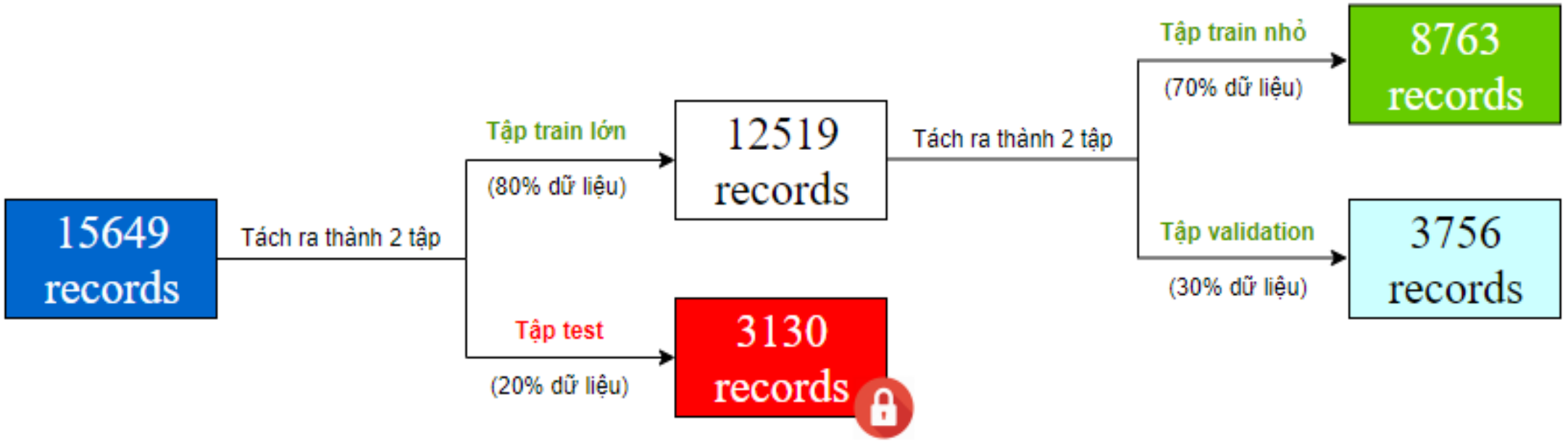
Lý do có nhiều giá trị thiếu tại các cột bên là do người muốn bán nhà lúc đăng tin không nhập đầy đủ các trường thông tin trên mà thường chỉ mô tả chi tiết tại phần description của bản tin hay vì lý do nào đó mà bản thân người bán cũng không biết rõ các thông tin trên.

Giải pháp: Sẽ bỏ đi các cột thực sự không cần thiết mà có nhiều giá trị thiếu. Và phần giá trị thiếu còn lại, nhóm sẽ tiếp tục thử tìm trong phần description của mỗi bản tin từ phần dữ liệu đã lấy được ban đầu. Nếu vẫn còn thiếu thì sẽ lắp đầy bằng các phương pháp như mean, mode, median, KNN.

	address_street	address_locality	address_region	address_code	date_sold	mortgage	info_type	info_bedrooms	info_bathrooms	info_size	...	taxes_lan
0	4 Silver Crk # 30	Irvine	CA	92603	11/22/2019	3543.0	Condominium	3.0	2.0	1576.0	...	211708.
1	28459 Alava	Mission Viejo	CA	92692	11/22/2019	2430.0	Single Family Residence	2.0	2.0	1503.0	...	281600.
2	19441 Hansen Ln	Huntington Beach	CA	92646	11/22/2019	6268.0	Single Family Residence	4.0	3.0	2926.0	...	459321.
3	23308 Copante # 103	Mission Viejo	CA	92692	11/22/2019	2158.0	Condominium	2.0	2.0	1308.0	...	326318.
4	1301 Burwood St	La Habra	CA	90631	11/22/2019	3816.0	Multi-Family Dwellings	NaN	NaN	3756.0	...	233089.
5	10281 Overhill Dr	Santa Ana	CA	92705	11/22/2019	7268.0	Single Family Residence	5.0	3.0	2814.0	...	525268.
6	52 Wild Horse	Irvine	CA	92602	11/22/2019	5669.0	Single Family Residence	4.0	3.0	2393.0	...	360108.
7	8597 Valley View St	Buena Park	CA	90620	11/22/2019	2839.0	Single Family Residence	3.0	2.0	1250.0	...	437818.
8	13887 La Jolla Plz	Garden Grove	CA	92844	11/22/2019	2226.0	Single Family Residence	3.0	3.0	1637.0	...	332109.
9	163 Jaripol Cir	Rancho Mission Viejo	CA	92694	11/22/2019	1881.0	Condominium	1.0	2.0	921.0	...	Nal

THU THẬP DỮ LIỆU – 10 DÒNG ĐẦU TIÊN CỦA DỮ LIỆU ĐƯỢC THU THẬP





Mọi thao tác tiền xử lý đều được thực hiện trên tập train nhỏ (8763 records), sau đó mới dùng các giá trị ước lượng từ tập train nhỏ để tiền xử lý tập validation (3756 records).

Cuối cùng, áp dụng toàn bộ thao tác tiền xử lý tìm được *(từ ở trên)* áp dụng tương tự cho tập train lớn và tập test (3130 records).

TIỀN XỬ LÝ DỮ LIỆU

Phần description thu thập được không giúp ích gì cho việc điền giá trị thiếu trong dữ liệu.

1. Cột address_street được ghi theo format:

   <số nhà> <hướng của đường> <tên đường> <chung cư (unit, #)>

   ví dụ: 2875 S Fairview St Unit B

   Nhóm chỉ lấy giá trị có nằm trong chung cư hay không? (1 là có, 0 là không). Xóa cột address_street.

2. Xóa 2 cột address_locality và cột address_region, giữ lại cột address_code.
3. Từ cột date_sold: rút trích time_sold (là month hay season) ra và xóa đi cột date_sold.
4. Để nguyên cột mortgage.
5. Cột info_type: sử dụng biến num_top_types để xử lý như BT03.
6. Giữ nguyên các cột info_bedroom, info_bathroom, info_size và info_lot_size.
7. Cột info_year_build vẫn giữ lại hay chuyển qua giai đoạn: <1900, [1990; 1950), [1950; 2000), >2000.
8. Xóa 3 cột info_property_id, info_county và info_parcel_number vì chúng khác biệt 100% và không có ý nghĩa khi huấn luyện.
9. Giữ nguyên hai cột taxes_land, taxes_improvements. Xóa cột taxes_total. Rút trích tiền từ taxes_taxes vì nó chứa cả thông tin dạng tiền và %, *ví dụ:* 6969 (1.23 %).
10. Giữ nguyên các cột school, foreclosures, total_crime, violent_crime và property_crime.
11. Xóa cột year_sold vì tất cả đều là 2019.

Các dạng cột:

- Cột dạng số (numerical), có 15 cột: apt_unit, mortgage, info_bedroom, info_bathroom, info_size, info_lot_size, taxes_land, taxes_improvements, taxes_taxes, school, foreclosures, total_crime, violent_crime và property_crime.

- Cột dạng chuỗi có giá trị rời rạc không thứ tự (categorical), có 3 cột: address_code, time_sold và info_type. Các cột này đưa vào huấn luyện sẽ sử dụng phương pháp one-hot để chuyển thành số.

Điền giá trị thiếu:

- Dùng KNN (n = 5) cho các cột: info_size và info_lot_size.
- Dùng most cho các cột: address_code, time_sold, info_type, info_bedroom và info_bathroom.
- Dùng mean cho các cột: mortgage, taxes_land, taxes_improvements và taxes_taxes.
- Dùng median cho các cột: school, foreclosures, total_crime, violent_crime và property_crime.

Các siêu tham số cần lưu ý:

- month_to_season: có chuyển tháng bán nhà thành bán nhà theo bốn mùa? (True/False).
- year_to_period: có chuyển thời gian xây nhà thành các khoảng thời gian mà nhóm cho trước? (True/False).
- num_top_types: số lượng top types như BT03 (số nguyên dương).

Các siêu tham số này cũng sẽ được học trong mô hình máy học.

Nhóm sử dụng 4 loại mô hình hồi quy để mô hình hóa dữ liệu:

- Linear Regression.
- K-Neighbors Regressor.
- Decision Tree Regressor.
- Random Forest Regressor.

- Công thức tính độ chính xác:

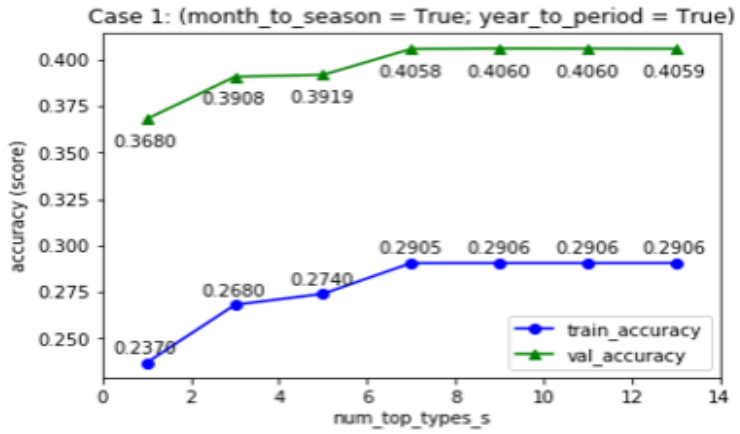
$$\text{score} = 1 - \frac{\sum_{i=1}^{n}(y\_true_i - y\_pred_i)^2}{\sum_{i=1}^{n}(y\_true_i - mean(y\_true))^2}$$

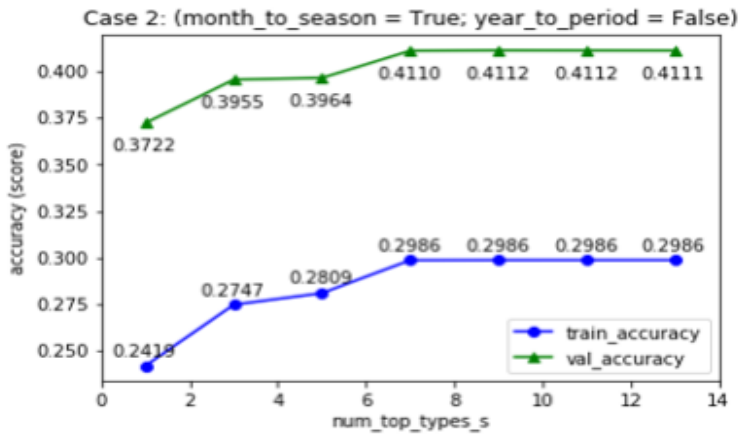
score có thể âm (<0), lúc này mô hình dự đoán có độ chính xác rất kém (không đáng tin cậy để sử dụng).

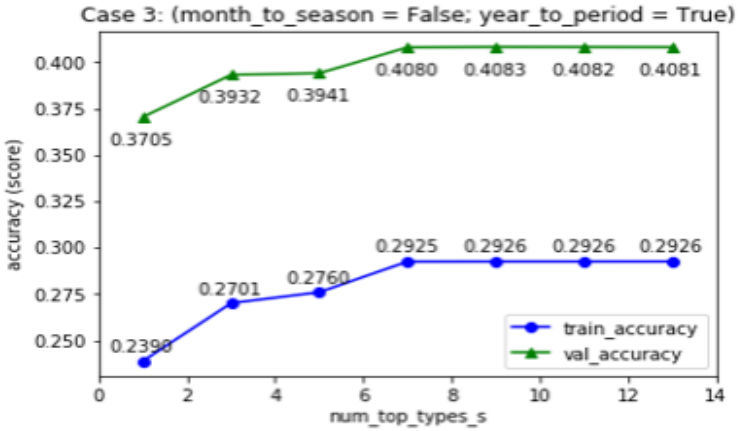
Điều bất thường là khi thử sử dụng mô hình Linear Regression *(month_to_seasons=True, year_to_period=True và num_top_types=5)* để dự đoán giá nhà thì thu được kết quả với độ chính xác (score) trên tập validation cực cao 0.9999.

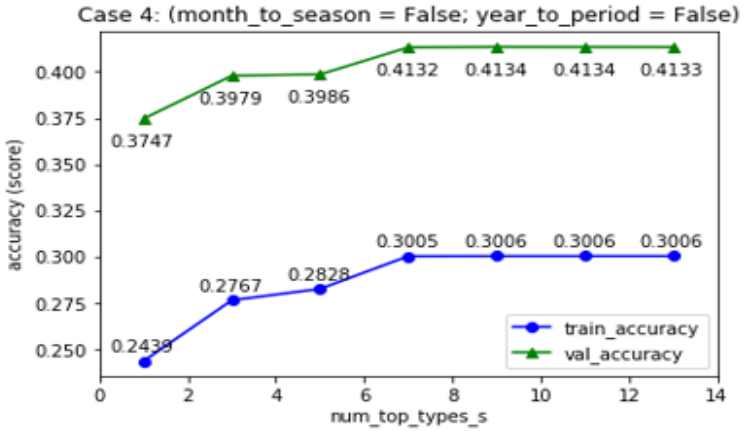
Phát hiện thấy vấn đề nằm ở cột mortgage, vì sau một vài lần thử thì thấy rằng: ta chỉ cần lấy mortgage nhân cho 220 thì sẽ ra được con số rất gần với giá nhà được bán thành công.

Bây giờ sẽ xóa đi cột mortgage và sử dụng lại mô hình Linear Regression *(month_to_seasons=True, year_to_period=True và num_top_types=5)* để dự đoán giá nhà thì thu được kết quả với độ chính xác (score) trên tập validation chỉ 0.3676.









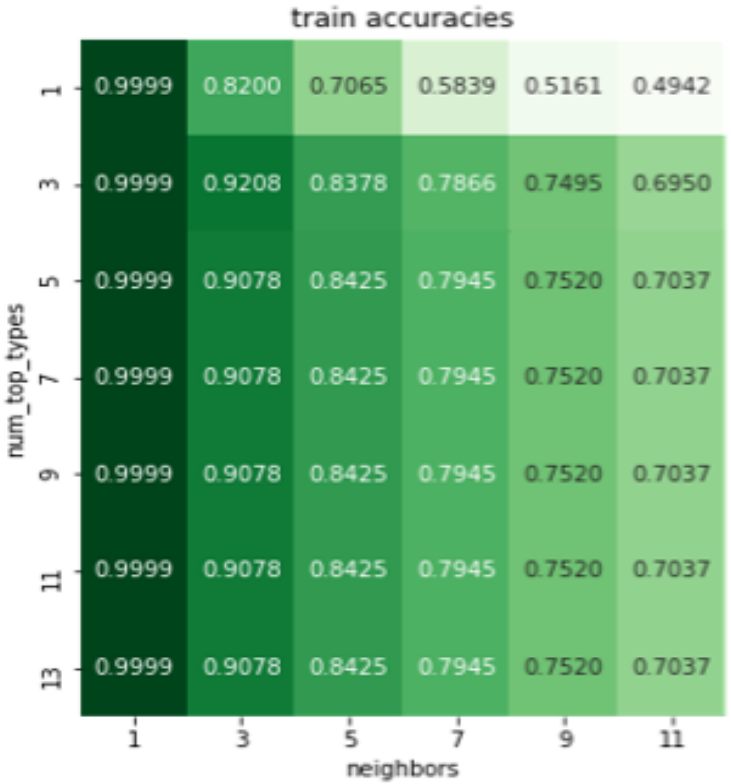
*Mô hình **Linear Regression** quá yếu trong tình huống này nên dự đoán bị underfitting, khi tăng chiều dữ liệu lên (đó là cho các siêu tham số bằng False, tăng số lượng num_top_types) thì thấy giúp học tốt hơn một chút, nhưng vẫn bị underfitting.*

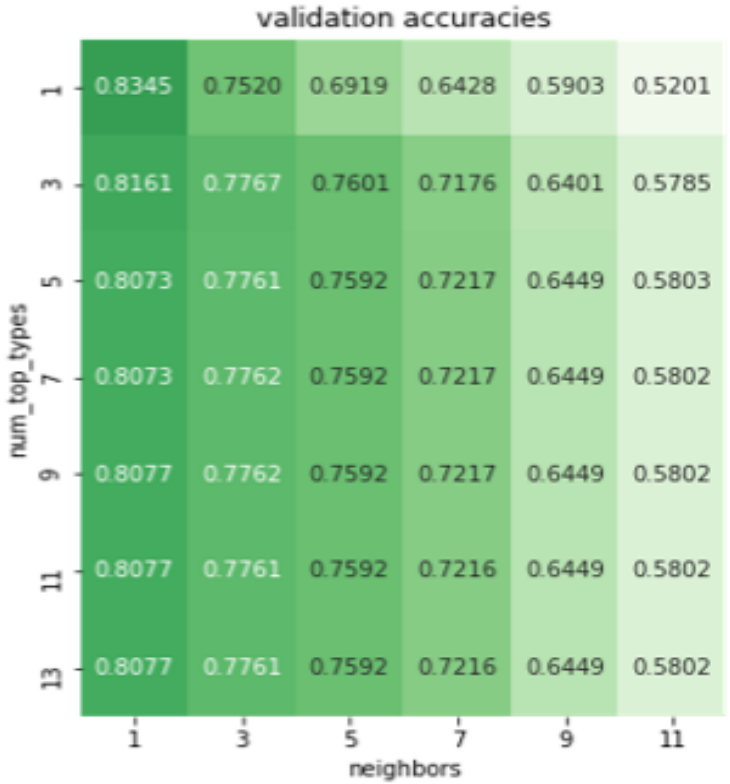
Độ chính xác *(score)* trên tập validation cao nhất là 0.4134, với month_to_season = False, year_to_period = False và num_top_types = 9.

MÔ HÌNH HÓA DỮ LIỆU – LINEAR REGRESSION

K-Neighbors Regressor

Case 1: month_to_seasons = True và year_to_period = True





- neighbors càng tăng thì khiến mô hình càng bị underfitting.

- num_top_types càng tăng thì càng giúp tang độ chính xác (score) trên cả 2 tập, nhưng đến một ngưỡng nào đó thì tăng thêm độ chính xác (score) trên 2 tập vẫn vậy.

Case 1: độ chính xác *(score)* trên tập validation cao nhất là 0.8345, với num_top_types = 1 và neighbors = 1

MÔ HÌNH HÓA DỮ LIỆU – K-NEIGHBORS REGRESSOR

Nhóm cũng tiến hành thử nghiệm cho 3 trường hợp còn lại với tập giá trị num_top_types = [1, 3, 5, 7, 9, 11, 13] và neighbors = [1, 3, 5, 7, 9 ,11]:

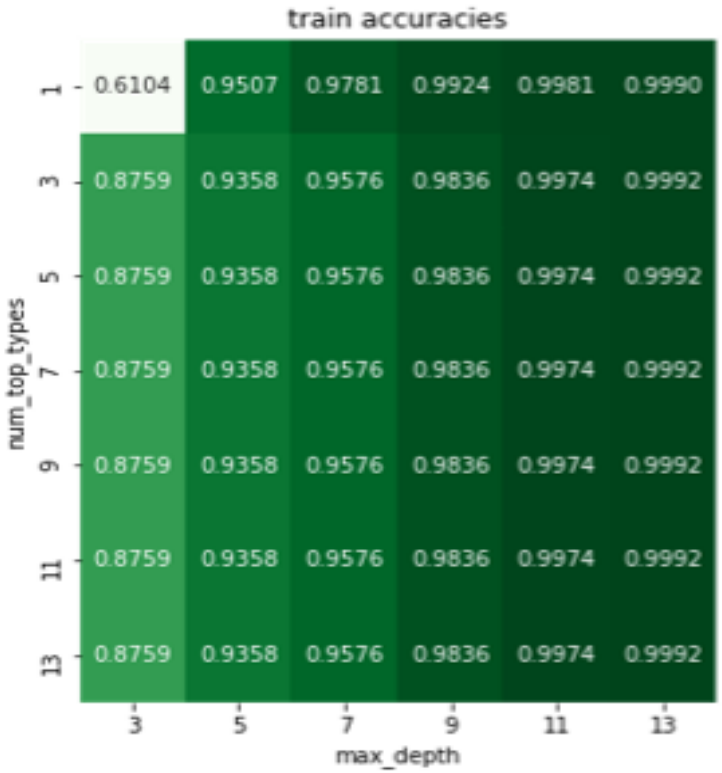
- Case 2: month_to_season = True và year_to_period = False, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.8353 ứng với num_top_types = 3 và neighbors = 1.
- Case 3: month_to_season = False và year_to_period = True, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.8429 ứng với num_top_types = 1 và neighbors = 1.
- Case 4: month_to_season = False và year_to_period = False, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.8353 ứng với num_top_types = 3 và neighbors = 1.

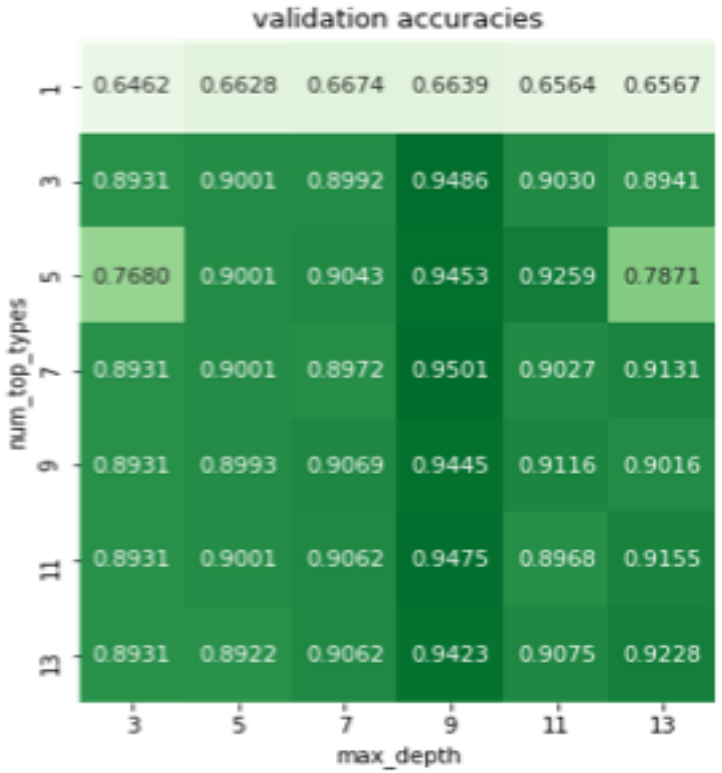
Trong trường hợp dùng **K-Neightbors Regressor** fit vào tập train và dự đoán trên tập validation thì thấy rằng: càng tăng neighbors thì càng khiến mô hình dễ bị underfitting do có quá nhiều láng giềng gây nhiễu ảnh hưởng đến kết quả.

Ghi nhận độ chính xác *(score)* cao nhất trên tập validation khi dùng mô hình **K-Neighbors Regressor** là 0.8429 ứng với các siêu tham số month_to_season = False, year_to_period = True, num_top_types = 1 và neighbors = 1.

Decision Tree Regressor

Case 1: month_to_seasons = True và year_to_period = True





- num_top_types bằng 1
→ max_depth quá nhỏ bị underfitting, về sau có tăng max_depth lên ở giá trị bao nhiêu thì cũng lại bị overfitting.

- num_top_types tăng dần lên:
→ càng tăng max_depth thì tỏ ra có hiệu quả nhưng tăng một hồi sẽ có dấu hiệu bị overfitting.

- num_top_types ban đầu tang thì cũng giúp tăng độ chính xác (score) trên cả 2 tập, một lúc sau thì thấy độ chính xác (score) trên tập validation bắt đầu có dấu hiệu giảm xuống.

Case 1: độ chính xác *(score)* trên tập validation cao nhất là 0.9501, với num_top_types = 7 và max_depth = 9

Nhóm cũng tiến hành thử nghiệm cho 3 trường hợp còn lại với tập giá trị num_top_types = [1, 3, 5, 7, 9, 11, 13] và neighbors = [1, 3, 5, 7, 9 ,11]:

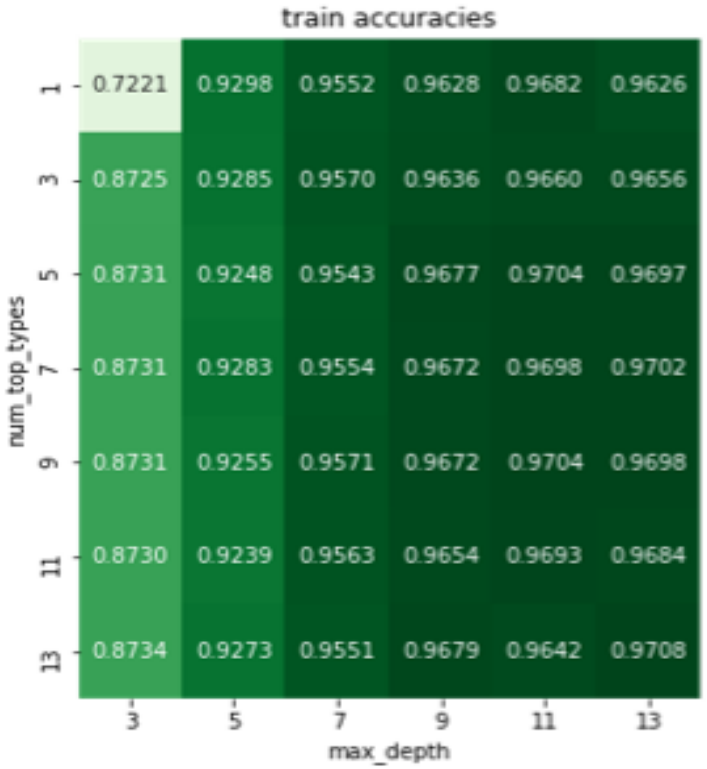
- Case 2: month_to_season = True và year_to_period = False, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.9396 ứng với num_top_types = 7 và max_depth = 11.
- Case 3: month_to_season = False và year_to_period = True, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.9486 ứng với num_top_types = 13 và max_depth = 9.
- Case 4: month_to_season = False và year_to_period = False, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.9363 ứng với num_top_types =7 và max_depth = 11.

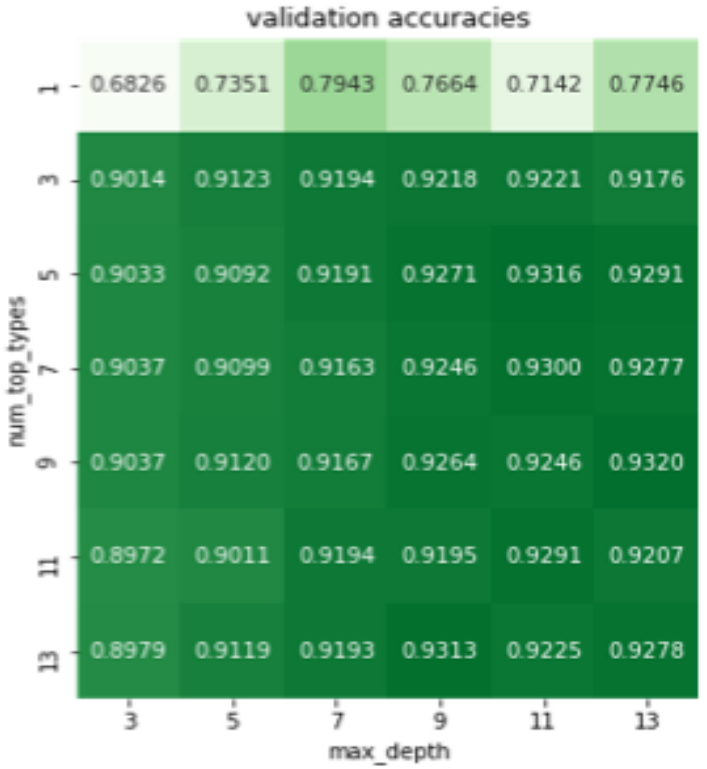
Trong trường hợp dùng **Decision Tree Regressor** để fit vào tập train và dự đoán trên tập validation thì thấy rằng: càng tăng max_depth thì mô hình càng fit vào dữ liệu do đó có thể dẫn đến trường hợp bị overfitting. Việc dùng num_top_types ban đầu tăng dần lên là có hiệu qua nhưng lúc sẽ có dấu hiệu mô hình quá fit với dữ liệu và độ chính xác *(score)* trên tập validation bắt đầu có dấu hiệu giảm.

Ghi nhận độ chính xác *(score)* cao nhất trên tập validation khi dùng mô hình **Decision Tree Regressor** là 0.9501 ứng với các siêu tham số month_to_season = True, year_to_period = True, num_top_types = 7 và max_depth = 9.

Random Forest Regressor

Case 1: month_to_seasons = True và year_to_period = True





- Độ lỗi bên tập train tăng (giảm) thì độ lỗi bên tập validation cũng tăng (giảm) theo.

- Sự ảnh hưởng của các tham số max_depth và num_top_types là khá tương tự như mô hình Decision Tree Regressor.

Case 1: độ chính xác *(score)* trên tập validation cao nhất là 0.9320, với num_top_types = 9 và max_depth = 13

MÔ HÌNH HÓA DỮ LIỆU – RANDOM FOREST REGRESSOR

Nhóm cũng tiến hành thử nghiệm cho 3 trường hợp còn lại với tập giá trị num_top_types = [1, 3, 5, 7, 9, 11, 13] và neighbors = [1, 3, 5, 7, 9 ,11]:

- Case 2: month_to_season = True và year_to_period = False, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.9409 ứng với num_top_types = 5 và max_depth = 11.
- Case 3: month_to_season = False và year_to_period = True, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.9304 ứng với num_top_types = 7 và max_depth = 9.
- Case 4: month_to_season = False và year_to_period = False, độ chính xác *(score)* cao nhất thu được ở tập validation là 0.9434 ứng với num_top_types = 9 và max_depth = 9.

Trong trường hợp dùng **Random Forest Regressor** để fit vào tập train và dự đoán trên tập validation thì thấy rằng: sự ảnh hưởng của các siêu tham số đến độ lỗi trên 2 tập là khá tương tự như trong mô hình **Decision Tree Regressor**, chỉ khác ở đây là dùng nhiều cây (mặc định 10 cây) và độ lỗi trên tập train tăng (giảm) thì độ lỗi trên tập validation cũng tăng (giảm) theo.

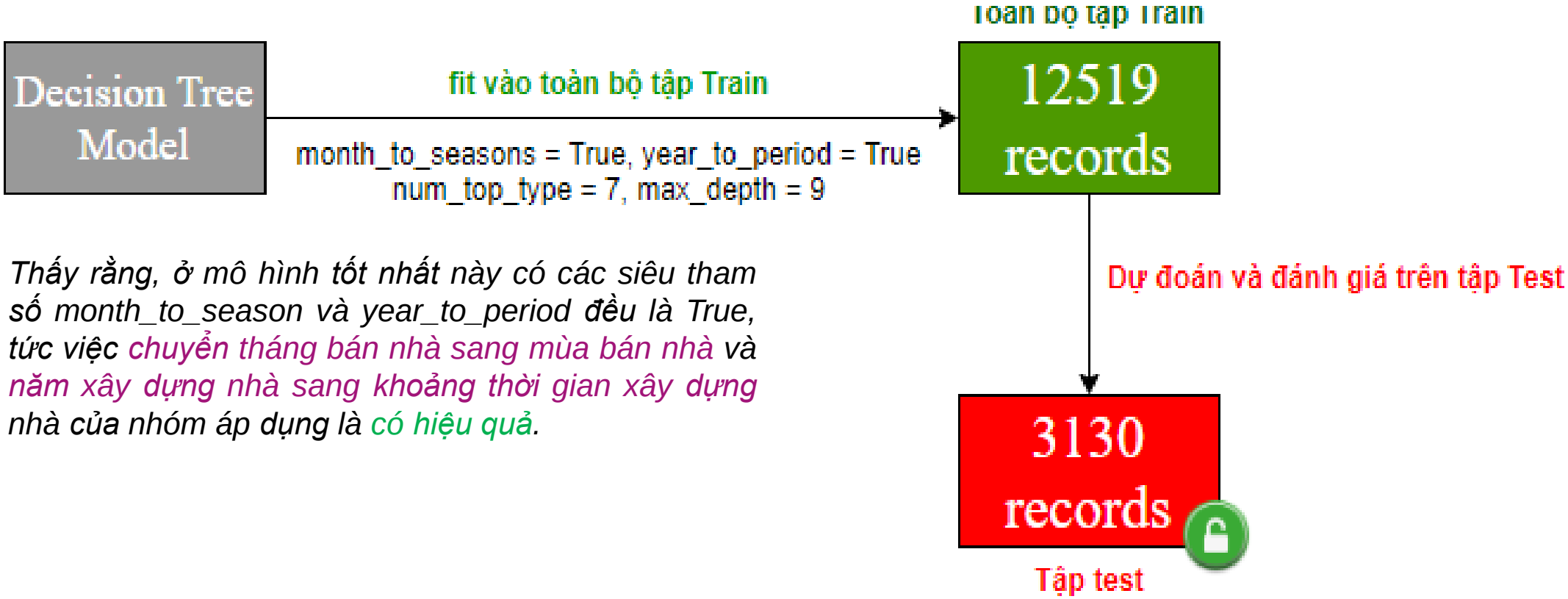
Ghi nhận độ chính xác *(score)* cao nhất trên tập validation khi dùng mô hình **Random Forest Regressor** là 0.9434 ứng với các siêu tham số month_to_season = False, year_to_period = False, num_top_types = 9 và max_depth = 9.

Bảng thống kê Validation Accuracy lớn nhất của các mô hình hồi quy được sử dụng trong Project

Model	month_to_season	year_to_period	num_to_types	neighbors	max_depth	Validation Accuracy (score)
Linear Regression	False	False	9	None	None	0.4131
K-Neighbors Regressor	False	True	1	1	None	0.8429
Decision Tree Regressor	True	True	7	None	9	0.9501
Random Forest Regressor	False	False	9	None	9	0.9434

Mô hình chiến thắng là Decision Tree Regressor với độ chính xác *(score)* trên tập validation là 0.9501.

MÔ HÌNH HÓA DỮ LIỆU – MÔ HÌNH TỐT NHẤT



Thấy rằng, ở mô hình tốt nhất này có các siêu tham số month_to_season và year_to_period đều là True, tức việc chuyển tháng bán nhà sang mùa bán nhà và năm xây dựng nhà sang khoảng thời gian xây dựng nhà của nhóm áp dụng là có hiệu quả.

Độ chính xác *(score)* thu được trên tập test là **0.9477**

MÔ HÌNH HÓA DỮ LIỆU – DỰ ĐOÁN TRÊN TẬP TEST

Bài học rút ra

Tìm hiểu kỹ dữ liệu thu thập và các thuộc tính để chọn đưa vào huấn luyện.

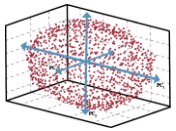
Bắt đầu thực hiện đồ án sớm, mỗi ngày một ít.

Mong muốn

Thu thập thêm dữ để tăng độ chính xác khi dự đoán.

(do thời gian thu thập dữ liệu là đầu tháng 12, nên trang web vẫn chưa cập nhật dữ liệu của tháng 12)

Thử dùng thêm các kỹ thuật giảm chiều dữ liệu như PCA xem như thế nào.



TỔNG KẾT

THANK YOU FOR WATCHING