**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CMC**



**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN: HỌC MÁY VÀ KHAI PHÁ DỮ LIỆU**

**Tên đề tài: Nghiên cứu bài toán dự đoán giá nhà**

**Nhóm sinh viên: Nguyễn Thành Long – BIT230245**

**Nguyễn Thanh Lâm – BIT230227**

**Bùi Duy Hùng – BIT230184**

**Giảng viên hướng dẫn: Phạm Thị Anh Lê**

**Hà Nội, 2025**

**Hà Nội, tháng - 2025**

**MỤC LỤC**

**MỞ ĐẦU 2**

**PHẦN I. TỔNG QUAN 3**

**1.1. GIỚI THIỆU BÀI TOÁN** 3

**1.2. MỘT SỐ THÔNG TIN LIÊN QUAN** 3

**1.3. KẾT LUẬN** 4

**PHẦN II. PHƯƠNG PHÁP THỰC HIỆN 5**

**2.1. DỮ LIỆU SỬ DỤNG** 5

**2.1.1. NGUỒN DỮ LIỆU** 5

**2.1.2. ĐỌC VÀ MÔ TẢ DỮ LIỆU** 5

**2.1.3. TIỀN XỬ LÝ DỮ LIỆU** 27

**2.1.4. CHIA DỮ LIỆU** 27

**2.1.5. BIỂU ĐỒ PHÂN PHỐI GIÁ NHÀ** 27

**2.1.6. KẾT QUẢ DỰ ĐOÁN** 27

**2.1.7. CÁC BIẾN SỐ VÀ ĐẶC ĐIỂM DỮ LIỆU** 28

**2.2. TIỀN XỬ LÝ DỮ LIỆU** 28

**2.2.1. XỬ LÝ GIÁ TRỊ THIẾU** 28

**2.2.2. CHUYỂN ĐỔI BIẾN SỐ** 29

**2.3. XÂY DỰNG MÔ HÌNH** 29

**2.3.1. CÁC MÔ HÌNH ĐƯỢC SỬ DỤNG** 29

**2.3.1.1. GRADIENT BOOSTING** 29

**2.3.1.2. RIDGE REGRESSION** 30

**2.3.2. QUY TRÌNH ĐÁNH GIÁ MÔ HÌNH** 30

**PHẦN III. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ 32**

**3.1. KẾT QUẢ THỰC NGHIỆM** 32

**3.1.1. SO SÁNH CÁC MÔ HÌNH** 32

**3.1.2. ĐÁNH GIÁ HIỆU SUẤT MÔ HÌNH THÔNG QUA RMSLE** 32

**3.2. PHÂN TÍCH KẾT QUẢ** 33

**3.2.1. ĐÁNH GIÁ ĐỘ CHÍNH XÁC CỦA CÁC MÔ HÌNH** 33

**3.2.2. PHÂN TÍCH SAI SỐ DỰ ĐOÁN** 33

**PHẦN IV. KẾT LUẬN 36**

**TÀI LIỆU THAM KHẢO 37**

**MỞ ĐẦU**

Trong bối cảnh thị trường bất động sản ngày càng phát triển và biến động, việc dự đoán giá nhà đã trở thành một nhu cầu thiết yếu không chỉ đối với các nhà đầu tư mà còn đối với người tiêu dùng thông thường. Giá nhà không chỉ phản ánh tình hình kinh tế mà còn ảnh hưởng trực tiếp đến quyết định đầu tư của nhiều cá nhân và tổ chức. Sự hiểu biết về giá trị bất động sản giúp các bên liên quan có thể đưa ra những quyết định hợp lý và kịp thời trong việc mua bán, cho thuê hoặc đầu tư.

Ngày nay, với sự phát triển mạnh mẽ của công nghệ thông tin và các thuật toán học máy, việc dự đoán giá nhà đã trở nên khả thi và hiệu quả hơn bao giờ hết. Các thuật toán này có khả năng phân tích và xử lý một lượng lớn dữ liệu, từ đó rút ra những mối quan hệ phức tạp giữa giá nhà và các yếu tố ảnh hưởng như diện tích, số phòng, vị trí địa lý, tình trạng bất động sản, và nhiều yếu tố khác.

Mục tiêu của đề tài này là phát triển một ứng dụng sử dụng các thuật toán học máy để dự đoán giá nhà dựa trên các yếu tố nêu trên. Ứng dụng này không chỉ giúp người dùng có cái nhìn rõ ràng hơn về giá trị bất động sản mà còn hỗ trợ họ trong việc đưa ra quyết định đầu tư thông minh. Qua đó, đề tài hy vọng sẽ góp phần làm phong phú thêm kiến thức về thị trường bất động sản và ứng dụng công nghệ trong lĩnh vực này.

# PHẦN I. TỔNG QUAN

**1.1. Giới thiệu bài toán**

Bài toán dự đoán giá nhà là một bài toán hồi quy, trong đó chúng ta cần dự đoán giá bán của một ngôi nhà dựa trên các đặc điểm của nó. Các yếu tố ảnh hưởng đến giá nhà bao gồm vị trí, diện tích, số phòng ngủ, số phòng tắm, và nhiều yếu tố khác như tình trạng của ngôi nhà và các tiện ích xung quanh. Việc dự đoán chính xác giá nhà không chỉ giúp người mua đưa ra quyết định tốt hơn mà còn hỗ trợ các nhà đầu tư trong việc phân tích thị trường.

**1.2. Một số thông tin liên quan**

Dữ liệu được sử dụng trong nghiên cứu này được lấy từ một tập dữ liệu công khai, bao gồm các thông tin về giá cả và các đặc điểm của nhà. Tập dữ liệu này có thể chứa hàng nghìn bản ghi, mỗi bản ghi đại diện cho một ngôi nhà với các thông tin khác nhau. Các yếu tố có thể ảnh hưởng đến giá nhà bao gồm:

- Vị trí: Mỗi khu vực có đặc điểm và giá trị riêng, như các khu phố khác nhau (Neighborhood) với các mức độ phát triển khác nhau.

- Diện tích: Diện tích đất (LotArea) và diện tích sử dụng (GrLivArea) của ngôi nhà có vai trò quan trọng trong việc xác định giá trị.

- Số phòng ngủ và phòng tắm: Số lượng phòng ngủ (Bedroom) và phòng tắm (FullBath, HalfBath) là những yếu tố cơ bản ảnh hưởng đến nhu cầu và giá trị của ngôi nhà.

- Tình trạng của ngôi nhà: Chất lượng tổng thể (OverallQual) và tình trạng (OverallCond) của ngôi nhà, cũng như chất liệu xây dựng (Exterior1st, Exterior2nd), đều có tác động lớn đến giá.

- Tiện ích xung quanh: Các yếu tố như loại đường (Street), tình trạng hạ tầng (Utilities), và các tiện ích gần kề (Condition1, Condition2) có thể làm tăng hoặc giảm giá trị của bất động sản.

**1.3. Kết luận**

Bài toán dự đoán giá nhà là một thách thức quan trọng trong lĩnh vực hồi quy, mang lại giá trị thiết thực cho người tiêu dùng và nhà đầu tư. Các yếu tố như vị trí, diện tích, số phòng ngủ và phòng tắm, tình trạng nhà, cùng với tiện ích xung quanh đều ảnh hưởng lớn đến giá trị bất động sản.

Dữ liệu từ tập dữ liệu công khai cung cấp thông tin phong phú, giúp chúng ta đưa ra những dự đoán chính xác hơn. Hiểu rõ các yếu tố này không chỉ hỗ trợ người mua trong quyết định mà còn giúp nhà đầu tư phân tích thị trường hiệu quả hơn, góp phần vào sự phát triển bền vững của thị trường bất động sản.

**PHẦN II. PHƯƠNG PHÁP THỰC HIỆN**

**2.1. Dữ liệu sử dụng**

Trong nghiên cứu này, nhóm sử dụng dữ liệu từ một tập dữ liệu công khai về giá nhà, bao gồm cả tập huấn luyện và tập kiểm tra trên Kaggle. Dữ liệu được tải về từ Google Drive và được xử lý để phục vụ cho việc xây dựng mô hình dự đoán.

**2.1.1. Nguồn dữ liệu**

Dữ liệu được tải về và giải nén từ Google Drive thông qua các lệnh sau:

# Tải dữ liệu từ Google Drive

!gdown --id 1Xn-aobJLKAq9APYOIENj4kYpk7Ox0NQH -O house-prices.zip

!unzip -o house-prices.zip

Sau đó kết quả tập test cũng được tải về:

# Tải kết quả tập test

!gdown --id 12LFeaeW9QJpEJPUAGxEipEh7h9rY66HU

**2.1.2. Đọc và mô tả dữ liệu**

Dữ liệu được đọc vào các DataFrame bằng pandas:

# Đọc dữ liệu

data\_df = pd.read\_csv('train.csv', index\_col=0)

test\_X = pd.read\_csv('test.csv', index\_col=0)

test\_y = pd.read\_csv('test\_result.csv', index\_col=0)['SalePrice']

Sau khi đọc dữ liệu, nhóm thực hiện việc mô tả và kiểm tra thông tin của tập dữ liệu huấn luyện:

# Mô tả dữ liệu

print("\nMô tả dữ liệu:")

print(data\_df.describe())

print("\nThông tin dữ liệu:")

print(data\_df.info())

Mô tả dữ liệu:

MSSubClass LotFrontage LotArea OverallQual OverallCond \

count 1460.000000 1201.000000 1460.000000 1460.000000 1460.000000

mean 56.897260 70.049958 10516.828082 6.099315 5.575342

std 42.300571 24.284752 9981.264932 1.382997 1.112799

min 20.000000 21.000000 1300.000000 1.000000 1.000000

25% 20.000000 59.000000 7553.500000 5.000000 5.000000

50% 50.000000 69.000000 9478.500000 6.000000 5.000000

75% 70.000000 80.000000 11601.500000 7.000000 6.000000

max 190.000000 313.000000 215245.000000 10.000000 9.000000

YearBuilt YearRemodAdd MasVnrArea BsmtFinSF1 BsmtFinSF2 ... \

count 1460.000000 1460.000000 1452.000000 1460.000000 1460.000000 ...

mean 1971.267808 1984.865753 103.685262 443.639726 46.549315 ...

std 30.202904 20.645407 181.066207 456.098091 161.319273 ...

min 1872.000000 1950.000000 0.000000 0.000000 0.000000 ...

25% 1954.000000 1967.000000 0.000000 0.000000 0.000000 ...

50% 1973.000000 1994.000000 0.000000 383.500000 0.000000 ...

75% 2000.000000 2004.000000 166.000000 712.250000 0.000000 ...

max 2010.000000 2010.000000 1600.000000 5644.000000 1474.000000 ...

WoodDeckSF OpenPorchSF EnclosedPorch 3SsnPorch ScreenPorch \

count 1460.000000 1460.000000 1460.000000 1460.000000 1460.000000

mean 94.244521 46.660274 21.954110 3.409589 15.060959

std 125.338794 66.256028 61.119149 29.317331 55.757415

min 0.000000 0.000000 0.000000 0.000000 0.000000

25% 0.000000 0.000000 0.000000 0.000000 0.000000

50% 0.000000 25.000000 0.000000 0.000000 0.000000

75% 168.000000 68.000000 0.000000 0.000000 0.000000

max 857.000000 547.000000 552.000000 508.000000 480.000000

PoolArea MiscVal MoSold YrSold SalePrice

count 1460.000000 1460.000000 1460.000000 1460.000000 1460.000000

mean 2.758904 43.489041 6.321918 2007.815753 180921.195890

std 40.177307 496.123024 2.703626 1.328095 79442.502883

min 0.000000 0.000000 1.000000 2006.000000 34900.000000

25% 0.000000 0.000000 5.000000 2007.000000 129975.000000

50% 0.000000 0.000000 6.000000 2008.000000 163000.000000

75% 0.000000 0.000000 8.000000 2009.000000 214000.000000

max 738.000000 15500.000000 12.000000 2010.000000 755000.000000

[8 rows x 37 columns]

Thông tin dữ liệu:

<class 'pandas.core.frame.DataFrame'>

Index: 1460 entries, 1 to 1460

Data columns (total 80 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 MSSubClass 1460 non-null int64

1 MSZoning 1460 non-null object

2 LotFrontage 1201 non-null float64

3 LotArea 1460 non-null int64

4 Street 1460 non-null object

5 Alley 91 non-null object

6 LotShape 1460 non-null object

7 LandContour 1460 non-null object

8 Utilities 1460 non-null object

9 LotConfig 1460 non-null object

10 LandSlope 1460 non-null object

11 Neighborhood 1460 non-null object

12 Condition1 1460 non-null object

13 Condition2 1460 non-null object

14 BldgType 1460 non-null object

15 HouseStyle 1460 non-null object

16 OverallQual 1460 non-null int64

17 OverallCond 1460 non-null int64

18 YearBuilt 1460 non-null int64

19 YearRemodAdd 1460 non-null int64

20 RoofStyle 1460 non-null object

21 RoofMatl 1460 non-null object

22 Exterior1st 1460 non-null object

23 Exterior2nd 1460 non-null object

24 MasVnrType 588 non-null object

25 MasVnrArea 1452 non-null float64

26 ExterQual 1460 non-null object

27 ExterCond 1460 non-null object

28 Foundation 1460 non-null object

29 BsmtQual 1423 non-null object

30 BsmtCond 1423 non-null object

31 BsmtExposure 1422 non-null object

32 BsmtFinType1 1423 non-null object

33 BsmtFinSF1 1460 non-null int64

34 BsmtFinType2 1422 non-null object

35 BsmtFinSF2 1460 non-null int64

36 BsmtUnfSF 1460 non-null int64

37 TotalBsmtSF 1460 non-null int64

38 Heating 1460 non-null object

39 HeatingQC 1460 non-null object

40 CentralAir 1460 non-null object

41 Electrical 1459 non-null object

42 1stFlrSF 1460 non-null int64

43 2ndFlrSF 1460 non-null int64

44 LowQualFinSF 1460 non-null int64

45 GrLivArea 1460 non-null int64

46 BsmtFullBath 1460 non-null int64

47 BsmtHalfBath 1460 non-null int64

48 FullBath 1460 non-null int64

49 HalfBath 1460 non-null int64

50 BedroomAbvGr 1460 non-null int64

51 KitchenAbvGr 1460 non-null int64

52 KitchenQual 1460 non-null object

53 TotRmsAbvGrd 1460 non-null int64

54 Functional 1460 non-null object

55 Fireplaces 1460 non-null int64

56 FireplaceQu 770 non-null object

57 GarageType 1379 non-null object

58 GarageYrBlt 1379 non-null float64

59 GarageFinish 1379 non-null object

60 GarageCars 1460 non-null int64

61 GarageArea 1460 non-null int64

62 GarageQual 1379 non-null object

63 GarageCond 1379 non-null object

64 PavedDrive 1460 non-null object

65 WoodDeckSF 1460 non-null int64

66 OpenPorchSF 1460 non-null int64

67 EnclosedPorch 1460 non-null int64

68 3SsnPorch 1460 non-null int64

69 ScreenPorch 1460 non-null int64

70 PoolArea 1460 non-null int64

71 PoolQC 7 non-null object

72 Fence 281 non-null object

73 MiscFeature 54 non-null object

74 MiscVal 1460 non-null int64

75 MoSold 1460 non-null int64

76 YrSold 1460 non-null int64

77 SaleType 1460 non-null object

78 SaleCondition 1460 non-null object

79 SalePrice 1460 non-null int64

dtypes: float64(3), int64(34), object(43)

memory usage: 923.9+ KB

None

Ý nghĩa của các cột:

# Ý nghĩa của các cột

with open('data\_description.txt', 'r') as f:

    print(f.read())

MSSubClass: Identifies the type of dwelling involved in the sale.

20 1-STORY 1946 & NEWER ALL STYLES

30 1-STORY 1945 & OLDER

40 1-STORY W/FINISHED ATTIC ALL AGES

45 1-1/2 STORY - UNFINISHED ALL AGES

50 1-1/2 STORY FINISHED ALL AGES

60 2-STORY 1946 & NEWER

70 2-STORY 1945 & OLDER

75 2-1/2 STORY ALL AGES

80 SPLIT OR MULTI-LEVEL

85 SPLIT FOYER

90 DUPLEX - ALL STYLES AND AGES

120 1-STORY PUD (Planned Unit Development) - 1946 & NEWER

150 1-1/2 STORY PUD - ALL AGES

160 2-STORY PUD - 1946 & NEWER

180 PUD - MULTILEVEL - INCL SPLIT LEV/FOYER

190 2 FAMILY CONVERSION - ALL STYLES AND AGES

MSZoning: Identifies the general zoning classification of the sale.

A Agriculture

C Commercial

FV Floating Village Residential

I Industrial

RH Residential High Density

RL Residential Low Density

RP Residential Low Density Park

RM Residential Medium Density

LotFrontage: Linear feet of street connected to property

LotArea: Lot size in square feet

Street: Type of road access to property

Grvl Gravel

Pave Paved

Alley: Type of alley access to property

Grvl Gravel

Pave Paved

NA No alley access

LotShape: General shape of property

Reg Regular

IR1 Slightly irregular

IR2 Moderately Irregular

IR3 Irregular

LandContour: Flatness of the property

Lvl Near Flat/Level

Bnk Banked - Quick and significant rise from street grade to building

HLS Hillside - Significant slope from side to side

Low Depression

Utilities: Type of utilities available

AllPub All public Utilities (E,G,W,& S)

NoSewr Electricity, Gas, and Water (Septic Tank)

NoSeWa Electricity and Gas Only

ELO Electricity only

LotConfig: Lot configuration

Inside Inside lot

Corner Corner lot

CulDSac Cul-de-sac

FR2 Frontage on 2 sides of property

FR3 Frontage on 3 sides of property

LandSlope: Slope of property

Gtl Gentle slope

Mod Moderate Slope

Sev Severe Slope

Neighborhood: Physical locations within Ames city limits

Blmngtn Bloomington Heights

Blueste Bluestem

BrDale Briardale

BrkSide Brookside

ClearCr Clear Creek

CollgCr College Creek

Crawfor Crawford

Edwards Edwards

Gilbert Gilbert

IDOTRR Iowa DOT and Rail Road

MeadowV Meadow Village

Mitchel Mitchell

Names North Ames

NoRidge Northridge

NPkVill Northpark Villa

NridgHt Northridge Heights

NWAmes Northwest Ames

OldTown Old Town

SWISU South & West of Iowa State University

Sawyer Sawyer

SawyerW Sawyer West

Somerst Somerset

StoneBr Stone Brook

Timber Timberland

Veenker Veenker

Condition1: Proximity to various conditions

Artery Adjacent to arterial street

Feedr Adjacent to feeder street

Norm Normal

RRNn Within 200' of North-South Railroad

RRAn Adjacent to North-South Railroad

PosN Near positive off-site feature--park, greenbelt, etc.

PosA Adjacent to postive off-site feature

RRNe Within 200' of East-West Railroad

RRAe Adjacent to East-West Railroad

Condition2: Proximity to various conditions (if more than one is present)

Artery Adjacent to arterial street

Feedr Adjacent to feeder street

Norm Normal

RRNn Within 200' of North-South Railroad

RRAn Adjacent to North-South Railroad

PosN Near positive off-site feature--park, greenbelt, etc.

PosA Adjacent to postive off-site feature

RRNe Within 200' of East-West Railroad

RRAe Adjacent to East-West Railroad

BldgType: Type of dwelling

1Fam Single-family Detached

2FmCon Two-family Conversion; originally built as one-family dwelling

Duplx Duplex

TwnhsE Townhouse End Unit

TwnhsI Townhouse Inside Unit

HouseStyle: Style of dwelling

1Story One story

1.5Fin One and one-half story: 2nd level finished

1.5Unf One and one-half story: 2nd level unfinished

2Story Two story

2.5Fin Two and one-half story: 2nd level finished

2.5Unf Two and one-half story: 2nd level unfinished

SFoyer Split Foyer

SLvl Split Level

OverallQual: Rates the overall material and finish of the house

10 Very Excellent

9 Excellent

8 Very Good

7 Good

6 Above Average

5 Average

4 Below Average

3 Fair

2 Poor

1 Very Poor

OverallCond: Rates the overall condition of the house

10 Very Excellent

9 Excellent

8 Very Good

7 Good

6 Above Average

5 Average

4 Below Average

3 Fair

2 Poor

1 Very Poor

YearBuilt: Original construction date

YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)

RoofStyle: Type of roof

Flat Flat

Gable Gable

Gambrel Gabrel (Barn)

Hip Hip

Mansard Mansard

Shed Shed

RoofMatl: Roof material

ClyTile Clay or Tile

CompShg Standard (Composite) Shingle

Membran Membrane

Metal Metal

Roll Roll

Tar&Grv Gravel & Tar

WdShake Wood Shakes

WdShngl Wood Shingles

Exterior1st: Exterior covering on house

AsbShng Asbestos Shingles

AsphShn Asphalt Shingles

BrkComm Brick Common

BrkFace Brick Face

CBlock Cinder Block

CemntBd Cement Board

HdBoard Hard Board

ImStucc Imitation Stucco

MetalSd Metal Siding

Other Other

Plywood Plywood

PreCast PreCast

Stone Stone

Stucco Stucco

VinylSd Vinyl Siding

Wd Sdng Wood Siding

WdShing Wood Shingles

Exterior2nd: Exterior covering on house (if more than one material)

AsbShng Asbestos Shingles

AsphShn Asphalt Shingles

BrkComm Brick Common

BrkFace Brick Face

CBlock Cinder Block

CemntBd Cement Board

HdBoard Hard Board

ImStucc Imitation Stucco

MetalSd Metal Siding

Other Other

Plywood Plywood

PreCast PreCast

Stone Stone

Stucco Stucco

VinylSd Vinyl Siding

Wd Sdng Wood Siding

WdShing Wood Shingles

MasVnrType: Masonry veneer type

BrkCmn Brick Common

BrkFace Brick Face

CBlock Cinder Block

None None

Stone Stone

MasVnrArea: Masonry veneer area in square feet

ExterQual: Evaluates the quality of the material on the exterior

Ex Excellent

Gd Good

TA Average/Typical

Fa Fair

Po Poor

ExterCond: Evaluates the present condition of the material on the exterior

Ex Excellent

Gd Good

TA Average/Typical

Fa Fair

Po Poor

Foundation: Type of foundation

BrkTil Brick & Tile

CBlock Cinder Block

PConc Poured Contrete

Slab Slab

Stone Stone

Wood Wood

BsmtQual: Evaluates the height of the basement

Ex Excellent (100+ inches)

Gd Good (90-99 inches)

TA Typical (80-89 inches)

Fa Fair (70-79 inches)

Po Poor (<70 inches

NA No Basement

BsmtCond: Evaluates the general condition of the basement

Ex Excellent

Gd Good

TA Typical - slight dampness allowed

Fa Fair - dampness or some cracking or settling

Po Poor - Severe cracking, settling, or wetness

NA No Basement

BsmtExposure: Refers to walkout or garden level walls

Gd Good Exposure

Av Average Exposure (split levels or foyers typically score average or above)

Mn Mimimum Exposure

No No Exposure

NA No Basement

BsmtFinType1: Rating of basement finished area

GLQ Good Living Quarters

ALQ Average Living Quarters

BLQ Below Average Living Quarters

Rec Average Rec Room

LwQ Low Quality

Unf Unfinshed

NA No Basement

BsmtFinSF1: Type 1 finished square feet

BsmtFinType2: Rating of basement finished area (if multiple types)

GLQ Good Living Quarters

ALQ Average Living Quarters

BLQ Below Average Living Quarters

Rec Average Rec Room

LwQ Low Quality

Unf Unfinshed

NA No Basement

BsmtFinSF2: Type 2 finished square feet

BsmtUnfSF: Unfinished square feet of basement area

TotalBsmtSF: Total square feet of basement area

Heating: Type of heating

Floor Floor Furnace

GasA Gas forced warm air furnace

GasW Gas hot water or steam heat

Grav Gravity furnace

OthW Hot water or steam heat other than gas

Wall Wall furnace

HeatingQC: Heating quality and condition

Ex Excellent

Gd Good

TA Average/Typical

Fa Fair

Po Poor

CentralAir: Central air conditioning

N No

Y Yes

Electrical: Electrical system

SBrkr Standard Circuit Breakers & Romex

FuseA Fuse Box over 60 AMP and all Romex wiring (Average)

FuseF 60 AMP Fuse Box and mostly Romex wiring (Fair)

FuseP 60 AMP Fuse Box and mostly knob & tube wiring (poor)

Mix Mixed

1stFlrSF: First Floor square feet

2ndFlrSF: Second floor square feet

LowQualFinSF: Low quality finished square feet (all floors)

GrLivArea: Above grade (ground) living area square feet

BsmtFullBath: Basement full bathrooms

BsmtHalfBath: Basement half bathrooms

FullBath: Full bathrooms above grade

HalfBath: Half baths above grade

Bedroom: Bedrooms above grade (does NOT include basement bedrooms)

Kitchen: Kitchens above grade

KitchenQual: Kitchen quality

Ex Excellent

Gd Good

TA Typical/Average

Fa Fair

Po Poor

TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)

Functional: Home functionality (Assume typical unless deductions are warranted)

Typ Typical Functionality

Min1 Minor Deductions 1

Min2 Minor Deductions 2

Mod Moderate Deductions

Maj1 Major Deductions 1

Maj2 Major Deductions 2

Sev Severely Damaged

Sal Salvage only

Fireplaces: Number of fireplaces

FireplaceQu: Fireplace quality

Ex Excellent - Exceptional Masonry Fireplace

Gd Good - Masonry Fireplace in main level

TA Average - Prefabricated Fireplace in main living area or Masonry Fireplace in basement

Fa Fair - Prefabricated Fireplace in basement

Po Poor - Ben Franklin Stove

NA No Fireplace

GarageType: Garage location

2Types More than one type of garage

Attchd Attached to home

Basment Basement Garage

BuiltIn Built-In (Garage part of house - typically has room above garage)

CarPort Car Port

Detchd Detached from home

NA No Garage

GarageYrBlt: Year garage was built

GarageFinish: Interior finish of the garage

Fin Finished

RFn Rough Finished

Unf Unfinished

NA No Garage

GarageCars: Size of garage in car capacity

GarageArea: Size of garage in square feet

GarageQual: Garage quality

Ex Excellent

Gd Good

TA Typical/Average

Fa Fair

Po Poor

NA No Garage

GarageCond: Garage condition

Ex Excellent

Gd Good

TA Typical/Average

Fa Fair

Po Poor

NA No Garage

PavedDrive: Paved driveway

Y Paved

P Partial Pavement

N Dirt/Gravel

WoodDeckSF: Wood deck area in square feet

OpenPorchSF: Open porch area in square feet

EnclosedPorch: Enclosed porch area in square feet

3SsnPorch: Three season porch area in square feet

ScreenPorch: Screen porch area in square feet

PoolArea: Pool area in square feet

PoolQC: Pool quality

Ex Excellent

Gd Good

TA Average/Typical

Fa Fair

NA No Pool

Fence: Fence quality

GdPrv Good Privacy

MnPrv Minimum Privacy

GdWo Good Wood

MnWw Minimum Wood/Wire

NA No Fence

MiscFeature: Miscellaneous feature not covered in other categories

Elev Elevator

Gar2 2nd Garage (if not described in garage section)

Othr Other

Shed Shed (over 100 SF)

TenC Tennis Court

NA None

MiscVal: $Value of miscellaneous feature

MoSold: Month Sold (MM)

YrSold: Year Sold (YYYY)

SaleType: Type of sale

WD Warranty Deed - Conventional

CWD Warranty Deed - Cash

VWD Warranty Deed - VA Loan

New Home just constructed and sold

COD Court Officer Deed/Estate

Con Contract 15% Down payment regular terms

ConLw Contract Low Down payment and low interest

ConLI Contract Low Interest

ConLD Contract Low Down

Oth Other

SaleCondition: Condition of sale

Normal Normal Sale

Abnorml Abnormal Sale - trade, foreclosure, short sale

AdjLand Adjoining Land Purchase

Alloca Allocation - two linked properties with separate deeds, typically condo with a garage unit

Family Sale between family members

Partial Home was not completed when last assessed (associated with New Homes)

Qua mô tả, nhóm có các nhận xét sau:

- Cần lưu ý tại các cột sau vì việc thiếu dữ liệu của các cột này là có chủ đích:

+ Alley: kiểu đường hiểm mà căn hộ tiếp xúc => Không tiếp xúc đường hẻm thì sẽ để trống.

+ BsmtQual, BsmtCond, BsmtExposure, BsmtFinType1, BsmtFinType2: các thuộc tính liên quan tới tầng hầm của căn hộ => Không có tầng hầm sẽ để trống

+ FireplaceQu: chất lượng lò sưởi => Không có lò sưởi sẽ để trống.

+ GarageType, GarageFinish, GarageQual, GarageCond: thuộc tính liên quan tới garage => Không có garage sẽ để trống.

+ PoolQC: chất lượng hồ bơi => Không có hồ bơi sẽ để trống.

+ Fence: chất lượng hàng rào => Không có hàng rào sẽ để trống.

+ MiscFeature: một số tiện nghi khác của căn hộ mà không được liệt kê bởi các thuộc tính trong bảng => Không có sẽ để trống.

- Có một số thuộc tính có dạng multiple choice -> cần lưu ý khi onehot (phương pháp mã hoá dữ liệu phân loại (categorical data) thành dạng số).

**2.1.3. Tiền xử lý dữ liệu**

Để chuẩn bị dữ liệu cho mô hình, nhóm đã xác định các cột số và cột phân loại:

- Các cột phân loại (categorical columns) được xử lý bằng cách sử dụng OneHotEncoder và SimpleImputer để thay thế giá trị thiếu.

- Các cột số (numerical columns) được xử lý bằng SimpleImputer để thay thế giá trị thiếu bằng giá trị trung bình.

# Thống kê sơ bộ cho các cột

cat\_cols = data\_df.dtypes.index[ data\_df.dtypes == 'O' ].tolist()

num\_cols = data\_df.dtypes.index[ data\_df.dtypes != 'O' ].tolist()

num\_cols.remove('SalePrice')

**2.1.4. Chia dữ liệu**

Dữ liệu được chia thành tập train và test với tỷ lệ 70-30:

# Chia dữ liệu thành tập huấn luyện và tập kiểm tra

data\_X = data\_df.drop('SalePrice', axis=1)

data\_y = data\_df['SalePrice']

train\_X, val\_X, train\_y, val\_y = train\_test\_split(data\_X, data\_y, test\_size=0.3)

**2.1.5. Biểu đồ phân phối giá nhà**

Trước khi xây dựng mô hình, nhóm cũng thực hiện phân tích sơ bộ bằng cách vẽ biểu đồ phân phối giá nhà:

sns.histplot(data\_df['SalePrice'], bins=50, kde=True)

Sau khi thực hiện phép biến đổi log trên giá nhà để làm giảm độ lệch:

data\_df["SalePrice"] = np.log1p(data\_df["SalePrice"])

**2.1.6. Kết quả dự đoán**

Sau khi huấn luyện mô hình, nhóm đã thực hiện dự đoán trên tập dữ liệu kiểm tra và lưu kết quả vào các file CSV để phân tích sau này:

gbr\_results.to\_csv('gbr\_final.csv', index=False)

ridge\_results.to\_csv('ridge\_final.csv', index=False)

**2.1.7. Các biến số và đặc điểm dữ liệu**

Tập dữ liệu huấn luyện bao gồm nhiều biến số khác nhau, chia thành hai loại chính: biến số định lượng (numerical variables) và biến số định tính (categorical variables).

- Biến số định lượng: thường chứa các giá trị số và có thể được sử dụng trong các phép toán số học. Một số biến số định lượng quan trọng trong tập dữ liệu này bao gồm:

+ SalePrice: Giá bán của ngôi nhà (biến mục tiêu)

+ LotArea: Diện tích lô đất

+ OverallQual: Đánh giá chất lượng tổng thể của ngôi nhà

+ YearBuilt: Năm xây dựng ngôi nhà

+ TotalBsmtSF: Diện tích tổng của tầng hầm

- Biến số định tính: thường là các thuộc tính mô tả và không thể thực hiện phép toán số học. Một số biến số định tính trong tập dữ liệu này bao gồm:

+ Neighborhood: Khu vực mà ngôi nhà toạ lạc

+ HouseStyle: Kiểu dáng của ngôi nhà

+ ExterQual: Chất lượng vật liệu bên ngoài

+ GarageType: Loại gara (nếu có)

- Đặc điểm dữ liệu:

+ Giá trị thiếu: Tập dữ liệu có thể chứa các giá trị thiếu, điều này cần được xử lý trước khi tiến hành xây dựng mô hình.

+ Phân phối giá: Giá bán của ngôi nhà có thể có phân phối không đồng đều, vì vậy việc áp dụng các phép biến đổi như log-transform có thể cần thiết để cải thiện hiệu suất của mô hình.

**2.2. Tiền xử lý dữ liệu**

**2.2.1. Xử lý giá trị thiếu**

Trong tập dữ liệu, có thể xuất hiện các giá trị thiếu, điều này có thể ảnh hưởng đến hiệu suất của mô hình. Để xử lý vấn đề này, nhóm sử dụng SimpleImputer từ thư viện sklearn để thay thế các giá trị thiếu:

- Biến số định tính: Các biến này được xử lý bằng cách thay thế giá trị thiếu bằng một giá trị hằng định, cụ thể là chuỗi 'MissingValue'. Điều này giúp mô hình nhận diện các trường hợp không có dữ liệu mà vẫn không làm mất thông tin.

- Biến số định lượng: Đối với các biến số này, nhóm sử dụng phương pháp thay thế giá trị thiếu bằng giá trị trung bình của biến. Điều này giúp duy trì tính đồng nhất và không làm sai lệch các phép toán số học.

# Tiền xử lý dữ liệu

categorical\_transformer = Pipeline(steps=[

    ('imputer', SimpleImputer(strategy='constant', fill\_value='MissingValue')),

    ('onehot', OneHotEncoder(handle\_unknown='ignore', sparse\_output=False))

])

numeric\_transformer = Pipeline(steps=[

    ('imputer', SimpleImputer(strategy='mean'))

])

**2.2.2. Chuyển đổi biến số**

Chuyển đổi biến số là một bước quan trọng nhằm cải thiện khả năng dự đoán của mô hình. Trong trường hợp này, nhóm thực hiện hai loại chuyển đổi:

- Chuyển đổi log: Để giảm độ lệch của giá trị mục tiêu (giá nhà), nhóm áp dụng phép biến đổi log trên biến SalePrice. Điều này giúp phân phối của giá mục tiêu trở nên đồng đều hơn, từ đó cải thiện hiệu suất của các mô hình hồi quy.

data\_df["SalePrice"] = np.log1p(data\_df["SalePrice"])

- Chuyển đổi biến phân loại: Các biến phân loại được chuyển đổi thành các biến nhị phân thông qua OneHotEncoder. Việc này giúp mô hình dễ dàng xử lý các biến phân loại mà không làm mất thông tin.

preprocessor = ColumnTransformer( transformers=[ ('numeric', numeric\_transformer, num\_cols), ('categorical', categorical\_transformer, cat\_cols) ])

Thông qua các bước tiền xử lý này, dữ liệu đã được chuẩn bị sẵn sàng cho việc xây dựng và huấn luyện mô hình. Điều này không chỉ giúp cải thiện độ chính xác của mô hình mà còn đảm bảo rằng các yếu tố quan trọng được xem xét trong quá trình dự đoán.

**2.3. Xây dựng mô hình**

Trong phần này, nhóm sẽ thảo luận về quy trình xây dựng mô hình dự đoán giá nhà, bao gồm các mô hình được sử dụng và quy trình đánh giá hiệu suất của chúng.

**2.3.1. Các mô hình được sử dụng**

**2.3.1.1. Gradient Boosting**

Mô hình Gradient Boosting là một kỹ thuật học máy mạnh mẽ, sử dụng phương pháp ensemble để cải thiện độ chính xác của các dự đoán. Mô hình này xây dựng nhiều cây quyết định (decision trees) theo cách mà mỗi cây mới sẽ cố gắng sửa chữa các lỗi của cây trước đó.

- Cách hoạt động: Gradient Boosting hoạt động bằng cách tối ưu hóa một hàm mất mát (loss function) thông qua quá trình học tuần tự, nơi mỗi cây được thêm vào để cải thiện hiệu suất của mô hình hiện tại.

- Ưu điểm: Mô hình này có khả năng xử lý tốt các biến số phi tuyến tính và thường mang lại kết quả chính xác hơn so với các mô hình hồi quy đơn giản.

from sklearn.ensemble import GradientBoostingRegressor

gbr = GradientBoostingRegressor(n\_estimators=100, random\_state=42)

gbr.fit(train\_X, train\_y)

**2.3.1.2. Ridge Regression**

Ridge Regression là một phương pháp hồi quy tuyến tính được điều chỉnh, nhằm giảm thiểu vấn đề overfitting bằng cách thêm một điều kiện vào hàm mất mát.

- Cách hoạt động: Mô hình này sử dụng một tham số điều chỉnh (alpha) để kiểm soát độ phức tạp của mô hình, giúp cải thiện độ chính xác khi có nhiều biến độc lập.

- Ưu điểm: Ridge Regression rất hiệu quả trong việc xử lý các biến số có mối tương quan cao và giúp ổn định các ước lượng hồi quy.

from sklearn.linear\_model import Ridge

ridge = Ridge(alpha=1.0)

ridge.fit(train\_X, train\_y)

**2.3.2. Quy trình đánh giá mô hình**

Để đánh giá hiệu suất của các mô hình, nhóm sử dụng tập dữ liệu kiểm tra và các chỉ số đánh giá như RMSE (Root Mean Squared Error) và R² (Coefficient of Determination).

- Chia dữ liệu: Dữ liệu được chia thành tập huấn luyện và tập kiểm tra với tỷ lệ 70-30, giúp đảm bảo rằng mô hình được huấn luyện trên một phần lớn của dữ liệu và kiểm tra trên phần còn lại.

train\_X, val\_X, train\_y, val\_y = train\_test\_split(data\_X, data\_y, test\_size=0.3)

- Đánh giá mô hình: Sau khi huấn luyện, nhóm thực hiện dự đoán trên tập kiểm tra và tính toán các chỉ số đánh giá để so sánh hiệu suất của các mô hình.

from sklearn.metrics import mean\_squared\_error, r2\_score

gbr\_preds = gbr.predict(val\_X)

ridge\_preds = ridge.predict(val\_X)

gbr\_rmse = np.sqrt(mean\_squared\_error(val\_y, gbr\_preds))

ridge\_rmse = np.sqrt(mean\_squared\_error(val\_y, ridge\_preds))

gbr\_r2 = r2\_score(val\_y, gbr\_preds)

ridge\_r2 = r2\_score(val\_y, ridge\_preds)

**PHẦN III. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ**

**3.1. Kết quả thực nghiệm**

Kết quả thực nghiệm được thu thập từ việc áp dụng hai mô hình chính: Gradient Boosting và Ridge Regression. Nhóm đã thực hiện các phép đo đánh giá để so sánh hiệu suất của các mô hình này trên tập dữ liệu kiểm tra.

#### 3.1.1. So sánh các mô hình

Để so sánh các mô hình, nhóm đã tính toán các chỉ số đánh giá như RMSE (Root Mean Squared Error) và R² (Coefficient of Determination). Dưới đây là kết quả so sánh:

- Gradient Boosting:

+ RMSE: Giá trị thấp hơn cho thấy mô hình này có khả năng dự đoán chính xác hơn.

+ R²: Giá trị gần 1 cho thấy mô hình giải thích tốt sự biến thiên của giá nhà.

- Ridge Regression:

+ RMSE: Mặc dù có thể cao hơn một chút so với Gradient Boosting, nhưng vẫn cho thấy khả năng dự đoán khá tốt.

+ R²: Giá trị cũng gần 1, nhưng không bằng mô hình Gradient Boosting.

print(f"Gradient Boosting RMSE: {gbr\_rmse}, R²: {gbr\_r2}")

print(f"Ridge Regression RMSE: {ridge\_rmse}, R²: {ridge\_r2}")

**3.1.2. Đánh giá hiệu suất mô hình thông qua RMSE**

RMSLE (Root Mean Squared Logarithmic Error) là một chỉ số đánh giá quan trọng khi làm việc với dữ liệu có phân phối không đồng đều, như giá nhà. RMSLE giúp đo lường độ chính xác của mô hình bằng cách so sánh log của giá dự đoán và log của giá thực tế.

- RMSLE được tính bằng công thức:

RMSLE =

Trong đó, yi là giá thực tế và là giá dự đoán.

- Đánh giá mô hình: Nhóm đã tính toán RMSLE cho cả hai mô hình để đánh giá hiệu suất của chúng.

from sklearn.metrics import mean\_squared\_log\_error

gbr\_rmsle = np.sqrt(mean\_squared\_log\_error(val\_y, gbr\_preds))

ridge\_rmsle = np.sqrt(mean\_squared\_log\_error(val\_y, ridge\_preds))

print(f"Gradient Boosting RMSLE: {gbr\_rmsle}")

print(f"Ridge Regression RMSLE: {ridge\_rmsle}")

Kết quả RMSLE cho thấy mô hình Gradient Boosting có hiệu suất tốt hơn so với Ridge Regression trong việc dự đoán giá nhà, đặc biệt là khi dữ liệu có phân phối không đồng đều.

**3.2. Phân tích kết quả**

**3.2.1. Đánh giá độ chính xác của các mô hình**

Để đánh giá độ chính xác của các mô hình, nhóm đã sử dụng các chỉ số như RMSE, R² và RMSLE. Những chỉ số này cung cấp cái nhìn tổng quan về hiệu suất của từng mô hình.

- Gradient Boosting:

+ Độ chính xác cao: Với RMSE và RMSLE thấp, mô hình này cho thấy khả năng dự đoán chính xác hơn so với Ridge Regression.

+ R² gần 1: Điều này cho thấy mô hình giải thích tốt sự biến thiên của giá nhà, tức là mô hình có thể nắm bắt được mối quan hệ giữa các biến độc lập và giá mục tiêu.

- Ridge Regression:

+ Độ chính xác tương đối: Mặc dù RMSE và RMSLE cao hơn so với Gradient Boosting, nhưng vẫn nằm trong mức chấp nhận được.

+ R² cũng gần 1: Tuy nhiên, mô hình này không thể đạt được độ chính xác như Gradient Boosting trong việc dự đoán giá nhà.

print(f"Gradient Boosting - RMSE: {gbr\_rmse}, R²: {gbr\_r2}, RMSLE: {gbr\_rmsle}")

print(f"Ridge Regression - RMSE: {ridge\_rmse}, R²: {ridge\_r2}, RMSLE: {ridge\_rmsle}")

**3.2.2. Phân tích sai số dự đoán**

Phân tích sai số dự đoán là một phần quan trọng trong việc hiểu rõ hơn về hiệu suất của mô hình. Nhóm đã thực hiện phân tích này bằng cách xem xét các sai số giữa giá thực tế và giá dự đoán.

- Tính toán RMSLE:

# Hàm tính RMSLE

def root\_MSLE(y\_true, y\_pred):

    return mean\_squared\_log\_error(y\_true, y\_pred) \*\* (1/2)

# Hàm đánh giá mô hình

def TestModel(model, name):

    pipeline = Pipeline(steps=[('preprocessor', preprocessor), ('regressor', model)])

    pipeline.fit(train\_X, train\_y)

    val\_pred = pipeline.predict(val\_X)

    rmsle = root\_MSLE(val\_y, val\_pred)

    print(f'RMSLE của {name}: {rmsle}')

    return pipeline, rmsle

Hàm root\_MSLE tính toán sai số trung bình bình phương log (RMSLE) giữa giá trị thực tế và giá trị dự đoán. Đây là một chỉ số quan trọng để đánh giá độ chính xác của mô hình.

- So sánh RMSLE giữa các mô hình:

# Hiển thị bảng so sánh RMSLE

rmsle\_df = pd.DataFrame(list(rmsle\_scores.items()), columns=["Model", "RMSLE"])

print("\nBảng so sánh RMSLE:")

print(rmsle\_df)

# So sánh kết quả bằng biểu đồ

plt.figure(figsize=(8, 5))

plt.bar(rmsle\_scores.keys(), rmsle\_scores.values(), color=['blue', 'green'])

plt.ylabel("RMSLE")

plt.title("So sánh RMSLE giữa các mô hình")

plt.show()

Tạo một DataFrame để hiển thị RMSLE của từng mô hình và vẽ biểu đồ cột để so sánh. Điều này giúp bạn dễ dàng nhận thấy mô hình nào hoạt động tốt hơn dựa trên sai số RMSLE.

- Dự đoán và so sánh giá trị thực tế với giá trị dự đoán:

plt.figure(figsize=(10, 5))

plt.scatter(test\_y, final\_gbr\_pred, alpha=0.5, label="Gradient Boosting")

plt.scatter(test\_y, final\_ridge\_pred, alpha=0.5, label="Ridge Regression")

plt.plot([min(test\_y), max(test\_y)], [min(test\_y), max(test\_y)], 'r--', label="Perfect Prediction")

plt.xlabel("Actual Price")

plt.ylabel("Predicted Price")

plt.legend()

plt.title("So sánh dự đoán và giá thực tế")

plt.show()

Đoạn mã này vẽ biểu đồ phân tán để so sánh giá trị thực tế với giá trị dự đoán từ hai mô hình. Đường thẳng màu đỏ thể hiện "Perfect Prediction" (dự đoán hoàn hảo). Nếu các điểm nằm gần đường này, điều đó cho thấy mô hình dự đoán tốt.

**PHẦN IV. KẾT LUẬN**

Đề tài đã xây dựng thành công một ứng dụng dự đoán giá nhà với độ chính xác cao. Các kết quả cho thấy rằng việc sử dụng các mô hình học máy có thể mang lại những dự đoán giá trị cho thị trường bất động sản. Tuy nhiên, còn nhiều yếu tố khác có thể ảnh hưởng đến giá nhà mà chưa được xem xét, do đó, nghiên cứu tiếp theo có thể mở rộng thêm các yếu tố này để cải thiện độ chính xác của mô hình. Việc tối ưu hóa mô hình và thử nghiệm với các thuật toán khác cũng là những hướng đi tiềm năng cho nghiên cứu trong tương lai.

**TÀI LIỆU THAM KHẢO**

1. [House Prices - Advanced Regression Techniques | Kaggle](https://www.kaggle.com/c/house-prices-advanced-regression-techniques)
2. [House Price Prediction with Creative Feature Engineering and Advanced Regression Techniques | Data Science Blog (nycdatascience.com)](https://nycdatascience.com/blog/student-works/house-price-prediction-with-creative-feature-engineering-and-advanced-regression-techniques/)
3. [#1 House Prices Solution [top 1%] | Kaggle](https://www.kaggle.com/jesucristo/1-house-prices-solution-top-1)
4. [Feature Engineering for House Prices | Kaggle](https://www.kaggle.com/chelseabrk/feature-engineering-for-house-prices)
5. [Ridge and Lasso Regression: L1 and L2 Regularization | by Saptashwa Bhattacharyya | Towards Data Science](https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b)
6. [5 Regression Loss Functions All Machine Learners Should Know | by Prince Grover | Heartbeat (fritz.ai)](https://heartbeat.fritz.ai/5-regression-loss-functions-all-machine-learners-should-know-4fb140e9d4b0)

**PHỤ LỤC**

https://github.com/nguyenlongnt270705/kaggle