**Abstract:**

Giá nhà luôn biến động và việc dự đoán giá nhà trong tương lai là điều cần thiết do tầm quan trọng của nó đối với người mua, nhà đầu tư bất động sản và nền kinh tế nói chung. Bài báo này phân tích các yếu tố quan trọng ảnh hưởng đến giá nhà và trình bày phương pháp dự đoán giá nhà tại Ames, Iowa bằng các kỹ thuật hồi quy tiên tiến, bao gồm phân tích tương quan, kỹ thuật đặc trưng và xây dựng mô hình.

Dự đoán được thực hiện bằng các phương pháp học máy trên tập dữ liệu chứa gần như đầy đủ thông tin về các ngôi nhà dân cư tại Ames, Iowa. Dữ liệu ban đầu được chia thành hai tập: tập huấn luyện và tập kiểm tra. Sáu mô hình đã được lựa chọn để đánh giá, bao gồm Hồi quy LASSO, Hồi quy Elastic Net, Hồi quy Tăng cường độ dốc (Gradient Boosting Regression), XGBoost, LightGBM và Mô hình kết hợp (Stacked Model). Các mô hình được kiểm định thông qua phương pháp đánh giá chéo năm lần (five-fold cross-validation) để kiểm tra hiệu suất thuật toán và báo cáo sai số căn bậc hai trung bình trên thang logarit (root-mean-square logarithmic error - RMSLE).

Kết quả thực nghiệm cho thấy mô hình Hồi quy Tăng cường độ dốc đạt sai số RMSLE thấp nhất là 0.04615 và giá trị đánh giá chéo thấp là 0.1174. Do đó, mô hình này được lựa chọn để dự đoán giá nhà, nhằm mang lại lợi ích cho người mua, nhà đầu tư và toàn xã hội.

**Keywords:**  Price  prediction;  feature  engineering;  LASSO  Regression;  Elastic  Net  Regression; Gradient Boosting Regression; XGBoost; LightGBM.

**1. Giới thiệu**

Với tốc độ hình thành và phát triển nhanh chóng của các đô thị cùng với sự gia tăng số lượng người nhập cư tại Hoa Kỳ, quá trình đô thị hóa đang diễn ra nhanh chóng, kéo theo một loạt thách thức mới. Một trong những trở ngại đáng kể nhất là sự gia tăng không ngừng của giá nhà do mật độ dân số ngày càng tăng. Theo nghiên cứu của Chenhao Zhou, giá nhà trung bình vào những năm 1970 vào khoảng 27.000 USD, nhưng với số tiền này ngày nay thậm chí không thể mua được một căn hộ hai phòng ngủ ở một số bang kém phát triển hơn [1]. Do đó, việc dự đoán giá nhà trở nên vô cùng quan trọng.

Khi con người thực hiện các giao dịch mua bán nhà, dù để ở hay đầu tư, họ đang tham gia vào thị trường bất động sản. Xu hướng giá nhà có ảnh hưởng đáng kể đến nền kinh tế nói chung. Cụ thể, thị trường bất động sản có mối quan hệ mật thiết với mức chi tiêu của người tiêu dùng. Xu hướng giá nhà tăng thường góp phần củng cố niềm tin và thúc đẩy mức chi tiêu cao hơn. Chủ sở hữu nhà sẽ có xu hướng chi tiêu nhiều hơn cho các nhu yếu phẩm hàng ngày, dịch vụ nhà ở, cải tạo, trang trí nội ngoại thất, từ đó góp phần thúc đẩy Tổng sản phẩm quốc nội (GDP) và tăng trưởng kinh tế. Ngược lại, xu hướng giá nhà giảm có thể làm suy giảm niềm tin của người tiêu dùng, do lo ngại rằng tài sản của họ có thể mất giá trị so với khoản thế chấp chưa thanh toán. Khi đó, người dân có xu hướng cắt giảm chi tiêu và hạn chế đầu tư, làm tăng nguy cơ suy thoái kinh tế.

Việc dự đoán giá nhà cũng rất quan trọng đối với cả người mua và nhà đầu tư. Nó giúp người mua có thêm thông tin về thị trường bất động sản hiện tại để lập kế hoạch tài chính phù hợp, đồng thời cung cấp những yếu tố quan trọng cần xem xét khi lựa chọn một căn nhà mơ ước. Quan trọng hơn, nó giúp xác định thời điểm thích hợp để mua nhà. Đối với nhà đầu tư, việc dự đoán giá nhà giúp xác định mức giá bán hợp lý và đánh giá xu hướng giá bất động sản tại một khu vực cụ thể nhằm giảm thiểu rủi ro đầu tư. Vì bất động sản là một tài sản quan trọng trong cuộc sống của mỗi người, nhưng không phải ai cũng là chuyên gia trong việc lựa chọn nhà ở hoặc đầu tư tiềm năng.

Khả năng nhận biết giá trị thực sự của một ngôi nhà của người mua thường bị hạn chế. Họ có thể dễ dàng coi bể bơi, số lượng phòng ngủ và diện tích phòng khách là những yếu tố quan trọng của một căn nhà lý tưởng. Tuy nhiên, người mua có thể không nhận thức được những đặc điểm quan trọng nhưng ít hiển nhiên hơn, chẳng hạn như vị trí gần đường sắt hướng tây nam, loại vật liệu phủ bên ngoài ngôi nhà hoặc chất lượng của gara. Do đó, bài báo này chứng minh rằng có nhiều yếu tố quan trọng khác ảnh hưởng đáng kể đến giá bán nhà.

Có nhiều phương pháp khác nhau để dự đoán giá nhà, trong đó có phương pháp phân tích chuỗi thời gian. Tương tự như phân tích hồi quy, phân tích chuỗi thời gian là một phương pháp dự báo dựa trên các điểm dữ liệu trong quá khứ. Tuy nhiên, dữ liệu trong chuỗi thời gian được thu thập theo một khoảng thời gian nhất định, thay vì ngẫu nhiên hoặc không liên tục. Điều kiện tiên quyết này khiến nhiều tập dữ liệu về giá nhà không phù hợp với phương pháp này, vì dữ liệu thường không bao gồm đầy đủ thông tin về thời gian. Phương pháp chuỗi thời gian được sử dụng khi có đủ dữ liệu lịch sử và có thể giả định rằng một số mô hình trong quá khứ sẽ tiếp tục trong tương lai.

Tuy nhiên, giá nhà là một trong những tài sản có biến động liên tục và không có đơn giá cố định như các hàng hóa và dịch vụ khác. Giá nhà không chỉ được xác định theo khoảng thời gian như tháng hay quý, mà còn phụ thuộc vào nhiều yếu tố khác nhau. Do đó, mô hình chuỗi thời gian không thể đủ mạnh để dự đoán giá nhà một cách chính xác.

Thay vì phụ thuộc quá nhiều vào một biến duy nhất như thời gian, các kỹ thuật hồi quy là một phương pháp tìm kiếm mối quan hệ giữa biến mục tiêu và nhiều biến độc lập có liên quan. Đây là một phương pháp thống kê đáng tin cậy để xác định các biến ảnh hưởng đến biến mục tiêu. Quá trình này giúp xác định những yếu tố nào quan trọng nhất trong tập dữ liệu, những yếu tố nào có thể bỏ qua và cách các biến này tương tác với nhau. Tập hợp tốt nhất của các biến đầu vào từ phương pháp hồi quy có thể tạo ra mô hình dự đoán có sai số căn bậc hai trung bình trên thang logarit (RMSLE) thấp nhất.

Hồi quy là một trong những khái niệm cơ bản và dễ hiểu nhất trong học máy, nhưng lại có sức mạnh lớn trong việc xây dựng mô hình và dự báo. Dữ liệu được sử dụng trong nghiên cứu này có nguồn gốc từ cuộc thi trên Kaggle, “House Prices: Advanced Regression Techniques”. Dữ liệu ban đầu được chia thành hai tập: tập huấn luyện (train set) và tập kiểm tra (test set) để phân tích và đánh giá mô hình.

Mục tiêu của nghiên cứu này có thể được tóm tắt thành hai điểm chính:

1. Xác định các yếu tố thực sự có liên quan đến sự biến động giá nhà.

2. Xây dựng mô hình để đánh giá và dự đoán giá bán của từng ngôi nhà theo từng ‘Id’ trong tập kiểm tra.

**2. Tổng quan tài liệu**

Nhằm tìm ra mô hình tối ưu nhất, các nhà phân tích đã tập trung vào việc xác định tập hợp đặc trưng phù hợp nhất. Có rất nhiều yếu tố ảnh hưởng đến giá nhà. Trong nghiên cứu của Rahidi và cộng sự, nhóm nghiên cứu đã phân loại các yếu tố tiềm năng này thành ba nhóm chính: điều kiện vật lý, khái niệm và vị trí [2].

Nhóm *điều kiện vật lý* bao gồm các đặc điểm hữu hình của một bất động sản có thể quan sát trực tiếp, chẳng hạn như số lượng phòng ngủ, diện tích phòng khách và năm xây dựng. Nhóm *khái niệm* bao gồm những ý tưởng do các nhà phát triển bất động sản đề xuất nhằm thu hút người mua tiềm năng, ví dụ như phong cách thiết kế hiện đại hay môi trường sống lành mạnh [3]. Cuối cùng, nhóm *vị trí* được coi là một trong những yếu tố quan trọng nhất ảnh hưởng đến giá nhà, bởi lẽ vị trí quyết định giá trị của đất đai [4]. Ngoài ra, vị trí cũng bao gồm các yếu tố tiện ích như khả năng tiếp cận bệnh viện, trường học, trung tâm thương mại và ga tàu [5,6].

Sau khi tổng hợp các nghiên cứu trước đây, các đặc trưng được lựa chọn trong nghiên cứu này cũng có thể được phân thành ba nhóm chính:

• *Xây dựng*: Bao gồm các biến thể hiện đặc điểm vật lý của kiến trúc (ví dụ: ‘OverallQual’).

• *Không gian*: Ghi nhận các yếu tố liên quan đến đặc tính không gian của căn nhà (ví dụ: ‘TotalBsmtSF’).

• *Vị trí*: Thể hiện vị trí địa lý và môi trường xung quanh căn nhà (ví dụ: ‘Neighborhood’).

Có nhiều giả thuyết cho rằng các đặc trưng thuộc nhóm *vị trí* có thể là yếu tố quan trọng nhất ảnh hưởng đến giá nhà. Nghiên cứu này sẽ tiến hành phân tích để xác định các đặc trưng có tác động mạnh nhất nhằm phục vụ quá trình dự đoán giá nhà.

**3. Phương pháp nghiên cứu**

Dự án dự đoán giá nhà này sẽ được thực hiện bằng Python Notebook với các thư viện chính sau:

• **NumPy**: Hỗ trợ xử lý mảng dữ liệu.

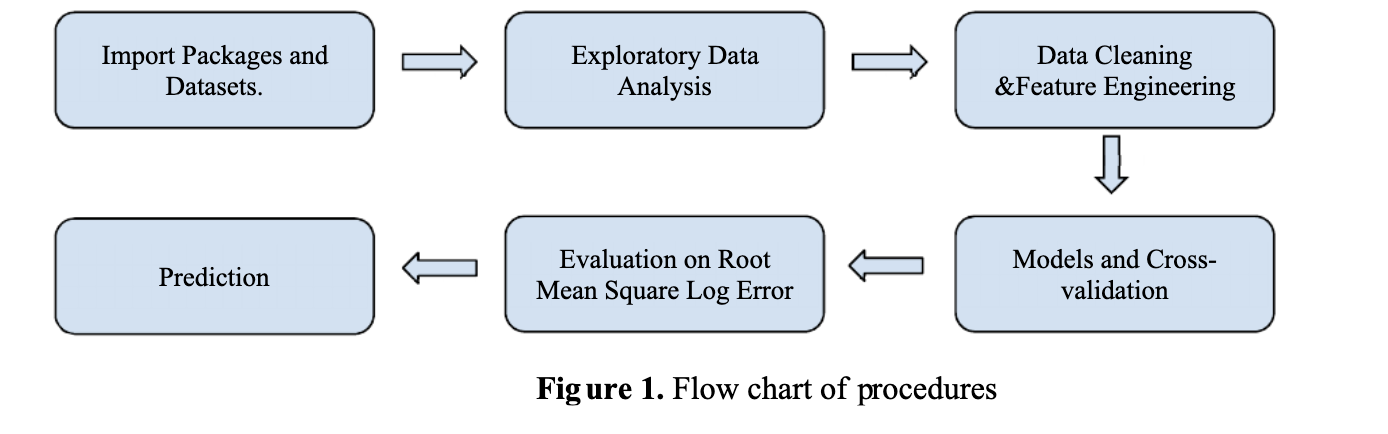
• **pandas**: Dùng để tiền xử lý dữ liệu.

• **matplotlib** và **seaborn**: Hỗ trợ phân tích dữ liệu khám phá (EDA) và trực quan hóa dữ liệu.

• **SciPy**: Được sử dụng cho các phân tích thống kê như kiểm tra tính chuẩn (normality) và độ lệch (skewness).

• **scikit-learn**: Dùng để xây dựng và đánh giá các mô hình học máy.

Các bước cơ bản trong quy trình nghiên cứu được thể hiện trong Hình 1 dưới đây.



**3.1 Tập dữ liệu**

Tập dữ liệu **Ames House Price** được chọn từ Kaggle và đã được phân chia thành hai tập:

• **Tập huấn luyện** gồm 1460 căn nhà với các mã định danh khác nhau (‘Id’) và 81 đặc trưng, bao gồm cả biến mục tiêu ‘SalePrice’. Dữ liệu này được dùng để huấn luyện mô hình nhằm dự đoán giá bán (‘SalePrice’) của các căn nhà trong **tập kiểm tra**, gồm 1459 căn với 80 đặc trưng còn lại.

• Các biến trong tập dữ liệu được chia thành hai loại: **biến số** (‘numerical’) và **biến phân loại** (‘categorical’).

• Cột ‘Id’ được đặt làm chỉ mục (index) để thuận tiện trong việc ghi nhận dữ liệu, nhưng không được sử dụng như một đặc trưng để dự đoán giá nhà.

**3.2 Phân tích dữ liệu khám phá (Exploratory Data Analysis - EDA)**

Để hiểu rõ hơn về tập dữ liệu, cần xem xét ý nghĩa của các đặc trưng và mức độ liên quan của chúng đối với biến mục tiêu ‘SalePrice’. Một số câu hỏi quan trọng cần được đặt ra trong quá trình phân tích:

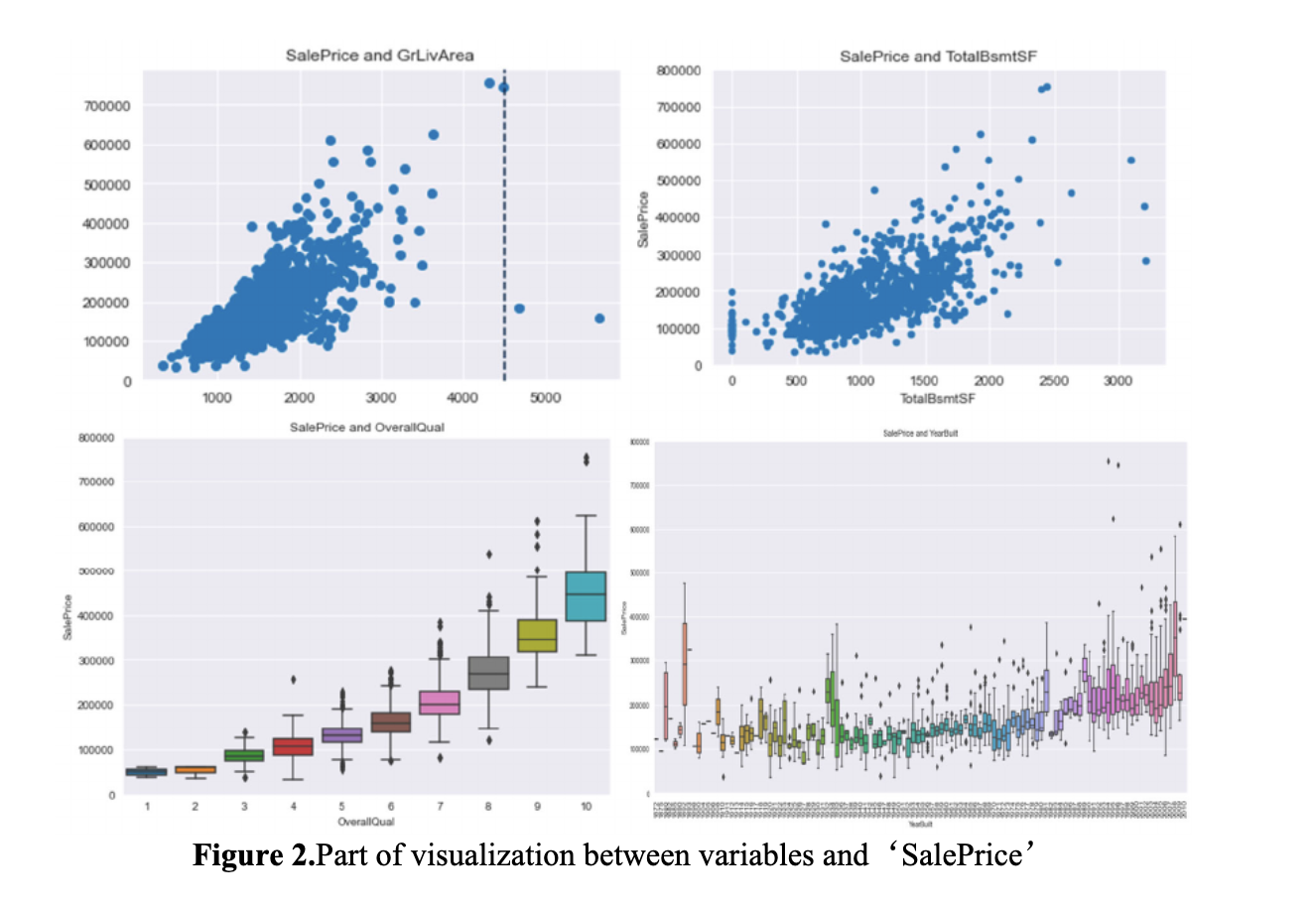
• **Những đặc trưng nào thường được quan tâm khi mua nhà?**

• **Đặc trưng nào có ảnh hưởng lớn hơn đến giá nhà?**

• **Có đặc trưng nào mang thông tin đã được mô tả trong các biến khác không?**

Dựa trên các yếu tố này, giả thuyết được đưa ra rằng **bốn đặc trưng ‘OverallQual’, ‘YearBuilt’, ‘TotalBsmtSF’, ‘GrLivArea’** có thể đóng vai trò quan trọng trong việc dự đoán giá nhà. Đây là hai biến thuộc nhóm **‘space’** và hai biến thuộc nhóm **‘building’**, điều này có phần bất ngờ vì thông thường, các đặc trưng thuộc nhóm **‘location’** mới được kỳ vọng có ảnh hưởng lớn nhất.

Để trực quan hóa mức độ liên quan giữa các đặc trưng này với ‘SalePrice’, các biểu đồ **scatter plot** và **boxplot**, như được thể hiện trong Hình 2, sẽ được sử dụng.



Kết quả phân tích cho thấy rằng cả **‘GrLivArea’** và **‘TotalBsmtSF’** đều có mối quan hệ tuyến tính với **‘SalePrice’**. Cả hai đều có tương quan dương, nghĩa là khi diện tích **GrLivArea** hoặc **TotalBsmtSF** tăng, giá bán **‘SalePrice’** cũng có xu hướng tăng theo.

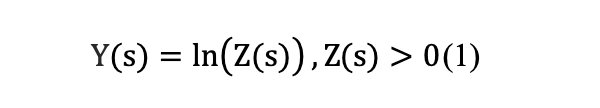
Tương tự, **‘OverallQual’** và **‘YearBuilt’** cũng thể hiện mối quan hệ với **‘SalePrice’**. Mối quan hệ này đặc biệt rõ ràng trong biểu đồ **boxplot** của **‘OverallQual’**, cho thấy giá bán tăng dần theo mức chất lượng tổng thể của ngôi nhà.

Ngoài ra, việc trực quan hóa dữ liệu còn giúp phát hiện **các điểm ngoại lai (outliers)**, có thể ảnh hưởng đáng kể đến mô hình. Trong trường hợp của **‘GrLivArea’**, biểu đồ **scatter plot** đầu tiên trong **Hình 2** cho thấy hai điểm dữ liệu có diện tích **GrLivArea** lớn nhưng giá bán **‘SalePrice’** lại thấp hơn nhiều so với phần lớn các điểm dữ liệu còn lại. Do chúng không đại diện chính xác cho xu hướng chung, hai điểm này được xem là ngoại lai và sẽ bị loại bỏ trong các phân tích tiếp theo.

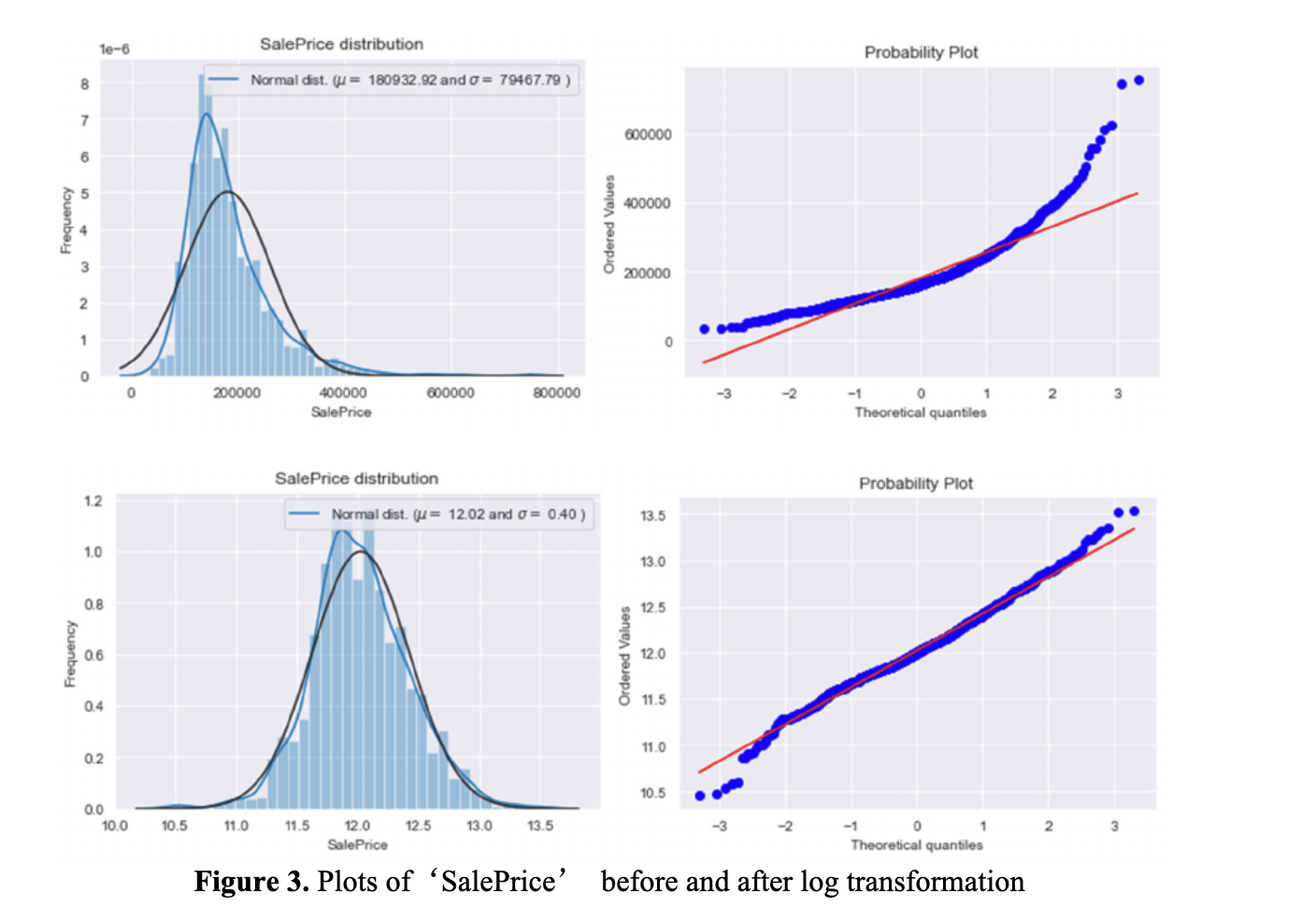
**Phân tích biến mục tiêu ‘SalePrice’**

Sau khi xem xét các đặc trưng quan trọng, bước tiếp theo là kiểm tra biến mục tiêu **‘SalePrice’**. Theo biểu đồ **phân phối (distribution plot)** và **xác suất (probability plot)** trong **Hình 3**, có thể thấy rằng **‘SalePrice’** có độ lệch phải (right skewed), nghĩa là phân phối không tuân theo quy luật chuẩn. Điều này có thể gây ảnh hưởng tiêu cực đến mô hình do sự tồn tại của các giá trị ngoại lai.

Để khắc phục, phương pháp **log transformation** được áp dụng, như thể hiện trong phương trình (1). Phương pháp này đơn giản là lấy **logarithm** của biến mục tiêu, thường được sử dụng khi phân phối của biến có độ lệch dương và tồn tại một số giá trị lớn bất thường.



Đây là trường hợp phù hợp với **‘SalePrice’**, vì vậy phương pháp **log transformation** được thực hiện để đưa phân phối của **‘SalePrice’** về dạng chuẩn, như thể hiện trong hai biểu đồ cuối cùng của **Hình 3**.

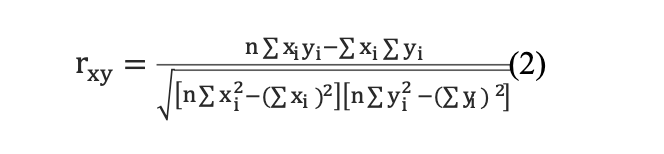


**3.3 Làm sạch dữ liệu và kỹ thuật đặc trưng**

Bộ dữ liệu huấn luyện và kiểm tra được gộp lại thành một tập dữ liệu chung **‘all\_data’**, gồm **2.917** căn nhà với **79** đặc trưng (sau khi loại bỏ cột **‘Id’** và biến mục tiêu **‘SalePrice’**). Biến mục tiêu **‘SalePrice’** được lưu riêng thành **‘y\_train’** để phục vụ quá trình làm sạch dữ liệu. Việc kết hợp cả hai bộ dữ liệu vào **‘all\_data’** giúp quá trình làm sạch trở nên thuận tiện hơn.

**Phân tích tương quan và vấn đề đa cộng tuyến**

Bước đầu tiên trong quá trình làm sạch dữ liệu là kiểm tra **tương quan Pearson** giữa các biến với nhau và với **‘SalePrice’**. Công thức tính tương quan Pearson được biểu diễn trong phương trình (2). Đây là công cụ thống kê phổ biến giúp đánh giá mức độ liên quan giữa hai biến mà không kết luận quan hệ nhân quả.



Hình **4** minh họa **10 đặc trưng có tương quan cao nhất với ‘SalePrice’**. Những đặc trưng đã được phân tích trước đó đều có mặt, xác nhận mức độ quan trọng của chúng trong việc dự đoán giá nhà.

Tuy nhiên, **đa cộng tuyến (multicollinearity)** là một vấn đề quan trọng cần xử lý. Đây là hiện tượng khi hai hoặc nhiều biến độc lập có mối tương quan cao với nhau, khiến mô hình mất đi tính chính xác do các biến không còn độc lập và làm giảm ý nghĩa thống kê của các biến độc lập.

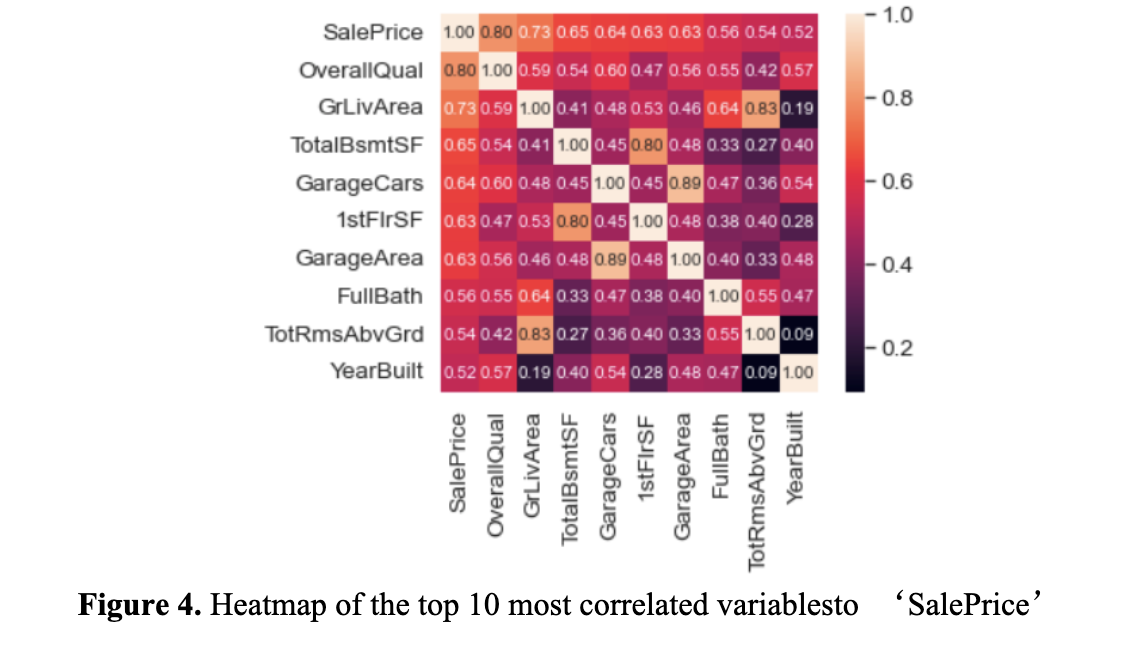
Để giảm đa cộng tuyến, một biểu đồ **heatmap** được sử dụng để phát hiện các cặp biến có mức tương quan cao. Theo Hình **5**, một số cặp biến có tương quan mạnh gồm:

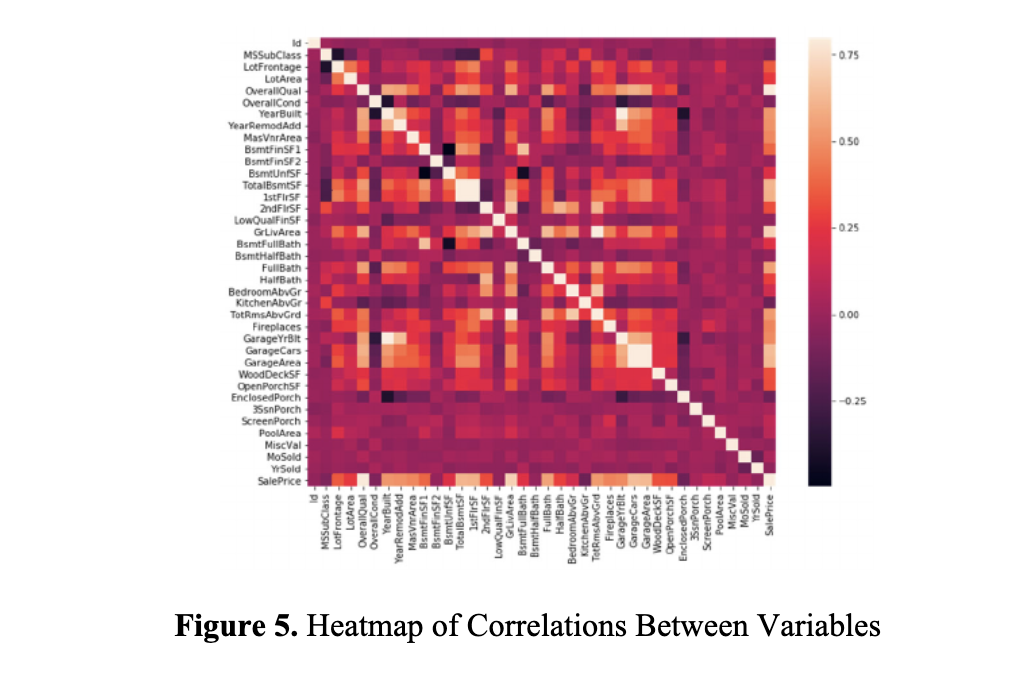
• **‘GarageCars’** và **‘GarageArea’**: Số lượng xe trong garage phụ thuộc trực tiếp vào diện tích garage, do đó chỉ giữ lại **‘GarageCars’** vì nó có tương quan cao hơn với **‘SalePrice’**.

• **‘TotalBsmtSF’** và **‘1stFlrSF’**: Hai biến này thể hiện mối tương quan mạnh do tổng diện tích tầng hầm thường tỷ lệ thuận với diện tích tầng một.

• **‘TotRmsAbvGrd’** và **‘GrLivArea’**: Tổng số phòng trên mặt đất có tương quan cao với diện tích không gian sống.

Ngoài ra, một số đặc trưng không có nhiều ý nghĩa hoặc ít ảnh hưởng đến giá nhà như **‘MasVnrType’**, **‘MasVnrArea’**, và **‘Utilities’** cũng được loại bỏ. Tổng cộng **6 biến** được xóa khỏi **‘all\_data’** để tăng hiệu suất và độ chính xác của mô hình dự đoán.



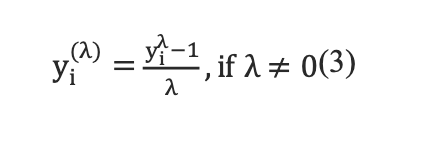


Tiếp theo, các giá trị bị thiếu được tìm ra và điền bằng các giá trị thích hợp. Có rất nhiều dữ liệu bị thiếu trong khá nhiều biến. Năm đặc trưng **‘PoolQC’**, **‘MiscFeature’**, **‘Alley’**, **‘Fence’**, **‘FireplaceQu’** đều có tỷ lệ giá trị thiếu trên **40%**, do đó cần loại bỏ không chỉ vì tỷ lệ thiếu quá cao mà còn bởi vì chúng dường như không phải là những đặc trưng quan trọng khi mọi người mua nhà. Hơn nữa, khi xem xét kỹ hơn các biến này, **‘PoolQC’**, **‘MiscFeature’** và **‘FireplaceQu’** có thể là những ứng viên tiềm năng của ngoại lệ do số lượng dữ liệu được ghi nhận rất ít.

Các đặc trưng còn lại có giá trị bị thiếu được điền bằng các giá trị thích hợp. Ví dụ, **‘LotFrontage’**, một biến số, được điền bằng **giá trị trung vị của ‘LotFrontage’ trong mỗi ‘Neighborhood’**; và các giá trị thiếu của một số đặc trưng phân loại liên quan đến tầng hầm được điền bằng **‘None’**, vì chúng có thể được hiểu là không có tầng hầm. Sau khi hoàn tất các bước trên, không còn giá trị thiếu nào trong **‘all\_data’**, với tổng cộng **68 đặc trưng**.

Sau khi xử lý giá trị thiếu, các đặc trưng phân loại được tách riêng với mục đích chuyển đổi sang dạng số thông qua **label encoding**, giúp máy tính có thể đọc và xử lý dữ liệu.

Sau đó, tất cả các đặc trưng đã ở dạng số và sẵn sàng để xử lý độ lệch, đây là một phép đo cho biết một biến bị lệch bao nhiêu so với phân phối chuẩn. Như đã đề cập ở phần trước khi chuyển đổi biến mục tiêu, độ lệch phần lớn là không mong muốn vì nó có thể dẫn đến phương sai quá lớn trong ước lượng. Phép biến đổi **Box-Cox**, được trình bày trong phương trình (3), là một phương pháp mạnh mẽ giúp ổn định phương sai và đưa biến về dạng phân phối chuẩn như đã thực hiện với **‘SalePrice’**. Do đó, phương pháp này được áp dụng cho các đặc trưng có độ lệch lớn hơn **0.75**.



**3.4 Mô hình và kiểm tra chéo**

Sau khi làm sạch dữ liệu, bước tiếp theo là tách lại tập huấn luyện và tập kiểm tra để tiến hành mô hình hóa. Tuy nhiên, **quá khớp (overfitting)** có thể xảy ra khi mô hình học cả những dao động ngẫu nhiên và nhiễu, dẫn đến việc mô hình khớp quá chặt với tập huấn luyện, gây ảnh hưởng tiêu cực đến độ chính xác của dự đoán. Để tránh điều này, **kiểm tra chéo (Cross-Validation)** là phương pháp tốt nhất, trong đó tập huấn luyện ban đầu được chia ngẫu nhiên thành các phần huấn luyện và một phần kiểm tra (validation), sau đó lặp lại nhiều lần để thu thập toàn bộ kết quả.

Trong dự án này, tập huấn luyện được chia thành **5 phần (folds)** và sai số căn bậc hai của logarit (**Root-Mean-Square Logarithmic Error - RMSLE**) được tính toán để đánh giá hiệu suất của mô hình.

Các mô hình được chọn để kết hợp với kiểm tra chéo bao gồm:

• **LASSO Regression**

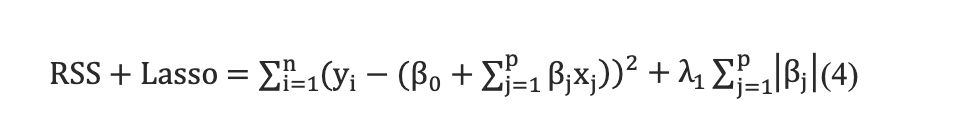
• **Elastic Net Regression**

• **Gradient Boosting Regression**

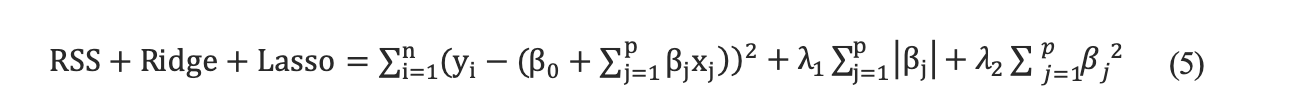
• **XGBoost**

• **LightGBM**

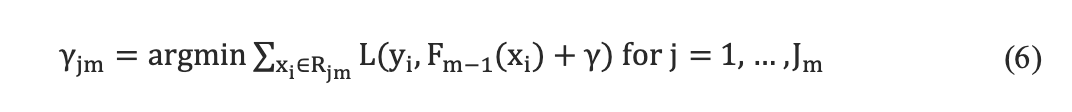
**LASSO Regression** và **Elastic Net Regression** là hai dạng mô hình **hồi quy có điều chuẩn (regularization models)**, giúp giảm thiểu phần dư (residuals). **LASSO**, như được biểu diễn trong phương trình (4), sử dụng kỹ thuật **thu nhỏ (shrinkage)**, trong đó các điểm dữ liệu được thu hẹp về một điểm trung tâm (trung bình) và áp dụng một hình phạt tương đương với tổng giá trị tuyệt đối của các trọng số để giảm độ phức tạp của mô hình, tránh quá khớp, đồng thời làm giảm ảnh hưởng của các đặc trưng gây **đa cộng tuyến (multicollinearity)** bằng cách thêm một hệ số phạt vào tổng bình phương phần dư (**Residual Sum of Squares - RSS**).



**Elastic Net Regression** tương tự như **LASSO Regression**, nhưng có thêm một hệ số phạt tương đương với tổng bình phương các trọng số, được biểu diễn trong phương trình (5).



**Gradient Boosting** là một thuật toán tập hợp, trong đó nhiều mô hình mới được thêm vào để sửa lỗi từ các mô hình trước và được kết hợp lại nhằm đạt kết quả tốt hơn. Thuật toán này còn được gọi là **gradient tree boosting**, vì tập hợp các mô hình bao gồm các cây quyết định. Nhằm mục đích giảm phần dư, mỗi lần một cây mới được thêm vào tập hợp cây. Sau đó, phương trình (6) được sử dụng để tính đầu ra cho từng lá sao cho tổng lỗi được tối thiểu hóa, trong đó các dự đoán trước đó cũng được tính đến.



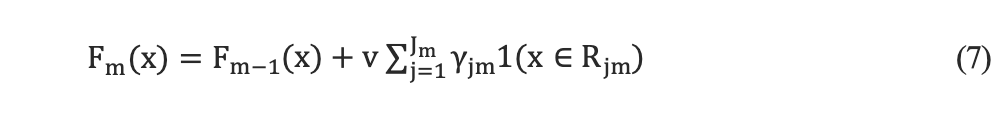
Cuối cùng, một dự đoán mới,  F\_m(x) , được tạo ra cho mỗi mẫu, bao gồm dự đoán trước đó và đầu ra của từng lá, được điều chỉnh bằng một tốc độ học, như được thể hiện trong phương trình (7).

Lastly, a new prediction, Fm(x), is made for each sample, which is composed by the last prediction

and the output for each leaf applied by a learning rate, shown in equation (7

Lastly, a new prediction, Fm(x), is made for each sample, which is composed by the last prediction

and the output for each leaf applied by a learning rate, shown in equation (7



XGBoosting và LightGBM đều tuân theo nguyên tắc của Gradient Boosting. XGBoosting là một biến thể được điều chỉnh nhiều hơn của Gradient Boosting. Đúng như tên gọi, nó đạt đến giới hạn tính toán cho các thuật toán cây tăng cường, sử dụng điều chuẩn tiên tiến để cải thiện khả năng tổng quát hóa của mô hình và kiểm soát hiện tượng overfitting.

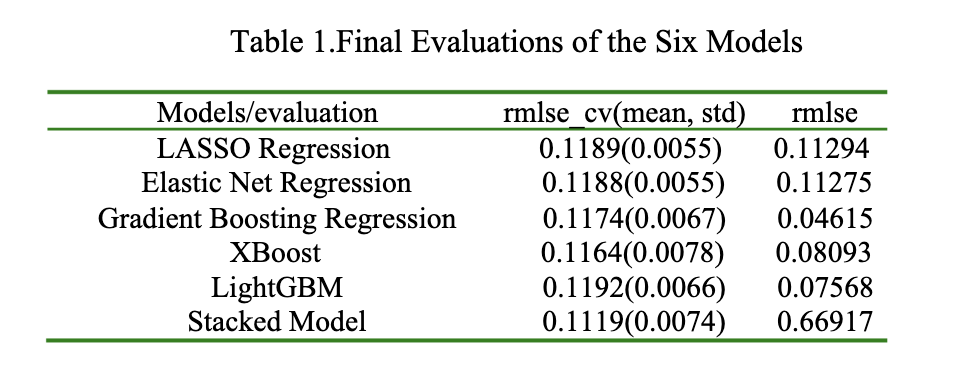
Tương tự như XGBoosting, LightGBM cũng sử dụng cây quyết định để hồi quy. So với sự phát triển cây theo mức của XGBoost, LightGBM sử dụng sự phát triển theo lá, giúp giảm tổn thất nhiều hơn và đạt độ chính xác cao hơn, đồng thời có tốc độ nhanh hơn. Tuy nhiên, overfitting có thể xảy ra, khiến XGBoost có thể tạo ra một mô hình vững chắc hơn [9].

Mô hình xếp chồng (Stacked model) là một kỹ thuật trong học tập tổ hợp (ensemble learning) kết hợp các phương pháp khác nhau để tạo ra dự đoán chính xác và ổn định hơn. Điều này quan trọng vì bằng cách kết hợp ưu điểm của nhiều thuật toán, độ biến động sẽ ít hơn so với một bộ hồi quy đơn lẻ. Trong dự án này, hàm ‘StackingCVRegressor’ được sử dụng để tìm mô hình xếp chồng và đây là mô hình cuối cùng để đưa ra dự đoán.

**3.5 Đánh giá trên lỗi căn bậc hai trung bình của logarit (RMLSE)**

Lỗi căn bậc hai trung bình của logarit (root-mean-square logarithmic error - RMLSE) là một trong những thước đo phổ biến nhất để đánh giá mô hình. Đây là phép biến đổi logarit của lỗi căn bậc hai trung bình (RMSE); do đó, có thể xem nó như là lỗi căn bậc hai trung bình của các giá trị dự đoán và giá trị thực đã được biến đổi logarit [10].

Để tránh lấy log của các giá trị có thể bằng không, một giá trị 1 được cộng thêm vào cả hai phía của giá trị thực tế và giá trị dự đoán trước khi thực hiện logarit tự nhiên. Các đánh giá cuối cùng cho các mô hình được thể hiện trong bảng 1, bao gồm cả giá trị trung bình và độ lệch chuẩn của lỗi căn bậc hai trung bình của logarit trong quá trình kiểm tra chéo (RMLSE\_CV) và lỗi căn bậc hai trung bình của logarit (RMLSE).

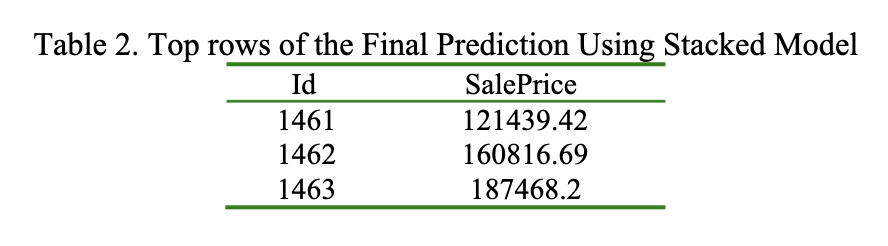


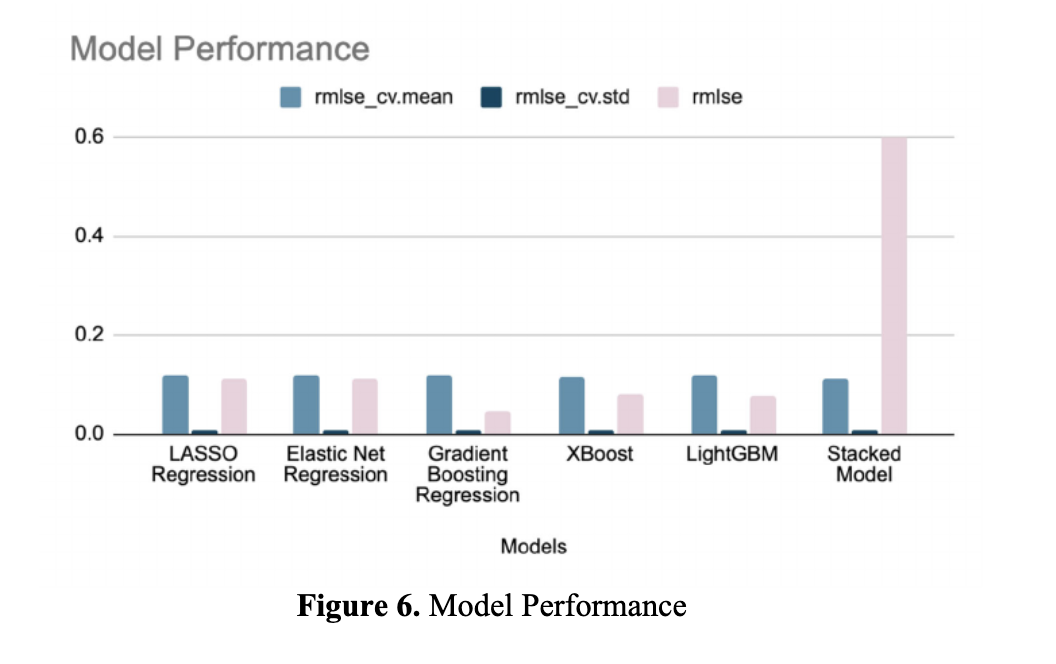
**4. Phân tích kết quả**

Các hàng đầu tiên của dự đoán cuối cùng được thể hiện trong Bảng 2, tuy nhiên, không có đáp án đúng nào được cung cấp. Cách duy nhất để kiểm tra kết quả là thông qua điểm số được đưa ra bởi cuộc thi trên Kaggle. Sau nhiều lần thử nghiệm, điểm số tốt nhất đạt được là 0.12444, xếp hạng trong top 14%.

Kết quả này cuối cùng là nhờ vào mô hình Stacked Model với vai trò là mô hình dự đoán cuối cùng, mặc dù sự cải thiện chỉ là 0.00028 so với kết quả khi sử dụng Gradient Boosting Regression làm mô hình dự đoán cuối cùng. Ban đầu, Gradient Boosting Regression được chọn vì nó có trung bình lỗi căn bậc hai trung bình của logarit (RMLSE) thấp trong quá trình kiểm tra chéo, cùng với độ lệch chuẩn thấp. Nó cũng có giá trị lỗi RMLSE nhỏ nhất trong tất cả các mô hình, điều này được thể hiện rõ trong Hình 6, trong khi đó, Stacked Model lại có hiệu suất kém nhất.

Kích thước mẫu nhỏ có thể là một yếu tố tiềm ẩn gây ra tình huống không mong đợi này. Hơn nữa, Stacked Model cũng mất quá nhiều thời gian để chạy, một điều kiện luôn không được khuyến khích. Do đó, Gradient Boosting Regression có thể là một mô hình phù hợp để sử dụng trong dự án này.





**5. Kết luận**

Một ngôi nhà có thể là tài sản giá trị nhất mà một người từng sở hữu. Giá của nó chắc chắn có ảnh hưởng lớn đến nhiều khía cạnh. Do đó, việc dự đoán giá nhà là rất cần thiết. Trong bài viết này, giá nhà tại Ames được dự đoán bằng các kỹ thuật hồi quy tiên tiến, bao gồm tương quan, kỹ thuật đặc trưng, mô hình điều chuẩn, mô hình tăng cường độ dốc và nhiều phương pháp khác.

Thông qua phân tích tương quan, giả thuyết về bốn đặc trưng có liên quan đáng kể đến biến mục tiêu không bị bác bỏ. Cụ thể, ‘OverallQual’ có hệ số tương quan là 0.8, ‘YearBuilt’ là 0.52, ‘TotalBsmtSF’ là 0.65 và ‘GrLivArea’ là 0.73. Ngoài ra, hiện tượng đa cộng tuyến được phát hiện qua biểu đồ nhiệt thể hiện tương quan giữa các biến.

Thông qua kỹ thuật đặc trưng, biến mục tiêu ‘SalePrice’ trở nên phân phối chuẩn hơn. Các giá trị bị thiếu được điền bằng các giá trị thích hợp. Độ lệch của các đặc trưng số được giảm thông qua phép biến đổi Box-Cox, và các đặc trưng phân loại được mã hóa nhãn để phục vụ cho việc học máy.

Thông qua quá trình mô hình hóa, mô hình hồi quy Gradient Boosting được chọn do có lỗi căn bậc hai trung bình của logarit (RMLSE) thấp nhất là 0.04615, cùng với giá trị trung bình RMLSE trong kiểm tra chéo thấp và thời gian chạy ngắn hơn.

Mặc dù mọi thứ có vẻ hợp lý, nhưng vẫn còn những vấn đề cần lưu ý cho các nghiên cứu tiếp theo. Các đặc trưng được phân loại thủ công, có nghĩa là chúng có thể được phân loại khác nhau tùy vào từng nhà nghiên cứu. Cách tiếp cận này có thể hữu ích trong việc xây dựng giả thuyết nhưng không thể áp dụng rộng rãi. Hơn nữa, các đặc trưng liên quan đến ‘vị trí’ lại có mức độ tương quan thấp hơn mong đợi, điều này có thể là do số lượng đặc trưng ‘vị trí’ trong tập dữ liệu còn ít. Đây luôn là một vấn đề trước khi các đặc trưng có thể được phân loại một cách hợp lệ.

Một vấn đề tiềm ẩn khác là kích thước mẫu nhỏ của tập dữ liệu được chọn, có thể là nguyên nhân khiến mô hình Stacked Model có lỗi căn bậc hai trung bình của logarit rất cao. Khi kích thước mẫu nhỏ, việc kết hợp mô hình theo cách thủ công với các trọng số khác nhau có thể là một phương án thay thế tốt hơn.

Cuối cùng, dự đoán chỉ dựa vào các đặc trưng của ngôi nhà, nhưng sự bất ổn của thị trường do các yếu tố xã hội và chính trị có thể bị bỏ qua. Trong lĩnh vực học máy, những vấn đề mới luôn xuất hiện, mang đến cơ hội cho các nhà nghiên cứu để tạo ra những cải tiến đáng kinh ngạc.

References

[1] Zhou, C. 2021. House price prediction using polynomial regression with particle swarm optimization.

Journal of Physics: Conference Series, 18(2).https://iopscience.iop.org/article/10.1088/1742-

6596/1802/3/032034/pdf

[2] R. A. Rahadi, S. K. Wiryono, D. P. Koesrindartotoor, and I. B. Syamwil, Factors influencing the price of

housing in Indonesia, Int. J. Hous. Mark. Anal., vol. 8, no. 2, pp. 169–188. 2015.

[3] Alfiyatin, A. N., Taufiq, H., Febrita, R. E., Mahmudy, W. F. 2017. Modeling house price prediction using

regression analysis and particle swarm optimization. International Journal of Advanced Computer Science

and Applications, 8(10),323-326.https://thesai.org/Downloads/Volume8No10/Paper\_42-

Modeling\_House\_Price\_Prediction\_using\_Linear\_Regression.pdf

[4] D. X. Zhu and K. L. Wei, The Land Prices and Housing Prices —— Empirical Research Based on Panel

Data of 11 Provinces and Municipalities in Eastern China, Int. Conf. Manag. Sci. Eng., 2013, pp. 2118–

2123.

[5] S. Kisilevich, D. Keim, and L. Rokach, A GIS-based decision support system for hotel room rate

estimation and temporal price prediction: The hotel brokers’ context, Decis. Support Syst., vol. 54, 2013,

pp. 1119– 1133.

[6] C. Y. Jim and W. Y. Chen. Value of scenic views: Hedonic assessment of private housing in Hong Kong,

Landsc. Urban Plan., vol. 91, no. 4, 2009, pp. 226–234.

[7] Marcelino, P.2017. Comprehensive Data Exploration with

Python.https://www.kaggle.com/code/pmarcelino/comprehensive-data-exploration-with-

python/notebook

[8] Serigne. 2017. Stacked Regressions: Top 4% on LeaderBoard.

https://www.kaggle.com/code/serigne/stacked-regressions-top-4-on-leaderboard/notebook#Modelling

[9] Saha, Sumit.2022. XGBoost vs LightGBM: How Are They Different.https://neptune.ai/blog/xgboost-vs-

lightgbm

[10] Gok, H. 2019. Metrics. https://hrngok.github.io/posts/metrics/

References

[1] Zhou, C. 2021. House price prediction using polynomial regression with particle swarm optimization.

Journal of Physics: Conference Series, 18(2).https://iopscience.iop.org/article/10.1088/1742-

6596/1802/3/032034/pdf

[2] R. A. Rahadi, S. K. Wiryono, D. P. Koesrindartotoor, and I. B. Syamwil, Factors influencing the price of

housing in Indonesia, Int. J. Hous. Mark. Anal., vol. 8, no. 2, pp. 169–188. 2015.

[3] Alfiyatin, A. N., Taufiq, H., Febrita, R. E., Mahmudy, W. F. 2017. Modeling house price prediction using

regression analysis and particle swarm optimization. International Journal of Advanced Computer Science

and Applications, 8(10),323-326.https://thesai.org/Downloads/Volume8No10/Paper\_42-

Modeling\_House\_Price\_Prediction\_using\_Linear\_Regression.pdf

[4] D. X. Zhu and K. L. Wei, The Land Prices and Housing Prices —— Empirical Research Based on Panel

Data of 11 Provinces and Municipalities in Eastern China, Int. Conf. Manag. Sci. Eng., 2013, pp. 2118–

2123.

[5] S. Kisilevich, D. Keim, and L. Rokach, A GIS-based decision support system for hotel room rate

estimation and temporal price prediction: The hotel brokers’ context, Decis. Support Syst., vol. 54, 2013,

pp. 1119– 1133.

[6] C. Y. Jim and W. Y. Chen. Value of scenic views: Hedonic assessment of private housing in Hong Kong,

Landsc. Urban Plan., vol. 91, no. 4, 2009, pp. 226–234.

[7] Marcelino, P.2017. Comprehensive Data Exploration with

Python.https://www.kaggle.com/code/pmarcelino/comprehensive-data-exploration-with-

python/notebook

[8] Serigne. 2017. Stacked Regressions: Top 4% on LeaderBoard.

https://www.kaggle.com/code/serigne/stacked-regressions-top-4-on-leaderboard/notebook#Modelling

[9] Saha, Sumit.2022. XGBoost vs LightGBM: How Are They Different.https://neptune.ai/blog/xgboost-vs-

lightgbm

[10] Gok, H. 2019. Metrics. https://hrngok.github.io/posts/metrics/

**References**

[1] Zhou, C. (2021). House price prediction using polynomial regression with particle swarm optimization. *Journal of Physics: Conference Series, 18(2)*. [Link](https://iopscience.iop.org/article/10.1088/1742-6596/1802/3/032034/pdf)

[2] Rahadi, R. A., Wiryono, S. K., Koesrindartotoor, D. P., & Syamwil, I. B. (2015). Factors influencing the price of housing in Indonesia. *International Journal of Housing Market and Analysis, 8(2),* 169–188.

[3] Alfiyatin, A. N., Taufiq, H., Febrita, R. E., & Mahmudy, W. F. (2017). Modeling house price prediction using regression analysis and particle swarm optimization. *International Journal of Advanced Computer Science and Applications, 8(10),* 323-326. [Link](https://thesai.org/Downloads/Volume8No10/Paper_42-Modeling_House_Price_Prediction_using_Linear_Regression.pdf)

[4] Zhu, D. X., & Wei, K. L. (2013). The Land Prices and Housing Prices – Empirical Research Based on Panel Data of 11 Provinces and Municipalities in Eastern China. *International Conference on Management Science and Engineering,* 2118–2123.

[5] Kisilevich, S., Keim, D., & Rokach, L. (2013). A GIS-based decision support system for hotel room rate estimation and temporal price prediction: The hotel brokers’ context. *Decision Support Systems, 54,* 1119–1133.

[6] Jim, C. Y., & Chen, W. Y. (2009). Value of scenic views: Hedonic assessment of private housing in Hong Kong. *Landscape and Urban Planning, 91(4),* 226–234.

[7] Marcelino, P. (2017). Comprehensive Data Exploration with Python. [Kaggle Notebook](https://www.kaggle.com/code/pmarcelino/comprehensive-data-exploration-with-python/notebook)

[8] Serigne. (2017). Stacked Regressions: Top 4% on Leaderboard. [Kaggle Notebook](https://www.kaggle.com/code/serigne/stacked-regressions-top-4-on-leaderboard/notebook#Modelling)

[9] Saha, S. (2022). XGBoost vs LightGBM: How Are They Different. [Neptune Blog](https://neptune.ai/blog/xgboost-vs-lightgbm)

[10] Gok, H. (2019). Metrics. [Blog](https://hrngok.github.io/posts/metrics/)