**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**ĐỒ ÁN**

**HỌC PHẦN MÁY HỌC ỨNG DỤNG**

**Đề tài**

**XÂY DỰNG MÔ HÌNH­­ MÁY HỌC ĐỂ DỰ ĐOÁN SỐ LƯỢNG XE ĐẠP ĐƯỢC THUÊ**

**Nhóm sinh viên thực hiện:**

**1. Nguyễn Quốc Việt B2111908**

**2. Trường Huỳnh Tú Như B2111893**

**3. Nguyễn Hồng Tuấn Phát B2111894**

**4. Lê Thị Tiến B2105647**

#### Cần Thơ, 4/2024

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**ĐỒ ÁN**

**HỌC PHẦN MÁY HỌC ỨNG DỤNG**

**Đề tài**

**XÂY DỰNG MÔ HÌNH­­ MÁY HỌC ĐỂ DỰ ĐOÁN SỐ LƯỢNG XE ĐẠP ĐƯỢC THUÊ**

**Giảng viên hướng dẫn: Nhóm sinh viên thực hiện:**

**Huỳnh Gia Khương 1. Nguyễn Q Việt B2111908**

**Nhóm: 6 2. Trương H.T Như B2111893**

**3. Nguyễn H.T Phát B2111894**

**4. Lê Thị Tiến B2105647**

#### Cần Thơ, 4/2024

**NHẬN XÉT CỦA GIẢNG VIÊN**

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

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

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

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

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

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

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

*Cần Thơ, ngày tháng năm*

(Ký và ghi rõ họ tên)

## MỤC LỤC

## Danh Mục Hình

## Danh Mục Bảng

**PHÂN CÔNG CÔNG VIỆC**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **MSSV** | **Họ tên SV** | **Công việc** | **Thời gian** | **Ký tên nhận việc** | **Ghi chú** |
| 1 | B2111908 | Nguyễn Quốc Việt | Trưởng nhóm, phân tích, xây dựng giải thuật, Làm Slide, Thuyết trình | 31/3 – 10/4 | A close up of a piece of paper  Description automatically generated |  |
| 2 | B2111893 | Trương Huỳnh Tú Như | Thư ký, phân tích, xây dựng giải thuật, Làm Slide, Thuyết trình | 31/3 – 10/4 | A close up of a signature  Description automatically generated |  |
| 3 | B2111893 | Nguyễn Hồng Tuấn Phát | phân tích, xây dựng giải thuật, Làm Slide, Thuyết trình | 31/3 – 10/4 | A piece of paper with blue writing  Description automatically generated |  |
| 4 | B21105647 | Lê Thị Tiến | phân tích, xây dựng giải thuật, Làm Slide, Thuyết trình | 31/3 – 10/4 | A close up of a paper  Description automatically generated |  |

**PHẦN NỘI DUNG**

## Mô tả dữ liệu

Tập dữ liệu Bike Sharing Demand được thu thập từ Kaggle, tập dữ liệu Bike Sharing Demand là một trong những cuộc thi của Kaggle diễn ra vào năm 2015 với câu hỏi được đặt ra là: Dựa vào data của một hãng cung cấp cho thuê xe đạp ở Washington D.C, người tham gia cần phải dự báo được số lượng xe đạp sẽ được thuê. Tập dữ liệu bao gồm 10886 mẫu và 12 đặc trưng. Trong đó count là thuộc tính mục tiêu cần dự đoán.

## 2. Ý nghĩa của dữ liệu

Từ việc phân tích các thuộc tính như *datetime, season, holiday, workingday, weather, temp, atemp, humidity, casual, registered* để dự đoán số lượng xe đạp được thuê ở cột thuộc tính mục tiêu là count.

## 3. Phân tích dữ liệu và lựa chọn mô hình

### 3.1. Phân tích dữ liệu

#### 3.1.1. Ý nghĩa của các đặc trưng

datetime: cho biết thời gian theo dịnh dạng %m/%d/%Y %H:%M:%hS

season: cho biết mùa

1: mùa xuân

2: mùa hè

3: mùa thu

4: mùa đông

holiday: Cho biết có phải ngày lễ hay không

0: không phải ngày lễ

1: ngày lễ

workingday: cho biết có phải là ngày làm việc không bao gồm cả cuối tuần và ngày lễ

1: ngày làm việc

0: ngày không làm việc

weather: Cho biết thời tiết

1: Trời trong, có mây, có mây rải rác

2: Sương mù + Đầy mây, sương mù + ít mây, sương mù

3: Tuyết nhẹ, mưa nhẹ + sấm sét + mây rải rác, mưa nhẹ + mây rải rác

4: Mưa lớn + mảnh băng + sấm sét + sương mù, tuyết + sương mù

temp: Nhiệt độ thực tế, tính theo độ Celsius (độ °C)

atemp: Nhiệt độ mà con người cảm nhận được, tính theo độ Celsius (độ °C)

humidity: Độ ẩm tương đối

windspeed: Tốc độ gió

casual: Số lượng xe mà khách vãng lai thuê

registered: Số lượng xe thuê mà khách có đăng ký dịch vụ thuê

count: Tổng số lượng xe đạp được thuê

month: tháng (1-12)

year: năm (2011, 2012)

hour: giờ trong ngày (0-23 giờ)

weekday: ngày trong tuần

0: thứ 2

1: thứ 3

2: thứ 4

3: thứ 5

4: thứ 6

5: thứ 7

6: chủ nhật

#### 3.1.2 Tiền xử lý và phân tích dũ liệu

A screenshot of a computer screen

Description automatically generated

Hình 1: Bản đồ nhiệt của các đặc trưng trước khi xử lý

[A screenshot of a computer

Description automatically generated](#hinh5)

Hình 2: Kiểm tra xem có thuộc tính nào có giá trị bị thiếu không

Do dữ liệu từ Bike Sharing Demand đã được xử lý làm sạch nên không có giá trị nào bị thiếu. Vì vậy không cần xử lý giá trị thiếu.

A screenshot of a computer screen

Description automatically generated

Hình 3: Kiểu dữ liệu của các đặc trưng trước khi xử lý

Do biến datetime có kiểu dữ liệu là object nên sẽ chuyển biến *datetime* sang các biến *year, month, weekday, hour* để thực hiện tính toán.

Do đề tài là dự đoán số lượng xe đạp được thuê mà biến *casual* + *registered* = *count* nên sẽ xoá biến *casual* + *registered* để xây dựng mô hình.

A screenshot of a computer

Description automatically generated

Hình 4: Bản đồ nhiệt của các đặc trưng sau khi xử lý

A screenshot of a computer

Description automatically generated

Hình 5:Kiểu dữ liệu của các đặc trưng sau khi xử lý

Vậy tập thuộc tính để xây dựng mô hình là: *season, holiday, workingday, holiday, weather, temp, atemp, humidity, windspeed, hour, month, weekday, year, count*. Trong đó *count* là thuộc tính mục tiêu cần dự đoán. Sau đây là biểu đồ phân phối của các thuộc tính.

A graph showing the different seasons

Description automatically generated with medium confidence

Hình 6: Biểu đồ phân phối giá trị của thuộc tính season

Dựa vào biểu đồ trên ta thấy dữ liệu *season* được phân phối đều nhau.

A white rectangular object with black text

Description automatically generated

Hình 7: Biểu đồ phân phối giá trị của thuộc tính holiday

Dựa vào biểu đồ trên ta thấy dữ liệu *holiday* phân phối ở cột not holiday tương đối cao trong khi cột holiday chiếm rất ít.

A white rectangular object with black text

Description automatically generated

Hình 8: Biểu đồ phân phối giá trị của thuộc tính workingday

Biểu đồ trên cho thấy dữ liệu *workingday* được phân phối chủ yếu ở cột working day.

A graph showing the different weather conditions

Description automatically generated with medium confidence

Hình 9: Biểu đồ phân phối giá trị của thuộc tính weather

Biểu đồ trên cho thấy dữ liệu weather được phân phối tương đối cao ở cột *Clear* (trời trong), rải rác ở các cột *Mist* (sương mù) và *Light Snow* (tuyết nhẹ), hầu như không có ở cột *Heavy Rain* (mưa lớn).

A graph showing the distribution of temperature

Description automatically generated

Hình 10: Biểu đồ phân phối giá trị của thuộc tính temp

Biểu đồ trên cho thấy dữ liệu temp được phân phối rải rác ở các cột, trong đó tương đối cao ở khoảng 10°C đến 30°C.

A graph of a bar graph

Description automatically generated with medium confidence

Hình 11: Biểu đồ phân phối giá trị của thuộc tính atemp

Biểu đồ trên cho thấy dữ liệu *atemp* được phân phối gần tương tự như *temp*.

A graph showing the amount of humidity

Description automatically generated

Hình 12: Biểu đồ phân phối giá trị của thuộc tính humidity

Trong biểu đồ, ta thấy hầu hết các ngày trong dữ liệu humidity đều có độ ẩm từ 50 đến 90.

A graph of a distribution of windspeed

Description automatically generated

Hình 13: Biểu đồ phân phối giá trị của thuộc tính windspeed

Biểu đồ trên cho thấy dữ liệu *windspeed* dường như được phân phối chủ yếu ở khoảng 5 đến 20.

A diagram of a distribution of hours

Description automatically generated

Hình 14: Biểu đồ phân phối giá trị của thuộc tính hour

Dựa vào biểu đồ trên ta thấy dữ liệu hour được phân phối khá đều ở các cột tuy nhiên có vài cột lại khá cao.

A graph of blue bars with white text

Description automatically generated

Hình 15: Biểu đồ phân phối giá trị của thuộc tính month

Dựa vào biểu đồ ta thấy dữ liệu *month* được phân phối đều nhau.

A graph showing the number of days and months

Description automatically generated with medium confidence

Hình 16: Biểu đồ phân phối giá trị của thuộc tính weekday

Biểu đồ trên cho thấy dữ liệu *weekday* được phân phối tương đối đều nhau.

A white rectangular object with text

Description automatically generated

Hình 17: Biểu đồ phân phối giá trị của thuộc tính year

Dựa vào biểu đồ trên ta thấy dữ liệu year được phân phối đều nhau ở cả 2 năm 2011 và 2012.

A graph of a number of bars

Description automatically generated

Hình 18: Biểu đồ phân phối giá trị của thuộc tính count

Biểu đồ trên cho thấy dữ liệu count được phân phối chủ yếu ở khoảng 0 đến 200 và giảm dần qua các cột.

A graph showing different seasons

Description automatically generated

Hình 19: Biểu diễn mối quan hệ giữa Season và Count

Nhu cầu xe được thuê cao nhất vào mùa thu (fall) và thấp nhất vào mùa xuân (spring).

A graph of a graph with a purple and blue rectangle

Description automatically generated with medium confidence

Hình 20: Biểu diễn mối quan hệ giữa Holiday và Count

Nhu cầu thuê xe vào ngày bình thường dường như cao hơn ngày lễ.

A blue and purple squares

Description automatically generated

Hình 21: Biểu diễn mối quan hệ giữa Workingday và Count

Nhu cầu xe được thuê của working day cao hơn not working day.

A graph showing different weather conditions

Description automatically generated

Hình 22: Biểu diễn mối quan hệ giữa Weather và Count

Nhu cầu thuê xe khi trời trong (clear) cao nhất, trong khi có tuyết nhẹ (light snow) sẽ ít được thuê.

A graph showing a graph of a temperature

Description automatically generated with medium confidence

Hình 23: Biểu diễn mối quan hệ giữa Temp và Count

Nhu cầu cao khi thời tiết tốt và nhiệt độ dễ chịu (20°C đến 30°C).

A graph showing a number of blue dots

Description automatically generated

Hình 24: Biểu diễn mối quan hệ giữa Atemp và Count

Nhu cầu thuê xe cao khi nhiệt độ trong khoảng 20°C đến 40°C.

A graph showing the amount of humidity

Description automatically generated

Hình 25: Biểu diễn mối quan hệ giữa Humidily và Count

Nhu cầu thuê xe cao khi độ ẩm nằm trong khoảng 30 đến 80.

A graph showing a number of wind speed

Description automatically generated

Hình 26: Biểu diễn mối quan hệ giữa Windspeed và Count

Nhu cầu thuê xe đạp cao khi tốc độ gió trong khoảng 10 đến 30.

A graph showing the year and year

Description automatically generated

Hình 27: Biểu diễn mối quan hệ giữa Year và Count

Dựa trên biểu đồ ta thấy năm 2012 số lượng xe đạp được thuê cao hơn nhiều so với nằm 2011.

### 3.2. Lựa chọn mô hình

Sử dụng nghi thức Hold-out và K-fold để phân chia tập dữ liệu và 5 mô hình: Phương pháp tổng hợp mô hình, KNeighborsRegressor, LinearReGressor, RandomForestRegressor, DecisionTreeRegressor để huấn luyện tập dữ liệu.

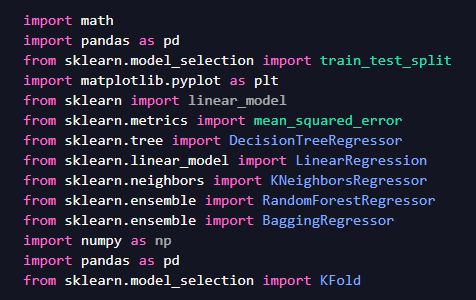
## 4. Cấu hình máy tính

|  |  |
| --- | --- |
| **CPU** | **Intel Core i3-8145U 2.10Ghz** |
| **RAM** | **8GB** |
| **ROM** | **SSD 512GB** |
| **GPU** | **Intel UHD Graphics 620** |

### 5. Huấn luyện và Kết quả thực nghiệm

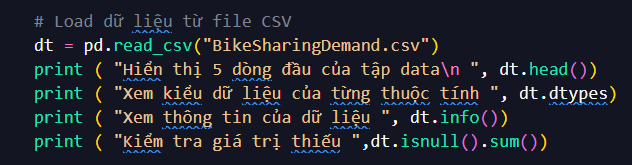
## 5.1. Các bước huấn luyện đầu tiên

**Bước 1:** Nhập tất cả thư viện cần thiết.



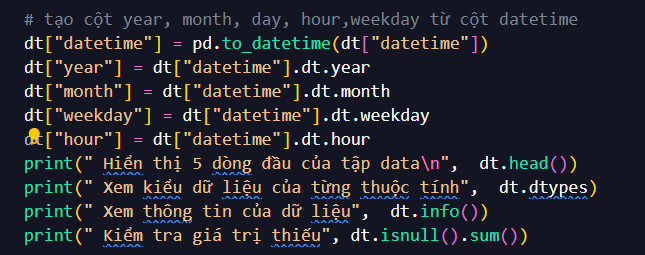
Hình 28: Nhập thư viện cần thiết

**Bước 2:** Đọc file BikeSharingDemand.csv và in ra các thông tin cần thiết.

****

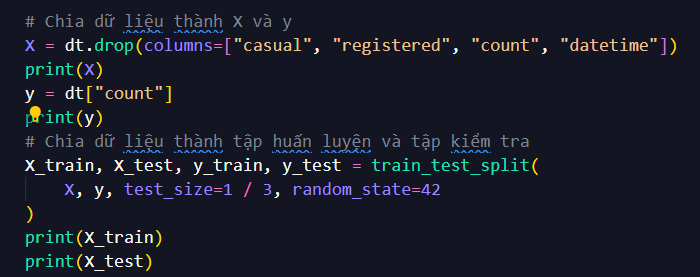
Hình 29: Đọc tập tin BikesharingDemand

**Bước 3:** Xử lý data



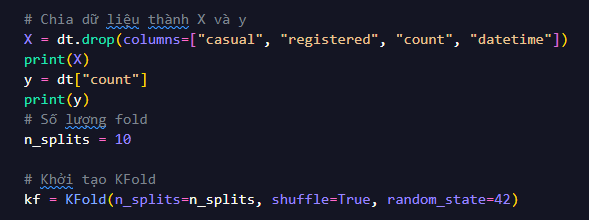
Hình 30: Tiến hành xử lý dữ liệu

**Bước 4:** Chia tập dữ liệu theo Hold-out train 2/3 và test 1/3. Nếu chia Hold-out thì thực hiện bước 4.



Hình 31: Chia tập dữ liệu theo Hold-out

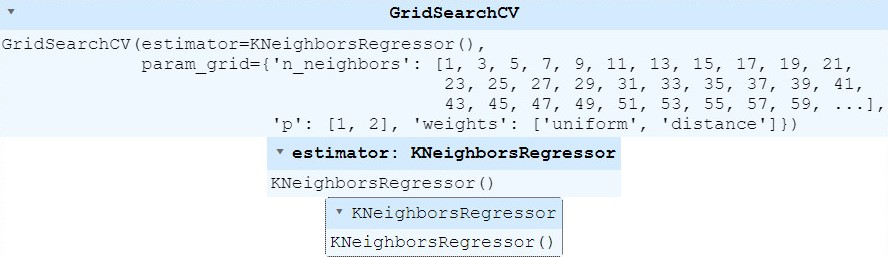
**Bước 5:** Chia tập dữ liệu theo K-fold. Nếu chia K-fold thì thực hiện bước 5.



Hình 32: Chia tập dữ liệu theo K-fold

### 5.2. KneighborsRegressor

Sử dụng GridSearchCV để tìm ra tham số tốt nhất cho mô hình, ta được:



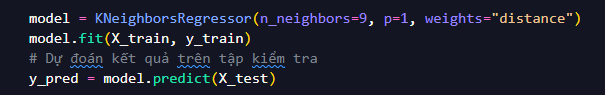
Hình 33: Kết quả của GridSearchCV đối với KneighborsRegressor

Kết quả sau khi tìm tham số tốt nhất:



Hình 34: Kết quả tham số tốt nhất sau khi chạy GridSearchCV

Huấn luyện mô hình:

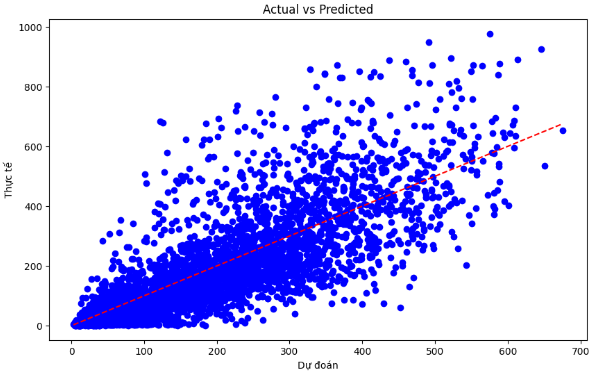


Hình 35: Huấn luyện mô hình KneighborsRegressor

Hold-out:



Hình 36: Kết quả của KneighborsRegressor với nghi thức Hold-out

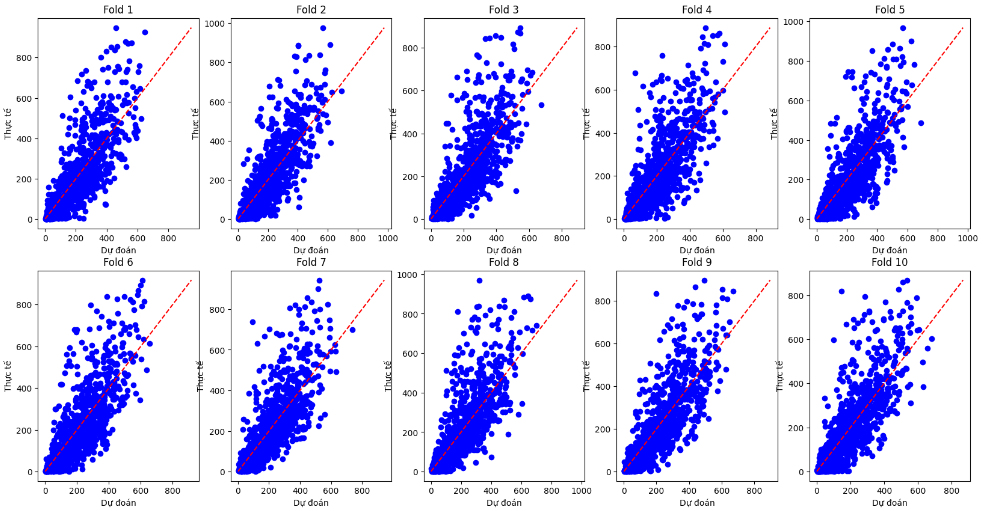


Hình 37: Đồ thị sau khi dự đoán theo Hold-out

K-fold:



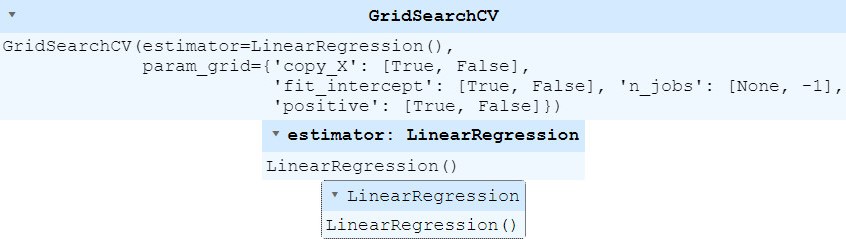
Hình 38: Kết quả của KneighborsRegressor với nghi thức K-fold



Hình 39: Đồ thị sau khi dự đoán theo K-fold

### 5.3. Linear Regression

Sử dụng GridSearchCV để tìm ra tham số tốt nhất cho mô hình, ta được:



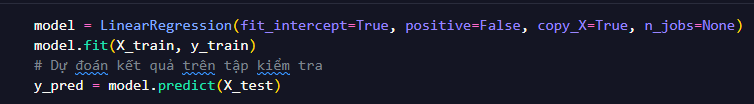
Hình 40: Kết quả của GridSearchCV đối với Linear Regression

Kết quả sau khi tìm tham số tốt nhất:



Hình 41: Kết quả tham số tốt nhất sau khi chạy GridSearchCV

Huấn luyện mô hình:

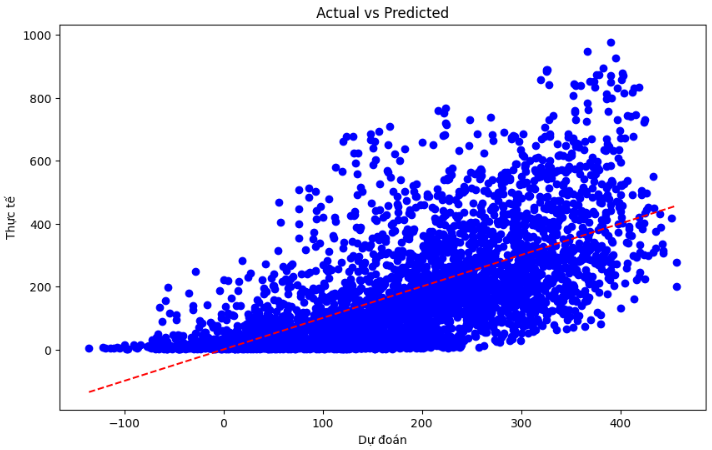


Hình 42: Huấn luyện mô hình Linear Regression

Hold-out:



Hình 43: Kết quả của Linear Regression với nghi thức Hold-out

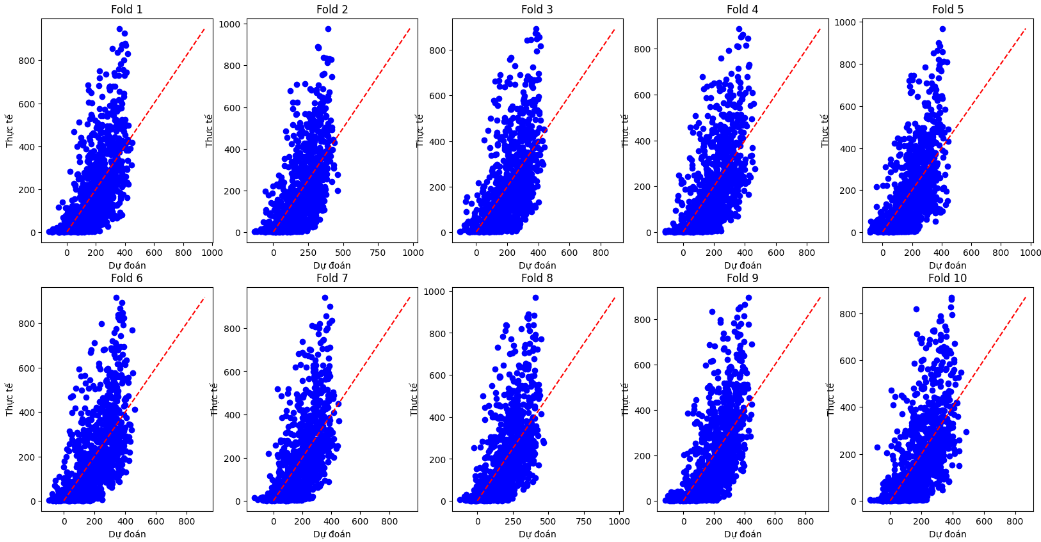


Hình 44: Đồ thị sau khi dự đoán theo Hold-out

K-fold:



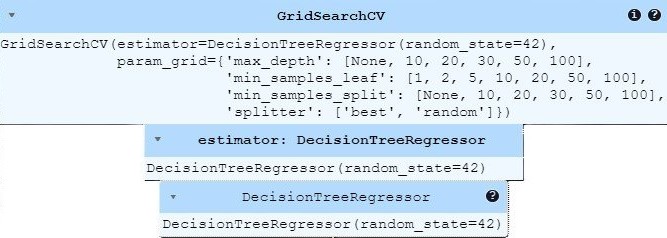
Hình 45: Kết quả của Linear Regression với nghi thức K-fold



Hình 46: Đồ thị sau khi dự đoán theo K-fold

### 5.4. DecisionTreeRegressor

Sử dụng GridSearchCV để tìm ra tham số tốt nhất cho mô hình, ta được:



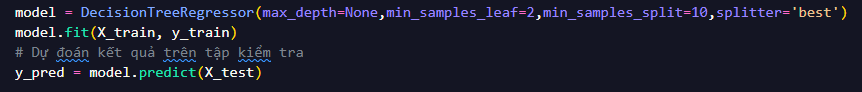
Hình 47: Kết quả của GridSearchCV đối với DecisionTreeRegressor

Kết quả sau khi tìm tham số tốt nhất:



Hình 48: Kết quả tham số tốt nhất sau khi chạy GridSearchCV

Huấn luyện mô hình:

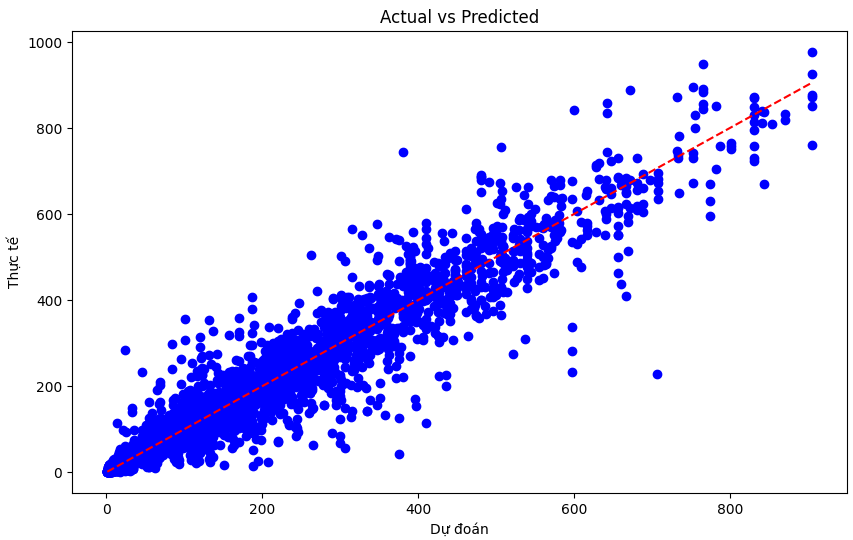


Hình 49: Huấn luyện mô hình DecisionTreeRegressor

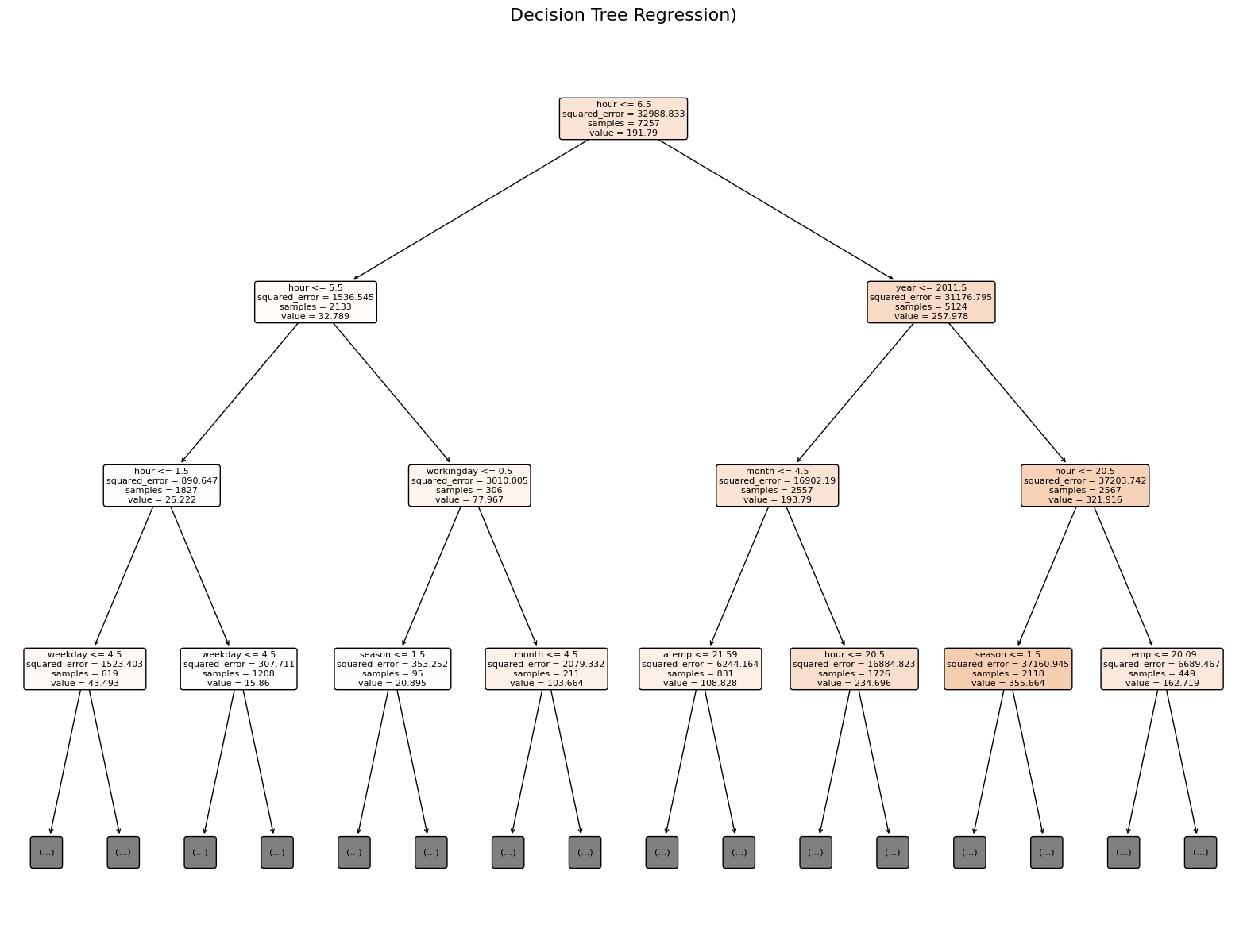
Hold-out:



Hình 50: Kết quả của DecisionTreeRegressor với nghi thức Hold-out



Hình 51: Đồ thị sau khi dự đoán theo Hold-out

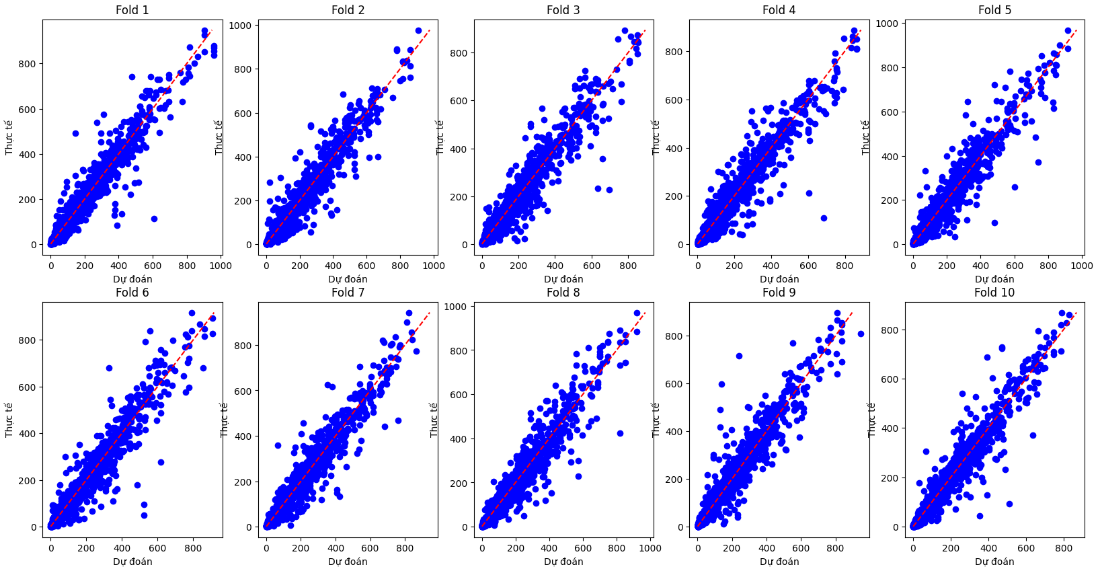


Hình 52: Cây quyết định sau khi dự đoán theo Hold-out

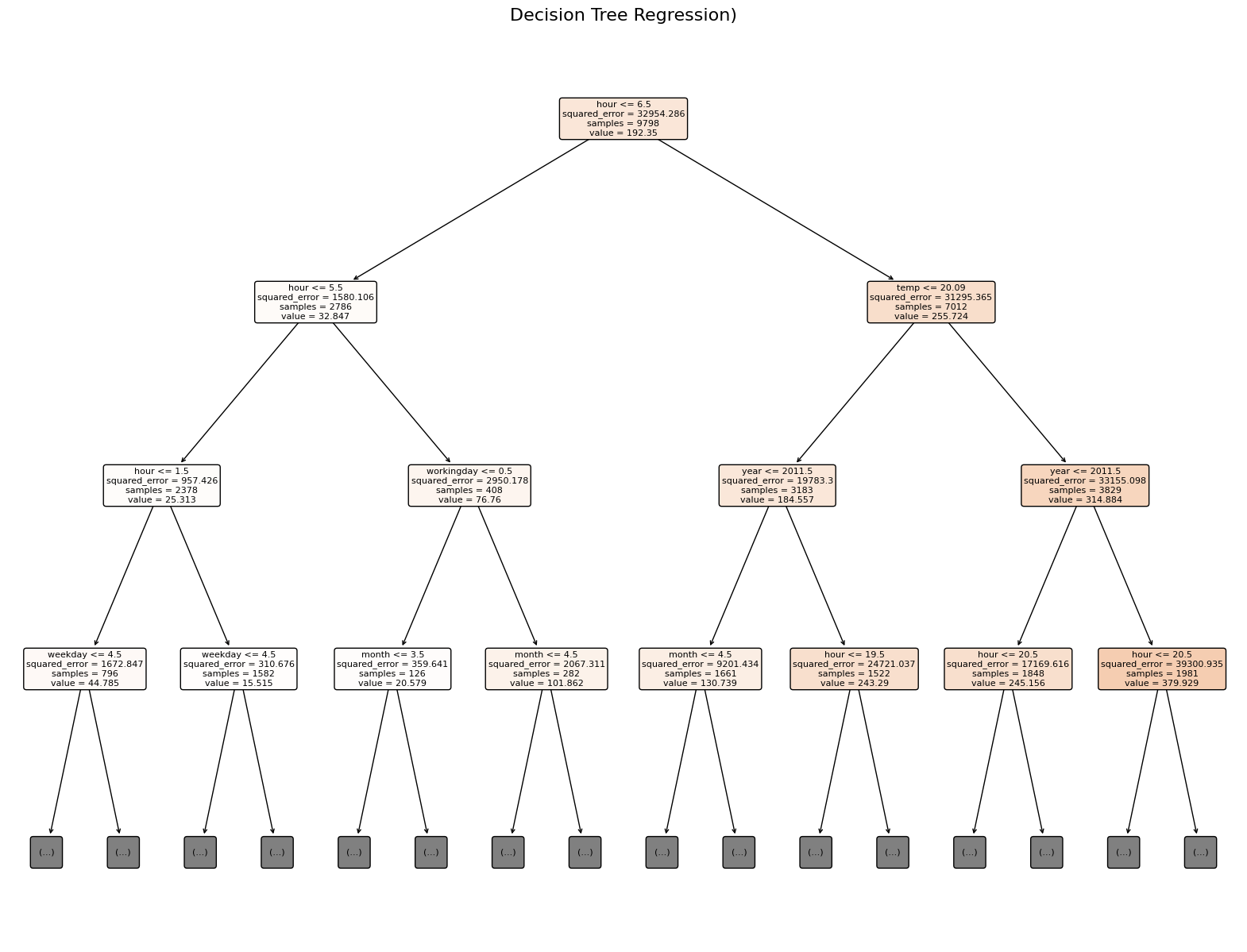
K-fold:



Hình 53: Kết quả của DecisionTreeRegressor với nghi thức K-fold



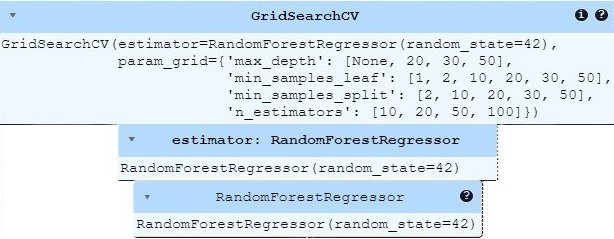
Hình 54: Đồ thị sau khi dự đoán theo K-fold



Hình 55: Cây quyết định sau khi dự đoán theo K-fold

### 5.5. RandomForestRegressor

Sử dụng GridSearchCV để tìm ra tham số tốt nhất cho mô hình, ta được:



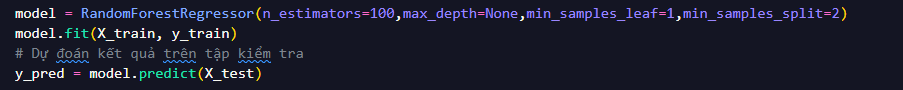
Hình 56: Kết quả của GridSearchCV đối với RandomForestRegressor

Kết quả sau khi tìm tham số tốt nhất:



Hình 57: Kết quả tham số tốt nhất sau khi chạy GridSearchCV

Huấn luyện mô hình:

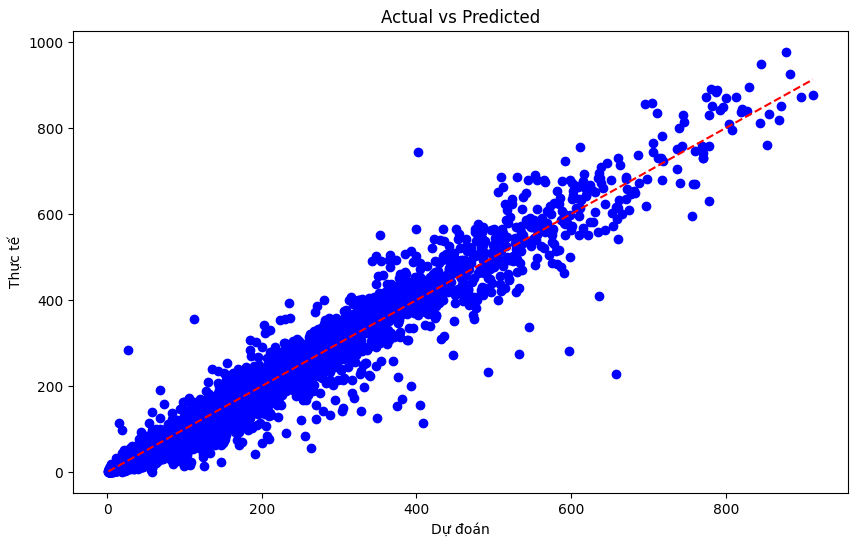


Hình 58: Huấn luyện mô hình RandomForestRegressor

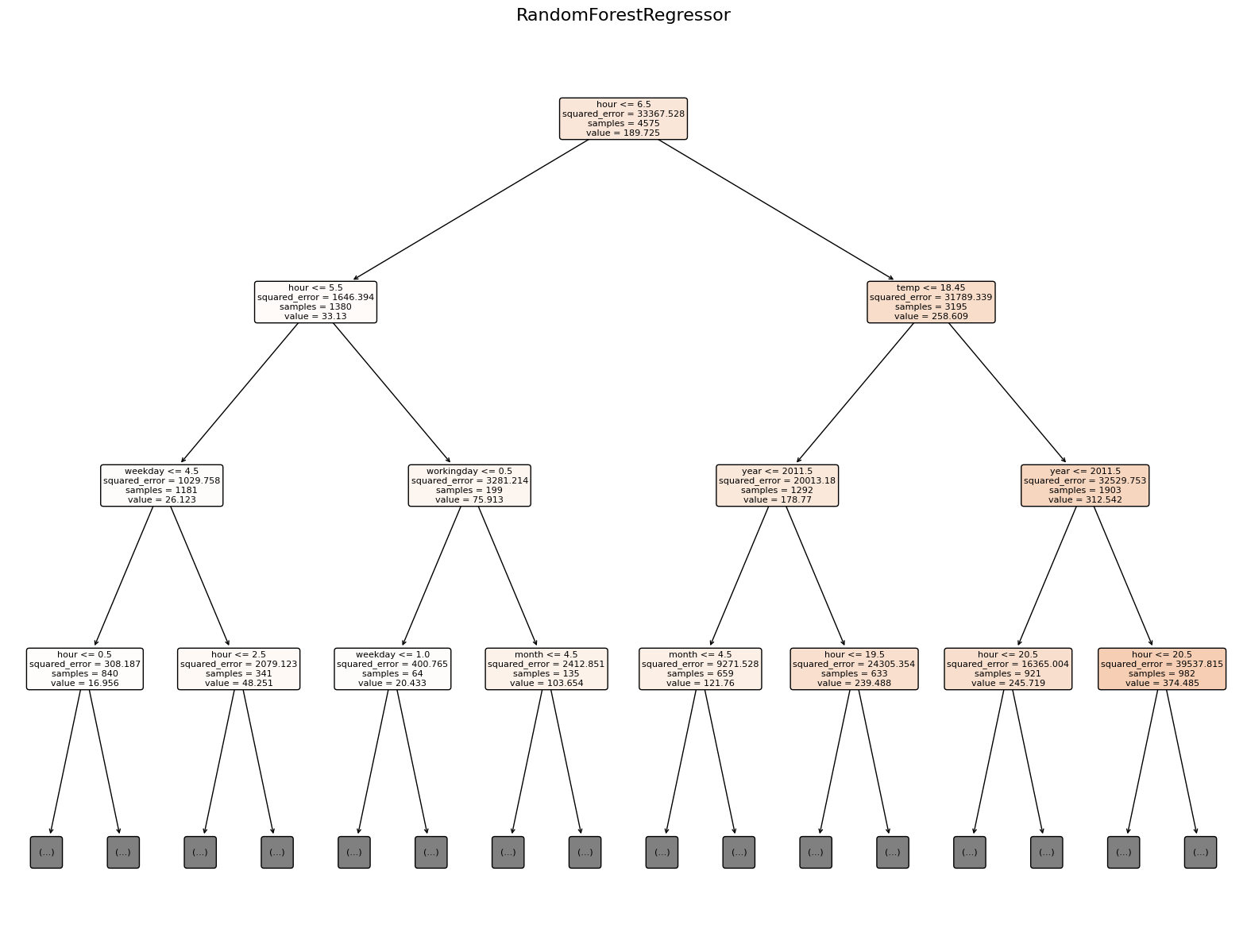
Hold-out:



Hình 59: Kết quả của RandomForestRegressor với nghi thức Hold-out



Hình 60: Đồ thị sau khi dự đoán theo Hold-out

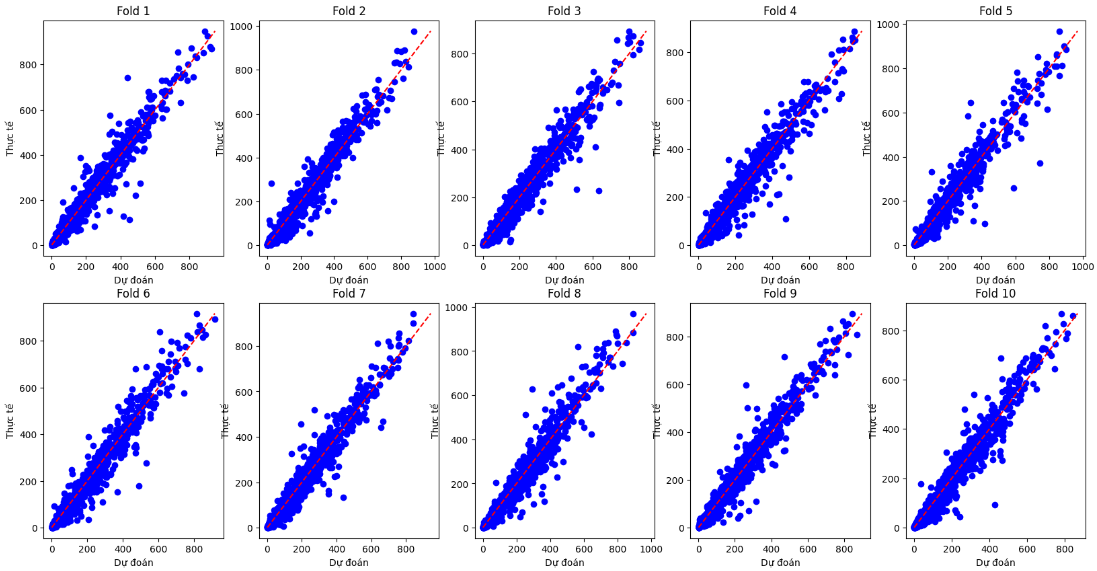


Hình 61: Cây quyết định sau khi dự đoán theo Hold-out

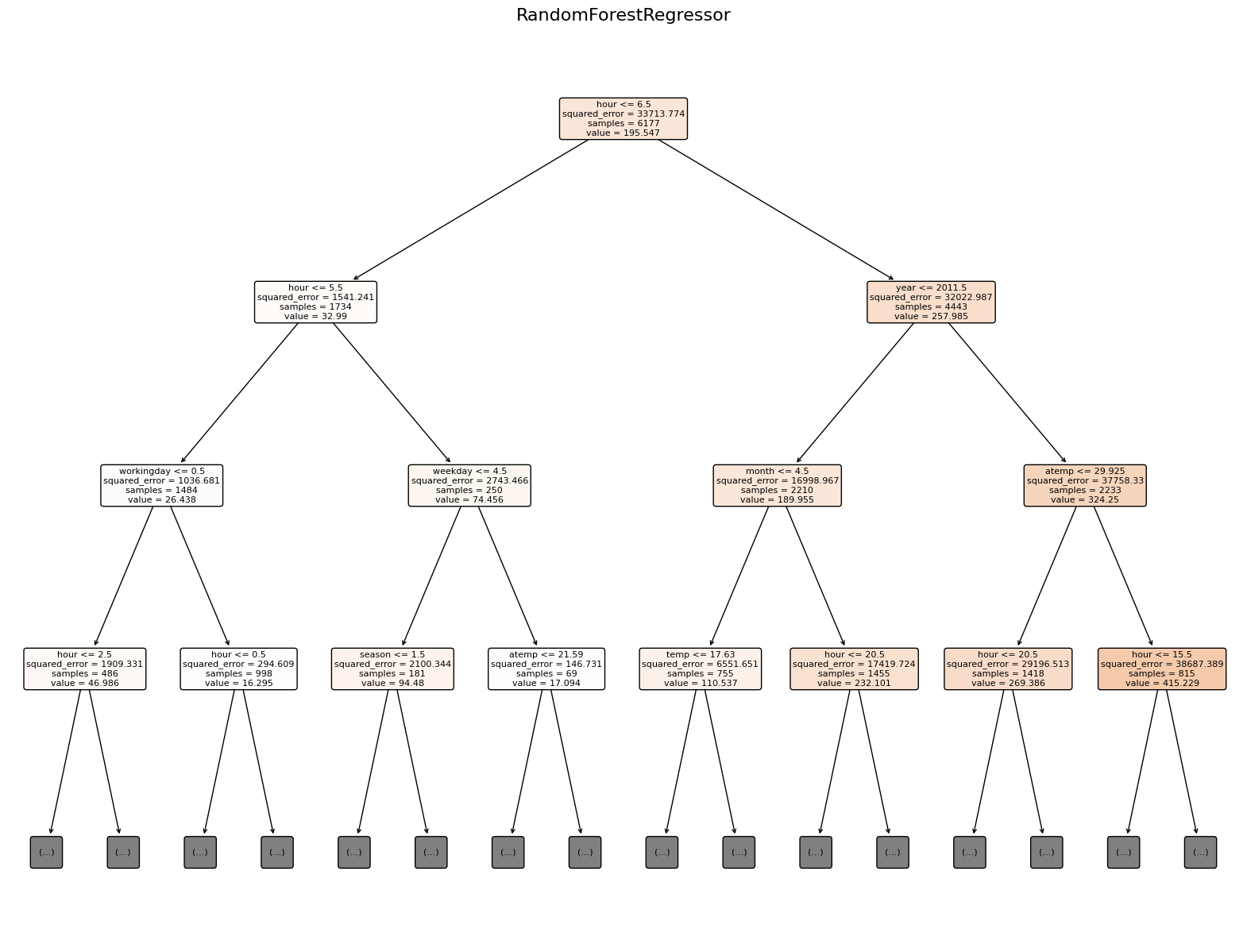
K-fold:



Hình 62: Kết quả của RandomForestRegressor với nghi thức K-fold



Hình 63: Đồ thị sau khi dự đoán theo K-fold

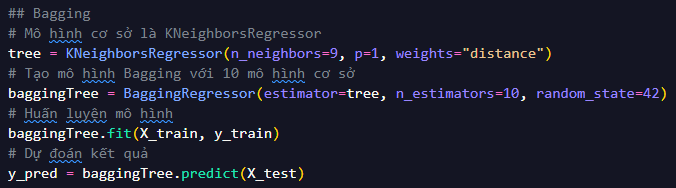
****

Hình 64: Cây quyết định sau khi dự đoán theo K-fold

### 5.6. Phương pháp tổng hợp mô hình (BaggingRegressor)

#### 5.6.1. Phương pháp tổng hợp mô hình với mô hình cơ sở là KneighborsRegressor

Huấn luyện mô hình bằng phương pháp BaggingRegressor với mô hình cơ sở là KneighborsRegressor:

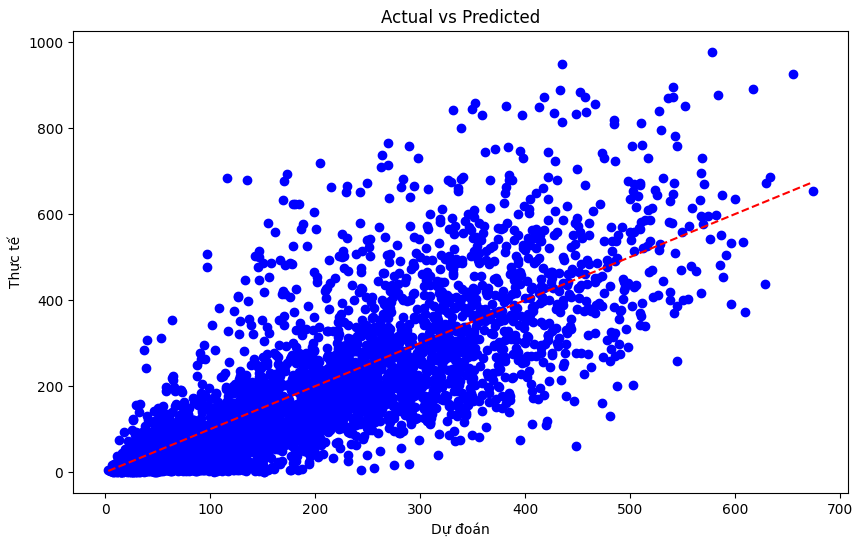


Hình 65: Huấn luyện mô hình BaggingRegressor với mô hình cơ sở là KneighborsRegressor

Hold-out:



Hình 66: Kết quả của KneighborsRegressor với nghi thức Hold-out

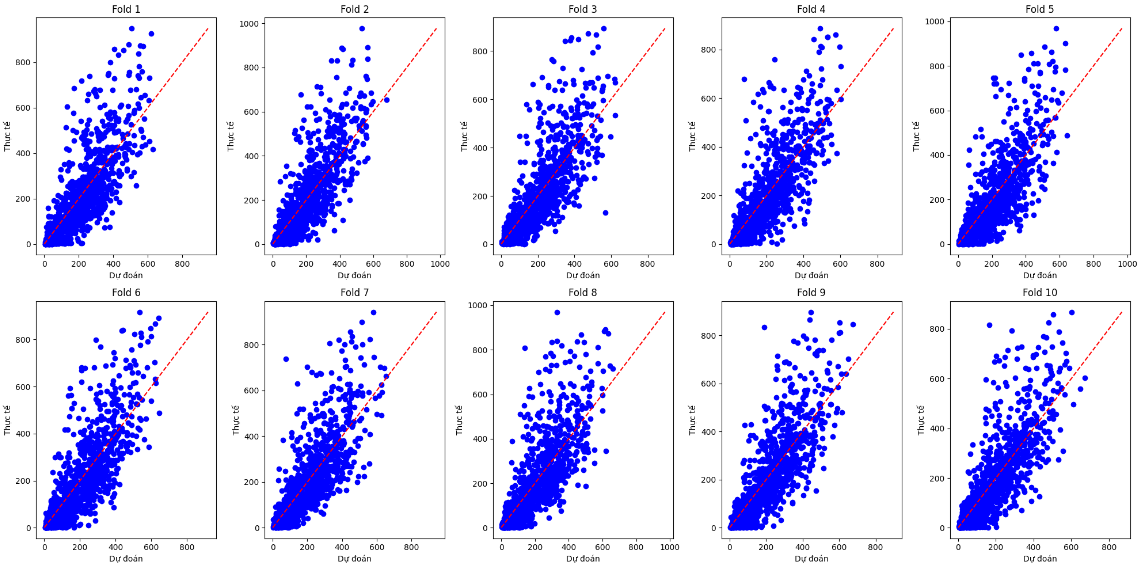


Hình 67: Đồ thị sau khi dự đoán theo Hold-out

K-fold:



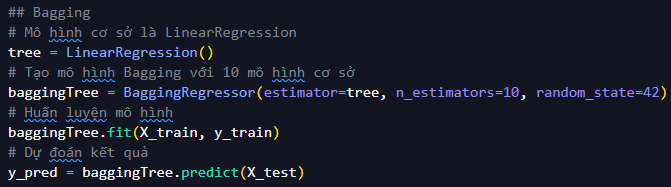
Hình 68: Kết quả của KneighborsRegressor với nghi thức K-fold



Hình 69: Đồ thị sau khi dự đoán theo K-fold

#### 5.6.2. Phương pháp tổng hợp mô hình với mô hình cơ sở là LinearRegression

Huấn luyện mô hình bằng phương pháp BaggingRegressor với mô hình cơ sở là Linear Regression:

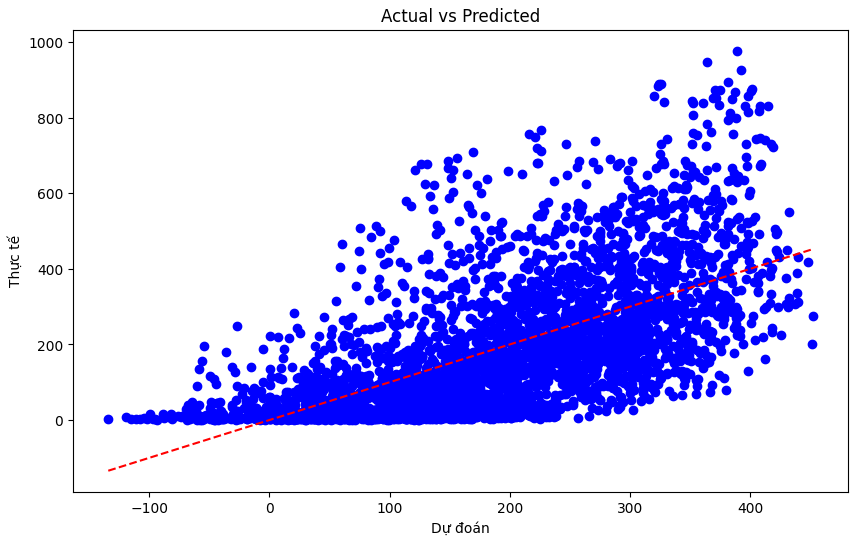


Hình 70: Huấn luyện mô hình BaggingRegressor với mô hình cơ sở là LinearRegression

Hold-out:



Hình 71: Kết quả của Linear Regression với nghi thức Hold-out

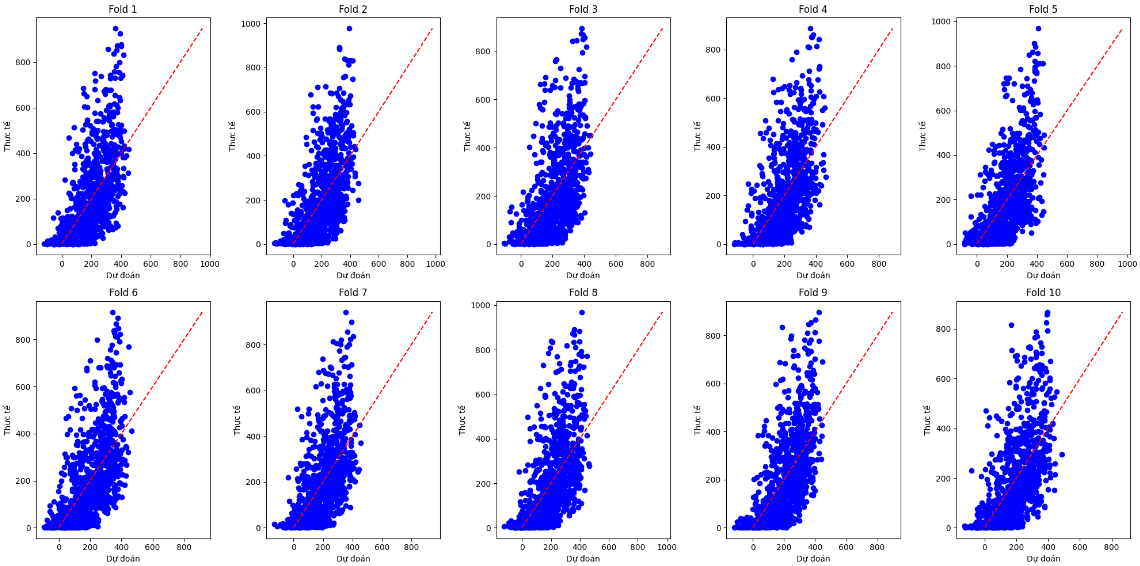


Hình 72: Đồ thị sau khi dự đoán theo Hold-out

K-fold:



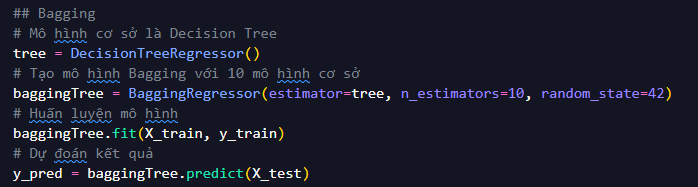
Hình 73: Kết quả của LinearRegression với nghi thức K-fold



Hình 74: Đồ thị sau khi dự đoán theo K-fold

#### 5.6.3. Phương pháp tổng hợp mô hình với mô hình cơ sở là DecisionTreeRegressor

Huấn luyện mô hình bằng phương pháp BaggingRegressor với mô hình cơ sở là DecisionTreeRegressor:

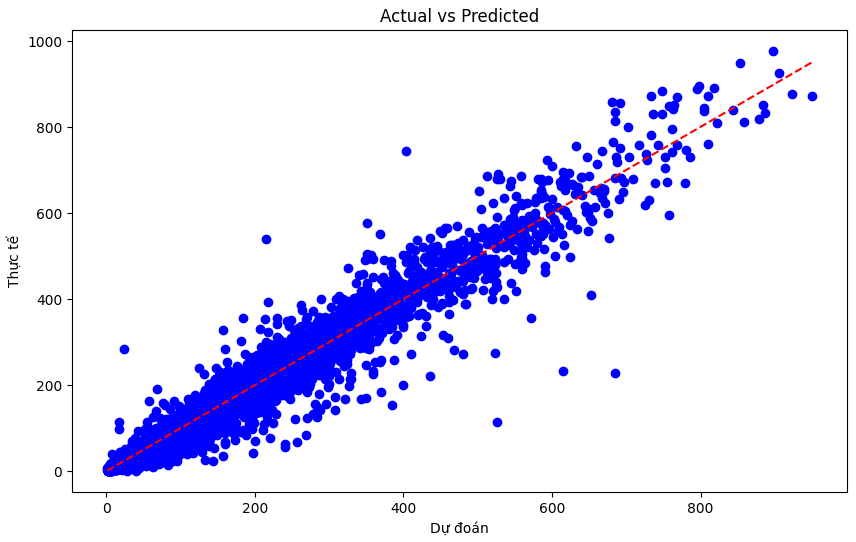


Hình 75: Huấn luyện mô hình BaggingRegressor với mô hình cơ sở là DecisionTreeRegressor

Hold-out:



Hình 76: Kết quả của DecisionTreeRegressor với nghi thức Hold-out

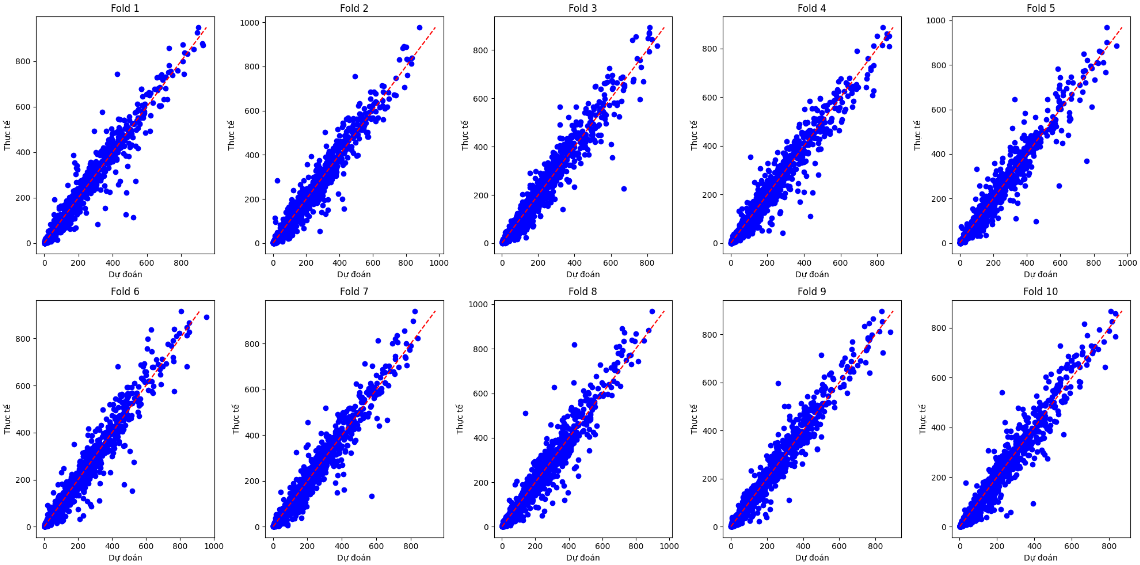


Hình 77: Đồ thị sau khi dự đoán theo Hold-out

K-fold:



Hình 78: Kết quả của DecisionTreeRegressor với nghi thức K-fold



Hình 79: Đồ thị sau khi dự đoán theo K-fold

## 6. Đánh giá mô hình

### 6.1 Đánh giá mô hình hồi quy

Với nghi thức Hold-out với tỷ lệ huấn luyện là 2/3 và kiểm tra là 1/3. Dưới đây là bảng so sánh kết quả khi chạy các giải thuật với nghi thích Hold-out.

Bảng 1: Bảng so sánh kết quả khi chạy các giải thuật với nghi thức Hold-out

|  |  |  |
| --- | --- | --- |
| Algorithm | MSE | RMSE |
| KNeighborsRegressor | 12031.71 | 109.69 |
| LinearRegression | 19820.64 | 140.79 |
| DecisionTreeRegressor | 2768.01 | 52.61 |
| RandomForestRegressor | 1679.90 | 40.99 |
| Bagging KNeighborsRegressor | 12251.95 | 110.69 |
| Bagging LinearRegression | 19850.00 | 140.89 |
| Bagging DecisionTreeRegressor | 1836.58 | 42.86 |

Với nghi thức K-fold chia tập dữ liệu thành 10 phần, sau đó tính giá trị trung bình với 10 phần trên. Dưới đây là bảng so sánh kết quả khi chạy các giải thuật với nghi thích K-fold.

Bảng 2: Bảng so sánh kết quả khi chạy các giải thuật với nghi thức K-fold

|  |  |  |
| --- | --- | --- |
| Algorithm | MSE | RMSE |
| KNeighborsRegressor | 11310.67 | 106.30 |
| LinearRegression | 20083.26 | 141.69 |
| DecisionTreeRegressor | 2680.10 | 51.74 |
| RandomForestRegressor | 1592.16 | 39.87 |
| Bagging KNeighborsRegressor | 11456.64 | 106.98 |
| Bagging LinearRegression | 20088.30 | 141.71 |
| Bagging DecisionTreeRegressor | 1837.31 | 42.80 |

### 6.2. Nhận xét kết quả thực nghiệm

Sau khi chạy các giải thuật trên ta thấy không có sự trên lệch lớn khi chạy tập dữ liệu bằng nghi thức Hold-out hay K-fold. Tuy nhiên có sự chênh lệch khá lớn giữa các giải thuật như KneighborsRegressor, LinearRegression, Bagging KneighborsRegressor, Bagging LinearRegression với các giải thuật như DecisionTreeRegressor, RandomForestRegressor, Bagging DecisionTreeRegressor.

Dựa trên [*Bảng so sánh kết quả khi chạy các giải thuật với nghi thức K-fold*](#BangKFold) và [*Bảng so sánh kết quả khi chạy các giải thuật với nghi thức Hold-out*](#BangHoldOut)*,* ta thấy các giải thuật khá tốt nhưDecisionTreeRegressor, RandomForestRegressor, Bagging DecisionTreeRegressor giá trị MSE chỉ dao động khoảng từ 1500-3000 và giá trị RMSE dao động khoảng 35-60. Trong khi đó các giải thuật KneighborsRegressor, LinearRegression, Bagging KneighborsRegressor, Bagging LinearRegression có giá trị MSE dao động khoảng 10000-20000 và giá trị RMSE dao động khoảng 100-150.

Do đó giải thuật tốt nhất của chúng tôi sau khi chạy các giải thuật trên là RandomForestRegressor với giá trị MSE là 1592.16, còn giá trị RMSE là 39.87.

## PHẦN KẾT LUẬN

### 1. Kết quả đạt được

Sau khi áp dụng các mô hình máy học khác nhau để huấn luyện, chúng tôi đã tìm ra được mô hình hồi quy tốt nhất cho tập dữ liệu Bike Sharing Demand là RandomForestRegressor.

A screenshot of a data

Description automatically generated

Hình 80: Kết quả áp dụng mô hình RandomForestRegressor vào tập dữ liệu thực tế

### 2. Hướng phát triển

* Thu thập thêm dữ liệu
* Tìm ra cách tiền xử lý dữ liệu tốt hơn
* Tìm ra các giải thuật tốt hơn hoặc các tham số tốt hơn tham số hiện tại
* Có thể hướng đến các đề tài khác như: Phân tích ảnh hưởng của thời tiết đến việc thuê xe, Phân tích hành vi thuê xe của các nhóm khách hàng khác nhau, So sánh lượng xe thuê giữa các mùa và một số sự kiện đặc biệt (ngày lễ, cuối tuần).

# TÀI LIỆU THAM KHẢO

[1]. Tập dữ liệu huấn luyện được lấy từ: <https://www.kaggle.com/c/bike-sharing-demand>

[2]. Tham khảo các tham số của các hàm từ: <https://scikit-learn.org/stable/>

[3].