

**ĐẠI HỌC HUẾ**

**KHOA KỸ THUẬT VÀ CÔNG NGHỆ**

🙠🙟🕮🙝🙢

**Icon

Description automatically generated**

**BÁO CÁO**

**ĐỒ ÁN**

**Học kỳ II, năm học 2022 - 2023**

**Học phần:**

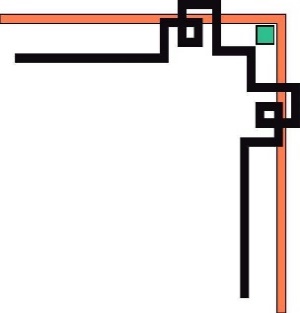
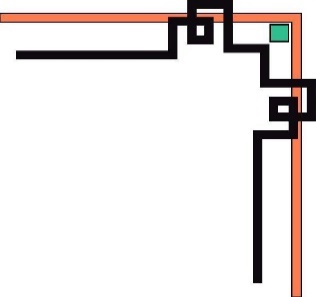
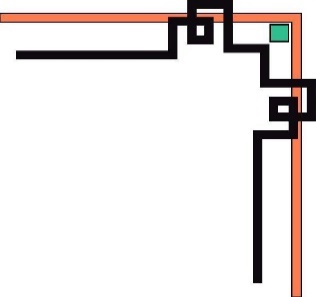
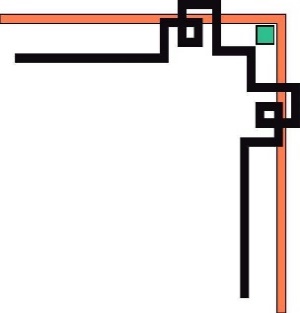
**CÁC THUẬT TOÁN TỐI ƯU CHO PHÂN TÍCH DỮ LIỆU**

**Đề tài: Gradient Descent cho hồi quy tuyến tính**

**đơn biến và ứng dụng**

|  |
| --- |
| **Số phách**  *(Do hội đồng chấm thi ghi)* |

**Thừa Thiên Huế, ngày….tháng….năm 2023**



**ĐẠI HỌC HUẾ**

**KHOA KỸ THUẬT VÀ CÔNG NGHỆ**

🙠🙟🕮🙝🙢

**Icon

Description automatically generated**

**BÁO CÁO**

**ĐỒ ÁN**

**Học kỳ II, năm học 2022 - 2023**

**Học phần:**

**CÁC THUẬT TOÁN TỐI ƯU CHO PHÂN TÍCH DỮ LIỆU**

**Đề tài: Gradient Descent cho hồi quy tuyến tính**

**đơn biến và ứng dụng**

**Giảng viên hướng dẫn: Hoàng Trọng Lợi**

**Sinh viên thực hiện: Hồ Minh Trí**

**Mã SV: 20E1020081**

**Lớp: Khoa học dữ liệu & Trí tuệ nhân tạo -K1**

|  |
| --- |
| **Số phách**  *(Do hội đồng chấm thi ghi)* |

**Thừa Thiên Huế, ngày….tháng….năm 2023**

# MỤC LỤC

[MỤC LỤC 1](#_Toc135319735)

[DANH MỤC HÌNH ẢNH 2](#_Toc135319736)

[GIỚI THIỆU 3](#_Toc135319737)

[CHƯƠNG I: HỒI QUY TUYẾN TÍNH ĐƠN BIẾN 3](#_Toc135319738)

[CHƯƠNG 2: THUẬT TOÁN GRADIENT DESCENT 3](#_Toc135319739)

[CHƯƠNG 3: ỨNG DỤNG CỦA GRADIENT DESCENT TRONG HỒI QUY TUYẾN TÍNH ĐƠN BIẾN 4](#_Toc135319740)

[CHƯƠNG 4: VÍ DỤ VÀ THỰC NGHIỆM SỐ 5](#_Toc135319741)

[CHƯƠNG 5: CÀI ĐẶT THUẬT TOÁN VÀ TRIỂN KHAI TRÊN PYTHON 6](#_Toc135319742)

[5.1. Cài đặt thuật toán: 6](#_Toc135319743)

[5.2. Triển khai thuật toán tên tập dữ liệu giá bất động sản: 8](#_Toc135319744)

[5.2.1. Mô tả dữ liệu 8](#_Toc135319745)

[5.2.2. Lọc nhiễu dữ liệu 9](#_Toc135319746)

[5.2.3 Giải quyết bài toán bằng python 10](#_Toc135319747)

[TỔNG KẾT 13](#_Toc135319748)

# DANH MỤC HÌNH ẢNH

[Hình 1. Biểu đồ thể hiện tập dữ liệu có đường thẳng hồi quy đi qua góc tọa độ 6](#_Toc135319749)

[Hình 2. Đồ thị hàm mất mát và sự giảm dần của hàm mất mát khi thực hiện Gradient descent với hệ số góc m và b =0 8](#_Toc135319750)

[Hình 3. Phân bố của Diện tích tích bất động sản cùng giá tương ứng của bất động sản 9](#_Toc135319751)

[Hình 4. Sự phân bố của diện tích ảnh hưởng đến giá bất động sản đã qua sàn lọc 10](#_Toc135319752)

[Hình 5. Sự giảm dần của hàm mất mát qua các lần lặp 12](#_Toc135319753)

[Hình 6. Phân bố của tập dữ liệu đã chuẩn hóa và đường thẳng hồi quy 12](#_Toc135319754)

# GIỚI THIỆU

Hồi quy tuyến tính đơn biến và Gradient Descent là hai khái niệm quan trọng trong lĩnh vực Khoa học dữ liệu và Machine Learning. Hồi quy tuyến tính đơn biến là một phương pháp phổ biến trong việc mô hình hóa mối quan hệ giữa biến phụ thuộc và biến độc lập. Trong khi đó, Gradient Descent là một thuật toán tối ưu hóa dựa trên gradient, được áp dụng rộng rãi để tìm giá trị tối ưu của hàm mất mát.

Bài tiểu luận này tập trung vào Gradient Descent cho hồi quy tuyến tính đơn biến và những ứng dụng của nó. Mục tiêu của chúng tôi là khám phá và hiểu rõ cách Gradient Descent làm việc trong bối cảnh của hồi quy tuyến tính đơn biến, cũng như áp dụng nó vào các bài toán thực tế.

# CHƯƠNG I: HỒI QUY TUYẾN TÍNH ĐƠN BIẾN

Trong phần này, chúng ta sẽ đi vào chi tiết về hồi quy tuyến tính đơn biến và các thành phần cơ bản của nó. Hồi quy tuyến tính đơn biến là một phương pháp mô hình hóa mối quan hệ giữa một biến phụ thuộc và một biến độc lập trong một mô hình tuyến tính đơn giản. Điều này cho phép chúng ta dự đoán giá trị của biến phụ thuộc dựa trên giá trị của biến độc lập.

Trong hồi quy tuyến tính đơn biến, chúng ta tìm một đường thẳng tốt nhất (được gọi là đường hồi quy) để biểu diễn mối quan hệ giữa biến phụ thuộc và biến độc lập. Đường hồi quy có phương trình:

y = β₀ + β₁x

Trong đó, y là biến phụ thuộc, x là biến độc lập và β₀, β₁ là các tham số mà chúng ta cần tìm để phù hợp với dữ liệu.

Để tìm các tham số β₀ và β₁, chúng ta sử dụng phương pháp bình phương tối thiểu (Least Squares Method). Phương pháp này giúp chúng ta tìm ra đường hồi quy sao cho tổng bình phương sai số giữa các điểm dữ liệu thực tế và giá trị dự đoán của đường hồi quy là nhỏ nhất.

Một trong những phương pháp tối ưu để tìm các tham số β₀ và β₁ trong hồi quy tuyến tính đơn biến là sử dụng Gradient Descent. Gradient Descent là một thuật toán tối ưu hóa được sử dụng rộng rãi trong Machine Learning để tìm giá trị tối ưu của một hàm mất mát.

# CHƯƠNG 2: THUẬT TOÁN GRADIENT DESCENT

Trong phần này, chúng ta sẽ đi sâu vào thuật toán Gradient Descent và cách nó được áp dụng vào hồi quy tuyến tính đơn biến. Gradient Descent là một thuật toán tối ưu hóa dựa trên gradient, tức là đạo hàm của hàm mất mát, để tìm giá trị tối ưu của hàm mất mát.

Thuật toán Gradient Descent hoạt động bằng cách điều chỉnh các tham số dự đoán theo hướng đối ngược với gradient của hàm mất mát. Quá trình này được lặp lại cho đến khi đạt được một điều kiện dừng, chẳng hạn như đạt đến một số lượng vòng lặp tối đa hoặc sự thay đổi của hàm mất mát nhỏ hơn một ngưỡng cho trước.

Cách thức cập nhật các tham số trong Gradient Descent được xác định bởi tỷ lệ học (learning rate). Learning rate quyết định tốc độ hội tụ của thuật toán. Nếu learning rate quá lớn, thuật toán có thể bỏ qua giá trị tối ưu. Ngược lại, nếu learning rate quá nhỏ, thuật toán có thể mất thời gian lâu để đạt được giá trị tối ưu.

Quá trình cập nhật tham số trong Gradient Descent được thực hiện theo các bước sau:

1. Khởi tạo ngẫu nhiên giá trị ban đầu cho các tham số.
2. Tính toán gradient của hàm mất mát tại các điểm dữ liệu hiện tại.
3. Cập nhật các tham số bằng cách di chuyển ngược dọc theo hướng gradient với tỷ lệ học đã cho.
4. Lặp lại bước 2 và 3 cho đến khi đạt được điều kiện dừng.

Một lợi thế của Gradient Descent là khả năng tìm giá trị tối ưu của hàm mất mát trong không gian lớn và không có giải pháp đóng. Nó cũng có thể được áp dụng cho các mô hình phức tạp hơn với nhiều tham số. Tuy nhiên, cần chú ý rằng Gradient Descent có thể bị mắc phải các vấn đề như xác định learning rate phù hợp và rơi vào các điểm tối thiểu cục bộ.

# CHƯƠNG 3: ỨNG DỤNG CỦA GRADIENT DESCENT TRONG HỒI QUY TUYẾN TÍNH ĐƠN BIẾN

Trong phần này, chúng ta sẽ khám phá các ứng dụng thực tế của Gradient Descent trong hồi quy tuyến tính đơn biến. Gradient Descent có thể được áp dụng để giải quyết nhiều bài toán trong các lĩnh vực khác nhau. **Dưới đây là một số ví dụ phổ biến:**

* Dự đoán giá nhà dựa trên diện tích:Gradient Descent có thể được sử dụng để xây dựng một mô hình hồi quy tuyến tính đơn biến để dự đoán giá trị của một căn nhà dựa trên diện tích của nó. Bằng cách tìm hiểu mối quan hệ giữa diện tích và giá nhà từ dữ liệu huấn luyện, chúng ta có thể sử dụng Gradient Descent để tìm các tham số tối ưu của đường hồi quy.
* Dự đoán doanh số bán hàng dựa trên chi phí quảng cáo: Gradient Descent cũng có thể được áp dụng trong việc dự đoán doanh số bán hàng dựa trên các chi phí quảng cáo. Bằng cách xây dựng một mô hình hồi quy tuyến tính đơn biến, chúng ta có thể tìm ra mối quan hệ giữa chi phí quảng cáo và doanh số bán hàng. Gradient Descent giúp chúng ta tìm ra các tham số tối ưu của mô hình để dự đoán doanh số bán hàng dựa trên chi phí quảng cáo mới.
* Dự đoán tiêu thụ nhiên liệu xe dựa trên tốc độ: Trong lĩnh vực ô tô, Gradient Descent có thể được sử dụng để dự đoán tiêu thụ nhiên liệu của một chiếc xe dựa trên tốc độ lái xe. Bằng cách xây dựng một mô hình hồi quy tuyến tính đơn biến, chúng ta có thể tìm ra mối quan hệ giữa tốc độ và tiêu thụ nhiên liệu. Gradient Descent giúp chúng ta tìm ra các tham số tối ưu của mô hình để dự đoán tiêu thụ nhiên liệu dựa trên tốc độ lái xe mới.

**Lợi ích của việc sử dụng Gradient Descent trong các ứng dụng hồi quy tuyến tính đơn biến là:**

* Tính linh hoạt: Gradient Descent có thể áp dụng cho các mô hình phức tạp với nhiều tham số và không cần phải giải phương trình đóng. Điều này cho phép chúng ta áp dụng phương pháp này cho các bài toán thực tế có dữ liệu lớn và không gian tham số phức tạp.
* Hiệu quả tính toán: Gradient Descent thường có hiệu quả tính toán tốt vì nó chỉ yêu cầu tính toán đạo hàm của hàm mất mát tại mỗi vòng lặp. Điều này đặc biệt hữu ích khi có hàng ngàn hoặc hàng triệu điểm dữ liệu.
* Tìm giá trị tối ưu: Gradient Descent giúp tìm giá trị tối ưu của hàm mất mát, đảm bảo rằng đường hồi quy tìm được là tốt nhất theo tiêu chí bình phương tối thiểu. Điều này cho phép chúng ta có một mô hình hồi quy có khả năng dự đoán tốt.

**Tuy nhiên, việc sử dụng Gradient Descent cũng có một số hạn chế cần xem xét:**

* Xác định learning rate: Việc xác định một learning rate phù hợp là quan trọng để đảm bảo thuật toán hội tụ đúng và hiệu quả. Nếu learning rate quá lớn, thuật toán có thể không hội tụ và dẫn đến phân kỳ. Nếu learning rate quá nhỏ, thuật toán sẽ mất thời gian lâu để đạt được giá trị tối ưu.
* Rơi vào điểm tối thiểu cục bộ: Gradient Descent có thể rơi vào điểm tối thiểu cục bộ nếu hàm mất mát không phải là một hàm lồi. Điều này có thể khiến thuật toán không tìm được giá trị tối ưu toàn cục và chỉ tìm được giá trị tối ưu cục bộ.
* Độ phức tạp của thuật toán: Gradient Descent có thể yêu cầu số lượng lặp lại lớn để đạt được kết quả tối ưu. Điều này đặc biệt đáng lưu ý khi dữ liệu lớn và không gian tham số phức tạp.

# CHƯƠNG 4: VÍ DỤ VÀ THỰC NGHIỆM SỐ

Trong phần này, chúng ta sẽ trình bày một ví dụ cụ thể và thực hiện một số thực nghiệm số để minh họa việc áp dụng Gradient Descent trong hồi quy tuyến tính đơn biến.

Ví dụ: Dự đoán giá nhà dựa trên diện tích

Giả sử chúng ta có một tập dữ liệu với thông tin về diện tích của các căn nhà và giá bán tương ứng. Mục tiêu là xây dựng một mô hình hồi quy tuyến tính đơn biến để dự đoán giá nhà dựa trên diện tích.

Bước 1: Chuẩn bị dữ liệu

* Tiến hành thu thập dữ liệu về diện tích và giá nhà từ các nguồn tin cậy.
* Chuẩn bị dữ liệu bằng cách chia thành hai phần: tập huấn luyện và tập kiểm tra. Tập huấn luyện sẽ được sử dụng để xây dựng mô hình, trong khi tập kiểm tra sẽ được sử dụng để đánh giá hiệu suất của mô hình.

Bước 2: Xác định mô hình hồi quy tuyến tính đơn biến

* Xác định mô hình hồi quy tuyến tính đơn biến: giá nhà = m \* diện tích + b.
* Cần tìm các tham số a và b tối ưu để mô hình phù hợp với dữ liệu.

Bước 3: Áp dụng Gradient Descent

* Khởi tạo ngẫu nhiên các giá trị ban đầu cho m và b.
* Lặp lại các bước sau cho đến khi đạt được điều kiện dừng:
  + Tính toán đạo hàm riêng của hàm mất mát theo m và b.
  + Cập nhật các tham số m và b bằng cách di chuyển ngược dọc theo hướng gradient với tỷ lệ học cho trước.

Bước 4: Đánh giá mô hình

* Sử dụng tập kiểm tra để đánh giá hiệu suất của mô hình dự đoán giá nhà.

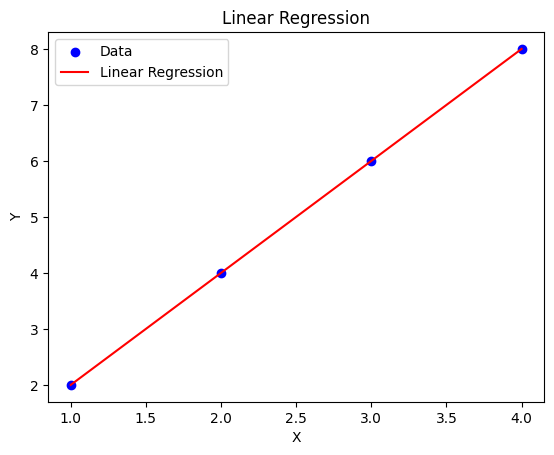
# CHƯƠNG 5: CÀI ĐẶT THUẬT TOÁN VÀ TRIỂN KHAI TRÊN PYTHON

## 5.1. Cài đặt thuật toán:

Với tập dữ liệu mẫu tượng trưng gồm X = [1;2;3;4] và Y = [2;4;6;8]

Ta cần tìm: Hệ số góc (m) và hệ số chặn (b)

Dễ dàng nhận thấy đường thẳng hồi quy đi qua góc tọa độ khi đó b = 0



Hình . Biểu đồ thể hiện tập dữ liệu có đường thẳng hồi quy đi qua góc tọa độ

Đặt b bằng 0 ta giải quyết bài toán bằng python

import numpy as np

# Tập dữ liệu

X = np.array([1, 2, 3, 4])

Y = np.array([2, 4, 6, 8])

m\_s = []

# Số lần lặp và learning rate

epochs = 10

learning\_rate = 0.03

# Khởi tạo hệ số m và b

m = 8

b = 0

# Gradient Descent

for epoch in range(epochs):

    # Tính giá trị dự đoán

    Y\_pred = m \* X #+ b

    # Tính gradient của hàm mất mát

    gradient\_m = -2 \* np.mean(X \* (Y - Y\_pred))

    #gradient\_b = -2 \* np.mean(Y - Y\_pred)

    # Cập nhật các hệ số m và b

    m\_s.append(m)

    m -= learning\_rate \* gradient\_m

    #b -= learning\_rate \* gradient\_b

m\_s.append(m)

# In ra kết quả

print("Hệ số góc (slope) m =", m)

#print("Hệ số chặn (intercept) b =", b)

Kết quả khi chạy đoạn mã: Hệ số góc (slope) m = 2.0151977097271487

Ta suy ra được phương trình của đường thẳng hồi quy là y =2x.

Xét giá trị của hàm mất mát khi b=0: lúc này biểu đồ của hàm mất mát có dạng Parabol khi ta bắt đầu thực hiện gradient Descent tại điểm m = 8 và tốc độ học là 0.03 ta nhận được sự giảm dần của giá trị hàm mất mát khi m tiến gần về 2.

A graph with a red line and a blue line

Description automatically generated with low confidence

Hình . Đồ thị hàm mất mát và sự giảm dần của hàm mất mát khi thực hiện Gradient descent với hệ số góc m và b =0

## 5.2. Triển khai thuật toán tên tập dữ liệu giá bất động sản:

**Nguồn dữ liệu:** [**https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques**](https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques)

### 5.2.1. Mô tả dữ liệu

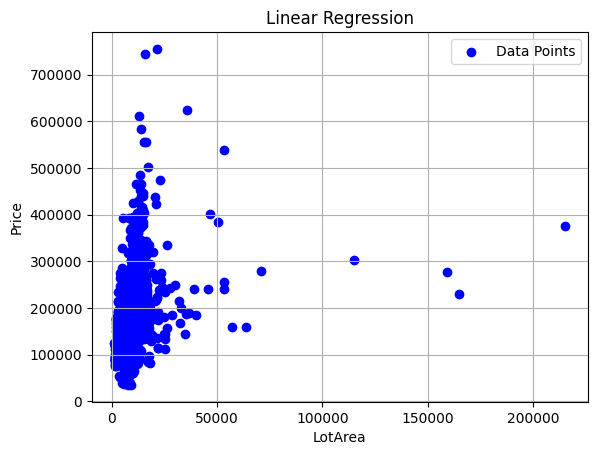
Tập dữ liệu liên quan đến những thuộc tính của các bất động sản. Phân tích các cột:

* Id: Mã số nhận dạng cho mỗi bất động sản.
* MSSubClass: Lớp xây dựng của bất động sản.
* MSZoning: Phân loại khu vực (zoning) tổng quát của bất động sản.
* LotFrontage: Chiều dài đường phố kết nối với bất động sản (theo đơn vị feet).
* LotArea: Diện tích đất (theo đơn vị feet vuông).
* Street: Loại đường truy cập vào bất động sản (Pave có nghĩa là đường bê tông).
* Alley: Loại đường hẻm truy cập vào bất động sản (NA có nghĩa là không có đường hẻm truy cập).
* LotShape: Hình dạng tổng quát của lô đất.
* LandContour: Địa hình đất.
* Utilities: Tiện ích công cộng (AllPub có nghĩa là tất cả các tiện ích công cộng đều có sẵn).
* LotConfig: Cấu trúc lô đất.
* LandSlope: Độ dốc đất.
* Neighborhood: Khu vực lân cận.
* …

Và SalePrice: Giá của Bất động sản tương ứng

### 5.2.2. Lọc nhiễu dữ liệu

Ở đây để phục vụ cho bài toán đơn biến ta chỉ xét một trường dữ liệu duy nhất đó chính là diện tích đất của bất động sản sẽ ảnh hướng đến giá của bất động sản như thế nào. Dựa trên Cột LotArea(Diện tích bất động sản) và SalePrice (Giá của bất động sản)



Hình . Phân bố của Diện tích tích bất động sản cùng giá tương ứng của bất động sản

Nhận thấy sự phân bố của giá bất động sản dựa trên diện tích đất có các điểm dữ liệu bị nhiễu, đặc biệt ở bất động sản có diện tích cao. Tiến hành lọc lấy những điểm dữ liệu có diện tích dưới 15000 feet vuông.

A picture containing text, screenshot, diagram, line

Description automatically generated

Hình . Sự phân bố của diện tích ảnh hưởng đến giá bất động sản đã qua sàn lọc

Sau khi lọc ra những bất động sản có diện tích dưới 15000 feet vuông nhận thấy bắt đầu có sự tương quan giữa Diện tích đất và Giá bất động sản.

### 5.2.3 Giải quyết bài toán bằng python

import numpy as np

import matplotlib.pyplot as plt

# Dữ liệu LotArea và SalePrice (ví dụ)

LotArea = np.array(df\_2["LotArea"])

SalePrice = np.array(df\_2["SalePrice"])

# Chuẩn hóa dữ liệu (phạm vi từ 0 đến 1)

LotArea\_normalized = (LotArea - LotArea.min()) / (LotArea.max() - LotArea.min())

SalePrice\_normalized = (SalePrice - SalePrice.min()) / (SalePrice.max() - SalePrice.min())

# Số lượng mẫu và learning rate

m = len(LotArea\_normalized)

learning\_rate = 0.01

# Thêm cột x0 = 1 vào ma trận LotArea\_normalized để tính intercept

X = np.c\_[np.ones(m), LotArea\_normalized]

# Khởi tạo ngẫu nhiên các tham số m và b

theta = np.random.rand(2)

# Hàm giả thiết

def hypothesis(X, theta):

    return X.dot(theta)

# Hàm mất mát (mean squared error)

def cost\_function(X, y, theta):

    m = len(y)

    predictions = hypothesis(X, theta)

    cost = np.sum((predictions - y) \*\* 2) / (2 \* m)

    return cost

# Gradient descent

def gradient\_descent(X, y, theta, learning\_rate, num\_iterations):

    m = len(y)

    cost\_history = []

    for i in range(num\_iterations):

        predictions = hypothesis(X, theta)

        error = predictions - y

        gradient = X.T.dot(error) / m

        theta -= learning\_rate \* gradient

        cost = cost\_function(X, y, theta)

        cost\_history.append(cost)

    return theta, cost\_history

# Thực hiện gradient descent

num\_iterations = 1000

theta\_final, cost\_history = gradient\_descent(X, SalePrice\_normalized, theta, learning\_rate, num\_iterations)

# Trực quan hóa kết quả

plt.plot(range(1, num\_iterations+1), cost\_history)

plt.xlabel('Iterations')

plt.ylabel('Cost')

plt.title('Gradient Descent - Cost History')

plt.show()

# Vẽ đường thẳng hồi quy trên dữ liệu chuẩn hóa

plt.scatter(LotArea\_normalized, SalePrice\_normalized, color='blue', label='Data')

plt.plot(LotArea\_normalized, hypothesis(X, theta\_final), color='red', label='Linear Regression')

plt.xlabel('LotArea (Normalized)')

plt.ylabel('SalePrice (Normalized)')

plt.title('Linear Regression - Gradient Descent')

plt.legend()

plt.show()

A picture containing text, screenshot, diagram, line

Description automatically generated

Hình . Sự giảm dần của hàm mất mát qua các lần lặp

A graph with blue dots and a red line

Description automatically generated with low confidence

Hình . Phân bố của tập dữ liệu đã chuẩn hóa và đường thẳng hồi quy

Ta tìm được phương trình hồi quy tương ứng:

SalePrice = 0.06467567 + 0.31580393 \* LotArea

# TỔNG KẾT

Trong bài tiểu luận này, chúng ta đã tìm hiểu về Gradient Descent và ứng dụng của nó trong hồi quy tuyến tính đơn biến. Gradient Descent là một thuật toán tối ưu hóa được sử dụng để tìm giá trị tối ưu của một hàm mất mát. Trong hồi quy tuyến tính đơn biến, Gradient Descent có thể được sử dụng để tìm các tham số tối ưu (a và b) cho mô hình hồi quy tuyến tính đơn biến (giá nhà = a \* diện tích + b).

Chúng ta đã xem xét các bước để áp dụng Gradient Descent trong hồi quy tuyến tính đơn biến, bao gồm chuẩn bị dữ liệu, xác định mô hình, áp dụng thuật toán Gradient Descent và đánh giá mô hình. Đồng thời, chúng ta cũng đã trình bày một ví dụ cụ thể và thực hiện một thực nghiệm số để minh họa việc sử dụng Gradient Descent trong dự đoán giá nhà dựa trên diện tích.

Cuối cùng, chúng ta cũng đã phân tích lợi ích và hạn chế của việc sử dụng Gradient Descent trong hồi quy tuyến tính đơn biến. Lợi ích của Gradient Descent bao gồm khả năng tối ưu hóa mô hình, khả năng xử lý dữ liệu lớn và tính linh hoạt. Tuy nhiên, cũng cần lưu ý đến một số hạn chế như tốc độ hội tụ chậm và khả năng rơi vào các điểm cực tiểu cục bộ.

Qua bài tiểu luận này, chúng ta hy vọng đã có cái nhìn tổng quan về Gradient Descent cho hồi quy tuyến tính đơn biến và ứng dụng của nó. Đồng thời, chúng ta cũng nhận thấy tầm quan trọng của việc thực nghiệm và áp dụng thuật toán để hiểu rõ hơn về công cụ này trong việc giải quyết các bài toán thực tế.

**ĐẠI HỌC HUẾ**

**KHOA KỸ THUẬT VÀ CÔNG NGHỆ**

&

**PHIẾU ĐÁNH GIÁ ĐỒ ÁN/TIỂU LUẬN/BÀI TẬP LỚN Học kỳ II, năm học 2022 - 2023**

|  |  |
| --- | --- |
| **Cán bộ chấm thi 1** | **Cán bộ chấm thi 2** |
| **Nhận xét:**  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  **Điểm đánh giá của CBCT1:**  Bằng số: .........................................  Bằng chữ: ....................................... | **Nhận xét:**  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  ..............................................................  **Điểm đánh giá của CBCT2:**  Bằng số: .........................................  Bằng chữ: ....................................... |

Điểm kết luận:

Bằng số:

Bằng chữ:

*Thừa Thiên Huế, ngày…tháng…năm 2023*

**Cán bộ chấm thi 1 Cán bộ chấm thi 2**

*(Ký và ghi rõ họ và tên) (Ký và ghi rõ họ và tên)*