**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: DỮ LIỆU LỚN**

**ĐỀ TÀI: PHÂN TÍCH VÀ DỰ ĐOÁN GIÁ BÁN XE CŨ**

**Giảng viên hướng dẫn: TS.Trần Qúy Nam**

**ThS.Lê Thị Thùy Trang**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Lớp |
| 1 | 157102076 | Vũ Minh Đức | 06/09/2003 | CNTT 16-02 |
| 2 | 1671020326 | Nguyễn Viết Trọng | 24/09/2004 | CNTT 16-02 |
| 3 | 1671020043 | Nguyễn Thị Linh Chi | 12/12/2003 | CNTT16-02 |
| 4 | 1571020147 | Dương Thị Huơng Lan | 29/01/2003 | CNTT16-02 |

### 

**Hà Nội, năm 2025**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: DỮ LIỆU LỚN**

**ĐỀ TÀI: PHÂN TÍCH VÀ DỰ ĐOÁN GIÁ BÁN XE CŨ**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Điểm | |
| Bằng Số | Bằng Chữ |
| 1 | 1571020076 | Vũ Minh Đức | 06/09/2003 |  |  |
| 2 | 1671020326 | Nguyễn Viết Trọng | 24/09/2004 |  |  |
| 3 | 1671020043 | Nguyễn Thị Linh Chi | 12/12/2003 |  |  |
| 4 | 1571020147 | Dương Thị Huơng Lan | 29/01/2003 |  |  |

### 

### CÁN BỘ CHẤM THI 1 CÁN BỘ CHẤM THI 2

**Hà Nội, năm 2025**

**LỜI NÓI ĐẦU**

Trong thời đại công nghệ số phát triển mạnh mẽ như hiện nay, dữ liệu đã trở thành một trong những tài nguyên quý giá nhất, đóng vai trò then chốt trong việc định hướng chiến lược và quyết định của các tổ chức, doanh nghiệp. Môn học "Dữ liệu lớn" (Big Data) không chỉ cung cấp những kiến thức nền tảng về cách thu thập, xử lý và phân tích khối lượng dữ liệu khổng lồ mà còn mở ra cánh cửa để khám phá những giá trị tiềm ẩn từ dữ liệu, giúp giải quyết các bài toán phức tạp trong nhiều lĩnh vực khác nhau.

Báo cáo này được thực hiện nhằm tổng hợp và trình bày những kiến thức, kỹ năng đã tiếp thu trong quá trình học tập môn "Dữ liệu lớn". Đồng thời, báo cáo cũng đề cập đến những ứng dụng thực tiễn của Big Data trong đời sống và kinh doanh, cũng như những thách thức và xu hướng phát triển trong tương lai. Qua đó, chúng tôi hy vọng có thể mang đến cái nhìn tổng quan và sâu sắc hơn về tầm quan trọng của dữ liệu lớn trong bối cảnh hiện nay.

Chúng em xin chân thành cảm ơn sự hướng dẫn tận tình của quý thầy cô, cũng như sự hỗ trợ từ các bạn đồng học, giúp chúng em hoàn thành báo cáo này. Mong rằng những nội dung trình bày trong báo cáo sẽ đóng góp một phần nhỏ vào sự hiểu biết chung về lĩnh vực đầy tiềm năng này.

Trân trọng!

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn chân thành đến cô Lê Thị Thùy Trang và thầy Trần Qúy Nam, những người đã tận tình giảng dạy và hướng dẫn chúng em trong suốt quá trình học tập môn Dữ liệu lớn. Nhờ những bài giảng sâu sắc, những ví dụ thực tế và sự hướng dẫn tận tình của cô, em đã có được nền tảng kiến thức vững chắc về cơ sở dữ liệu, từ lý thuyết đến ứng dụng thực tiễn.

Bên cạnh đó, em cũng xin cảm ơn các bạn bè đã hỗ trợ và cùng nhau trao đổi, thảo luận trong quá trình học tập, giúp em hiểu sâu hơn về các mô hình dữ liệu, cách thiết kế hệ thống và tối ưu hóa truy vấn.

Môn học này không chỉ giúp em hiểu rõ hơn về cách xây dựng, quản lý và tìm kiếm dữ liệu mà còn mang lại những kiến thức quý báu để ứng dụng vào các dự án thực tế trong tương lai.

Một lần nữa, em xin chân thành cảm ơn thầy cô và tất cả mọi người đã giúp đỡ em trong quá trình học tập và thực hiện bài tập lớn này!

MỤC LỤC

[**CHƯƠNG I: TỔNG QUAN LÝ THUYẾT** 8](#_Toc193051098)

[**1.1 Tổng quan về dữ liệu lớn** 8](#_Toc193051099)

[**1.2** **Khái niệm, ý nghĩa và ứng dụng của Hadoop** 8](#_Toc193051100)

[**1.3 Kiến trúc và quản trị Hadoop** 9](#_Toc193051101)

[**1.4 Các thành phần Hadoop** 9](#_Toc193051102)

[**1.5 Mô hình lập trình Hadoop Spark** 10](#_Toc193051103)

[**1.6 Phân tích đặc điểm lập trình Python, các thư viện Python trong trực quan hóa dữ liệu, phân lớp, phân cụm** 11](#_Toc193051104)

[**1.7 Phân tích đặc điểm lập trình R, các thư viện lập trình R trong phân tích dữ liệu** 14](#_Toc193051105)

[**CHƯƠNG II: XÂY DỰNG MÔ HÌNH DỰ ĐOÁN** 16](#_Toc193051106)

[**2.1. Thu thập và tiền xử lý dữ liệu** 16](#_Toc193051107)

[**2.2. Xây dựng mô hình** 16](#_Toc193051108)

[**2.3. Kết quả và đánh giá mô hình** 16](#_Toc193051109)

[**2.4. So sánh giá dự đoán và giá thực tế** 17](#_Toc193051110)

[**2.4. Trực quan hóa dữ liệu và kết quả dự đoán** 17](#_Toc193051111)

[**CHƯƠNG 3: KẾT QUẢ VÀ ĐÁNH GIÁ** 23](#_Toc193051112)

[**3.1. Hiệu suất mô hình** 23](#_Toc193051113)

[**3.2. Đánh giá khả năng tổng quát hóa** 23](#_Toc193051114)

[**3.3. Phân tích và so sánh kết quả** 23](#_Toc193051115)

[**3.4. Các yếu tố ảnh hưởng đến kết quả dự đoán** 23](#_Toc193051116)

[**3.5. Tổng kết và đề xuất** 24](#_Toc193051117)

[**KẾT LUẬN** 25](#_Toc193051118)

**BẢNG CÁC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| **1** | **CSDL** | **Cơ sở dữ liệu** |
| **2** | **QLKS** | **Quản lý khách sạn** |

# **CHƯƠNG I: TỔNG QUAN LÝ THUYẾT**

## **1.1 Tổng quan về dữ liệu lớn**

Dữ liệu lớn (Big data) là một thuật ngữ chỉ bộ dữ liệu lớn hoặc phức tạp mà các phương pháp truyền thống không đủ các ứng dụng để xử lý dữ liệu này.

Dữ liệu lớn là những nguồn thông tin có đặc điểm chung khối lượng lớn, tốc độ nhanh và dữ liệu định dạng dưới nhiều hình thức khác nhau, do đó muốn khai thác được đòi hỏi phải có hình thức xử lý mới để đưa ra quyết định, khám phá và tối ưu hóa quy trình.

* 1. **Khái niệm, ý nghĩa và ứng dụng của Hadoop**

là một dạng framework, cụ thể là Apache. Apache Hadoop là một mã nguồn mở cho phép sử dụng các distributed processing (ứng dụng phân tán) để quản lý và lưu trữ những tệp dữ liệu lớn. Hadoop áp dụng mô hình MapReduce trong hoạt động xử lý Big Data.

Vậy MapReduce là gì? MapReduce vốn là một nền tảng được Google tạo ra để quản lý dữ liệu của họ. Nhiệm vụ của MapReduce là tiếp nhận một khối lượng dữ liệu lớn. Sau đó sẽ tiến hành tách các dữ liệu này ra thành những phần nhỏ theo một tiêu chuẩn nào đó. Từ đó sẽ sắp xếp, trích xuất các tệp dữ liệu con mới phù hợp với yêu cầu của người dùng. Đây cũng là cách mà thanh tìm kiếm của Google hoạt động trong khi chúng ta sử dụng hằng ngày.

Còn bản thân Hadoop cũng là một dạng công cụ mẫu giúp phân tán dữ liệu theo mô hình như vậy. Cho nên MapReduce được sử dụng như một nền tảng lý tưởng của Hadoop. Về cơ bản, Hadoop sẽ giúp người dùng tổng hợp và xử lý một lượng thông tin lớn trong thời gian ngắn bằng MapReduce.

Còn với chức năng lưu trữ, Hadoop sẽ dùng HDFS. HDFS là gì? Nó được biết đến như một kho thông tin có độ truy cập nhạy và chi phí thấp.

Hadoop được phát triển nên từ ngôn ngữ Java. Tuy nhiên nó vẫn hỗ trợ một số ngôn ngữ lập trình khác như C++, Python hay Pearl nhờ cơ chế streaming.

## **1.3 Kiến trúc và quản trị Hadoop**

Vì sử dụng cùng lúc MapReduce và HDFS nên Hadoop sẽ có cấu trúc của cả 2 loại này. Hadoop kế thừa cấu trúc node từ HDFS. Cụ thể, một cụm Hadoop sẽ bao gồm 1 master node (node chủ) và rất nhiều worker/slave node (node nhân viên). Một cụm cũng bao gồm 2 phần là MapReduce layer và HDFS layer. Master node bao gồm JobTracker, TaskTracker, NameNode, và DataNode. Còn Worker/Slave node bao gồm DataNode và TaskTracker. Trong một số trường hợp, Worker/Slave node được dùng để làm dữ liệu hoặc tính toán.

**Quản trị Hadoop**

Vì sử dụng cùng lúc MapReduce và HDFS nên Hadoop sẽ có cấu trúc của cả 2 loại này. Hadoop kế thừa cấu trúc node từ HDFS. Cụ thể, một cụm Hadoop sẽ bao gồm 1 master node (node chủ) và rất nhiều worker/slave node (node nhân viên). Một cụm cũng bao gồm 2 phần là MapReduce layer và HDFS layer. Master node bao gồm JobTracker, TaskTracker, NameNode, và DataNode. Còn Worker/Slave node bao gồm DataNode và TaskTracker. Trong một số trường hợp, Worker/Slave node được dùng để làm dữ liệu hoặc tính toán.

Hadoop Apache bao gồm 4 module khác nhau. Sau đây sẽ là giới thiệu chi tiết về từng loại.

## **1.4 Các thành phần Hadoop**

* **Hadoop Common**

Hadoop Common được dùng như một thư viện lưu trữ các tiện ích của Java. Tại đây có những tính năng cần thiết để các modules khác sử dụng. Những thư viện này mang đến hệ thống file và lớp OS trừu tượng. Song song với đó, nó cũng lưu trữ các mã lệnh của Java để thực hiện quá trình khởi động Hadoop.

* **Hadoop YARN – Hadoop là gì?**

Phần này được dùng như một framework. Nó hỗ trợ hoạt động quản lý thư viện tài nguyên của các cluster và thực hiện chạy phân tích tiến trình.

* **Hadoop Distributed File System (HDFS)**

Một trong những vấn đề lớn nhất của các hệ thống phân tích Big Data là quá tải. Không phải hệ thống nào cũng đủ khỏe để có thể tiếp nhận một lượng thông tin khổng lồ như vậy. Chính vì thế, nhiệm vụ của Hadoop Distributed File System là phân tán cung cấp truy cập thông lượng cao giúp cho ứng dụng chủ. Cụ thể, khi HDFS nhận được một tệp tin, nó sẽ tự động chia file đó ra thành nhiều phần nhỏ. Các mảnh nhỏ này được nhân lên nhiều lần và chia ra lưu trữ tại các máy chủ khác nhau để phân tán sức nặng mà dữ liệu tạo nên.

Như đã nói ở trên, HDFS sử dụng cấu trúc master node và worker/slave node. Trong khi master node quản lý các file metadata thì worker/slave node chịu trách nhiệm lưu trữ dữ liệu. Chính vì thế nên worker/slave node cũng được gọi là data node. Một Data node sẽ chứa nhiều khối được phân nhỏ của tệp tin lớn ban đầu. Dựa theo chỉ thị từ Master node, các Data node này sẽ trực tiếp điều hành hoạt động thêm, bớt những khối nhỏ của tệp tin.

* **Hadoop MapReduce**

Module này hoạt động dựa trên YARN trong việc xử lý các tệp dữ liệu lớn. Hadoop MapReduce cho phép phân tán dữ liệu từ một máy chủ sang nhiều máy con. Mỗi máy con này sẽ nhận một phần dữ liệu khác nhau và tiến hành xử lý cùng lúc. Sau đó chúng sẽ báo lại kết quả lên máy chủ. Máy chủ tổng hợp thông tin lại rồi trích xuất theo như yêu cầu của người dùng.

## **1.5 Mô hình lập trình Hadoop Spark**

Trước khi Spark ra đời, Hadoop đang là một công cụ mạnh mẽ và phổ biến, tuy nhiên Hadoop có những hạn chế nhất định và Spark ra đời để cải thiện các hạn chế đó.

* Spark ra đời với mục tiêu:
* Tối ưu hóa để xử lý lặp đi lặp lại đối với các bài toán học máy
* Phân tích dữ liệu tương tác trong khi vẫn giữ được khả năng mở rộng và khả năng chịu lỗi của Hadoop MapReduce
* Xử lý dữ liệu tinh vi
* Xử lý dữ liệu streaming với độ trễ thấp
* Hadoop và Spark đồng hành với nhau, kết hợp cả hai, Spark có thể tận dụng các tính năng mà nó còn thiếu, chẳng hạn như hệ thống tệp. Spark cũng có thể tận dụng các lợi ích về bảo mật và quản lý tài nguyên của Hadoop. Với YARN, việc phân cụm Spark và quản lý dữ liệu dễ dàng hơn nhiều.

## **1.6 Phân tích đặc điểm lập trình Python, các thư viện Python trong trực quan hóa dữ liệu, phân lớp, phân cụm**

* Python đã trở thành một trong những ngôn ngữ yêu thích của các nhà khoa học dữ liệu trong một khoảng thời gian. Nó đã trở nên phổ biến rộng rãi trong lĩnh vực khoa học dữ liệu do có nhiều thư viện phong phú được sử dụng cho toàn bộ quá trình ra quyết định theo hướng dữ liệu.

Các thư viện này được sử dụng ở các bước khoa học dữ liệu khác nhau như tải dữ liệu, xử lý dữ liệu, trực quan hóa dữ liệu và mô hình hóa. Trong Python, các thư viện này được sử dụng để tự động hóa các tác vụ, dự đoán kết quả.

* Trong bài viết này, tôi sẽ giới thiệu một số thư viện Python hữu ích cho các nhà khoa học dữ liệu, kỹ sư dữ liệu và kỹ sư học máy. Chúng ta hãy cùng bắt đầu nào!
* **Pandas**

Pandas là một gói Python mã nguồn mở cung cấp cấu trúc dữ liệu linh hoạt và hiệu suất cao được sử dụng để làm việc với cả hai loại dữ liệu có cấu trúc và thời gian.

Gói này là một công cụ rất hữu dụng và là một công cụ hoàn hảo để thực hiện quá trình phân tích dữ liệu và xử lý dữ liệu. Nó là một công cụ mạnh mẽ được sử dụng để thao tác dữ liệu, làm sạch dữ liệu và tổng hợp dữ liệu.

Pandas hoạt động tốt khi làm việc với dữ liệu dạng bảng, thời gian và dữ liệu thống kê. Nó có hai loại cấu trúc dữ liệu chính, Series (1 chiều) và DataFrame (2 chiều).

* Công dụng của Pandas là gì?

Xử lý dữ liệu thiếu có trong tập dữ liệu

Cập nhật, chèn và xóa cột khỏi tập dữ liệu

Hợp nhất, nối và định hình lại tập dữ liệu

Nhóm tập hợp linh hoạt theo chức năng để tổng hợp và chuyển đổi dữ liệu

* **NumPy**

NumPy là một trong những gói cơ bản nhất trong Python. Đó là một gói xử lý mảng một cách tổng quát. Nó cung cấp hiệu suất cao khi làm việc với các đối tượng mảng đa chiều. NumPy là viết tắt của NUMerical PYthon.

Nó có một cộng đồng tham gia tích cực gồm hơn 700 người đang đóng góp trên GitHub.

Thư viện NumPy được sử dụng để xử lý các mảng lưu trữ, các giá trị của kiểu dữ liệu tương tự. Nó cũng thực hiện các phép toán trên các mảng và làm cho việc vector hóa dễ dàng hơn rất nhiều.

Công dụng của Numpy là gì?

Tạo và xử lý mảng N chiều

Các thao tác với mảng nâng cao như chia thành các phần, mảng ngăn xếp.

Định hình lại kích thước của các mảng

Cắt mảng cơ bản và lập chỉ số nâng cao

* **Matplotlib**

Matplotlib là công cụ trực quan hóa dữ liệu mạnh mẽ của Python. Nó là một thư viện đa nền tảng để tạo các ô 2 chiều với sự trợ giúp của các mảng. Thư viện này cũng được sử dụng để tạo các hình ảnh trực quan trong Python.

Matplotlib là một trong những thư viện vẽ sơ đồ hữu ích được sử dụng trong các dự án khoa học dữ liệu. Nó trực quan hóa dữ liệu như sơ đồ 2 chiều, biểu đồ và đồ thị.

Nó cũng cung cấp một API hướng đối tượng để nhúng các hình vẽ vào ứng dụng.

Công dụng của Matplotlib là gì?

Trực quan hóa việc phân phối dữ liệu để có được thông tin chi tiết

Vẽ biểu đồ như biểu đồ, biểu đồ phân tán, biểu đồ đường,...

Trực quan hóa các điểm dữ liệu trên mặt phẳng 2 chiều

* **SciPy**

Thư viện SciPy (Scientific Python) là một thư viện Python mã nguồn mở được sử dụng rộng rãi cho các phép tính toán cấp cao trong lĩnh vực toán học. Thư viện khoa học này bao gồm các mô-đun cho đại số tuyến tính, thống kê và tối ưu hóa.

Tài liệu phong phú của thư viện SciPy giúp làm việc với nó rất dễ dàng. Nó có một cộng đồng tích cực gồm hơn 600 người đóng góp trên Github.

* Công dụng của SciPy là gì?
* Đại số tuyến tính và các phép toán
* Tối ưu hóa thuật toán
* Giải các phương trình vi phân khác nhau
* Xử lý hình ảnh
* **Scikit Learn**

Scikit-learning là một thư viện Python dành cho học máy được xây dựng dựa trên SciPy. Năm 2007, dự án được David Cournapeau bắt đầu là một dự án Google Summer of Code.

Các nhà khoa học dữ liệu sử dụng thư viện này cho các tác vụ học máy như hồi quy, phân loại, phân cụm, lựa chọn mô hình và giảm kích thước.

Hầu hết thư viện Scikit-learning được viết bằng ngôn ngữ Python và nó sử dụng NumPy để có hiệu suất cao trong các phép tính toán mảng và đại số tuyến tính.

Công dụng của Scikit-Learn là gì?

Hồi quy, chẳng hạn như hồi quy tuyến tính và hồi quy Logistic

Phân loại, chẳng hạn như K-Nearest Neighbors

Phân cụm, chẳng hạn như K-Means

Lựa chọn mô hình

Giảm kích thước

* **PyCaret**

PyCaret, là một thư viện học máy mã nguồn mở bằng Python, cho phép bạn tự động hóa một số bước chính để đánh giá và so sánh các thuật toán học máy. Nó là một công cụ quản lý mô hình và học máy.

Thư viện này tự động hóa quy trình công việc học máy và thay thế hàng trăm dòng mã chỉ với vài bước. PyCaret giúp bạn làm việc hiệu quả hơn. Bạn sẽ dành ít thời gian hơn cho việc viết mã, và có thể tập trung vào các vấn đề hay bài toán cần giải quyết.

* Công dụng của PyCaret là gì?
* Tự động xác định các thao tác chuyển đổi dữ liệu được thực hiện
* Nó tự động điều chỉnh các siêu tham số của mô hình
* Nó tự động đánh giá và so sánh các mô hình học máy
* Thực hiện các thử nghiệm học máy từ đầu đến cuối trong vài dòng mã
* **TensorFlow**

Được phát triển bởi nhóm Google Brain, Tensorflow là một trong những Framework bằng Python phổ biến nhất cho học máy và học sâu. Nó được phát triển với mục đích tiến hành học máy và nghiên cứu mạng nơ-ron.

TensorFlow là một thư viện Python cho phép các nhà khoa học dữ liệu phát triển các mạng nơ-ron trong học sâu. Nó trở nên phổ biến với tư cách là một thư viện phổ được dùng rất nhiều cho học sâu trong một thời gian ngắn. Nó đã được viết bằng ngôn ngữ C ++, Python và CUDA.

TensorFlow cho phép các nhà khoa học dữ liệu phát triển các mô hình học sâu, đẩy mạnh công nghệ tiên tiến trong học máy và nhanh chóng xây dựng và triển khai các ứng dụng hỗ trợ học máy.

* Công dụng của TensorFlow là gì?
* Phân loại hình ảnh
* Nhận dạng giọng nói và hình ảnh
* Phát hiện video, phát hiện chuyển động
* **Keras**

Keras là một API học sâu cấp cao để xây dựng và đào tạo mạng nơ-ron. Nó chạy trên nền tảng học máy TensorFlow và sử dụng các gói học sâu khác như TensorFlow và Theano làm chương trình phụ trợ của nó.

Keras rất dễ sử dụng, trong khi TensorFlow phức tạp hơn nhiều. Tensorflow cung cấp cả API mức cao và mức thấp, và Keras chỉ cung cấp các API mức cao đơn giản và nhất quán.

Do đó, Keras là một lựa chọn tốt cho bạn nếu bạn không muốn sự phức tạp của TensorFlow.

* Công dụng của Keras là gì?
* Tính toán hàm mất mát
* Tạo mô hình phân loại hình ảnh
* Xác định độ chính xác của mô hình
* Xử lý hình ảnh

**1.7 Phân tích đặc điểm lập trình R, các thư viện lập trình R trong phân tích dữ liệu**

R là một ngôn ngữ lập trình và môi trường phần mềm được sử dụng rộng rãi trong phân tích dữ liệu và thống kê. Nó cung cấp một loạt các thư viện và gói phần mềm mạnh mẽ để xử lý, phân tích và trực quan hóa dữ liệu. Dưới đây là một số đặc điểm quan trọng của R và các thư viện phổ biến trong phân tích dữ liệu.

* **Đặc điểm của R:**

R là một ngôn ngữ lập trình miễn phí và mã nguồn mở.

Nó cung cấp một môi trường tính toán phân tán và mạnh mẽ cho phân tích dữ liệu.

R có cú pháp dễ học và đọc, giúp người dùng dễ dàng tạo, kiểm tra và sửa lỗi mã.

Nó hỗ trợ rất nhiều kiểu dữ liệu và toán tử, bao gồm cả vectơ, ma trận, danh sách, chuỗi và số.

R cung cấp một số lượng lớn các gói phần mềm và thư viện, cho phép người dùng mở rộng chức năng của nó và xử lý các tác vụ phức tạp.

* **Các thư viện phổ biến trong R cho phân tích dữ liệu:**

ggplot2: Thư viện này cung cấp các công cụ mạnh mẽ cho việc tạo ra đồ thị và biểu đồ trực quan. Nó dựa trên ý tưởng về "Grammar of Graphics" và cho phép người dùng dễ dàng tạo ra các biểu đồ phức tạp và tùy chỉnh chúng.

dplyr: Thư viện này cung cấp các công cụ cho việc xử lý và biến đổi dữ liệu. Nó cung cấp các hàm mạnh mẽ để lọc, sắp xếp, nhóm và tóm tắt dữ liệu một cách dễ dàng và hiệu quả.

tidyr: Thư viện này giúp người dùng làm việc với dữ liệu dạng "dài" và "rộng". Nó cung cấp các hàm để chuyển đổi dữ liệu giữa các định dạng này và làm cho dữ liệu trở nên dễ dàng để xử lý và phân tích.

caret: Thư viện này cung cấp các công cụ cho việc xây dựng và đánh giá mô hình dựa trên học máy. Nó cung cấp một giao diện thống nhất để làm việc với nhiều thuật toán máy học khác nhau và hỗ trợ việc chọn mô hình và điều chỉnh siêu tham số.

magrittr: Thư viện này cung cấp cú pháp cho phép người dùng kết hợp các hàm trong R một cách dễ dàng và đọc được. Nó cho phép việc viết mã theo kiểu "pipe" để xử lý dữ liệu một cách trực quan và tuần tự.

# **CHƯƠNG II: XÂY DỰNG MÔ HÌNH DỰ ĐOÁN**

## **2.1. Thu thập và tiền xử lý dữ liệu**

Dữ liệu được sử dụng trong đề tài là bộ dữ liệu **CAR DETAILS FROM CAR DEKHO**, bao **name**: Tên xe

**year**: Năm sản xuất xe

**selling\_price**: Giá bán xe (INR - đồng Rupee Ấn Độ)

**km\_driven**: Số km xe đã đi

**fuel**: Loại nhiên liệu (Diesel, Petrol, CNG, LPG)

**seller\_type**: Loại người bán

**transmission**: Loại hộp số (Automatic, Manual)

**owner**: Số đời chủ sở hữu trước

Dữ liệu gồm 4340 dòng và 8 cột. Đã loại bỏ cột name do không mang giá trị định lượng phục vụ dự đoán.

## **2.2. Xây dựng mô hình**

Mô hình được lựa chọn để dự đoán giá bán là **Random Forest Regression** với tham số:

n\_estimators = 100

random\_state = 42

## **2.3. Kết quả và đánh giá mô hình**

Sau khi huấn luyện và dự đoán trên tập dữ liệu kiểm tra (chiếm 20%), mô hình đạt kết quả như sau:

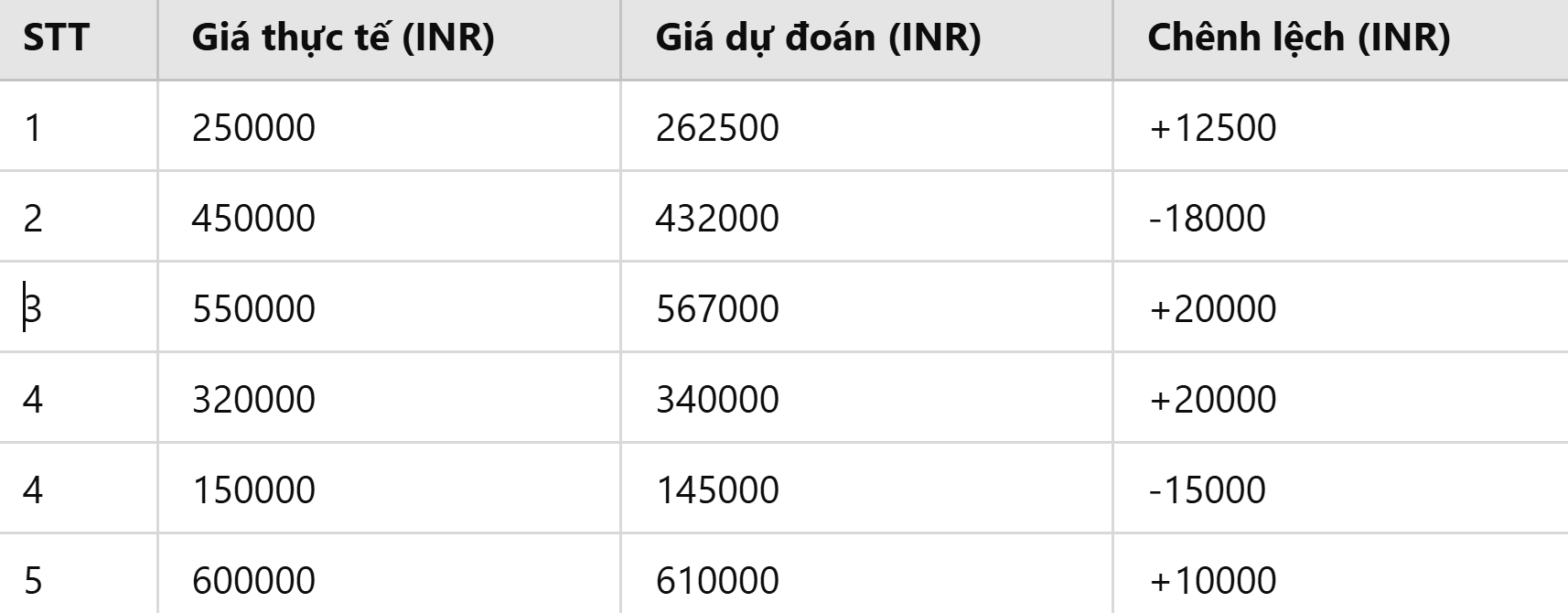
**Mean Squared Error (MSE)**: 227,478,606.71 INR²

**Hệ số xác định (R² Score)**: 0.94 (94%)

Mô hình Random Forest Regression đạt được R² = 0.94, cho thấy khả năng dự đoán rất tốt, chỉ sai lệch nhỏ so với giá thực tế.

## **2.4. So sánh giá dự đoán và giá thực tế**

Bảng dưới đây trình bày một số mẫu kết quả dự đoán:

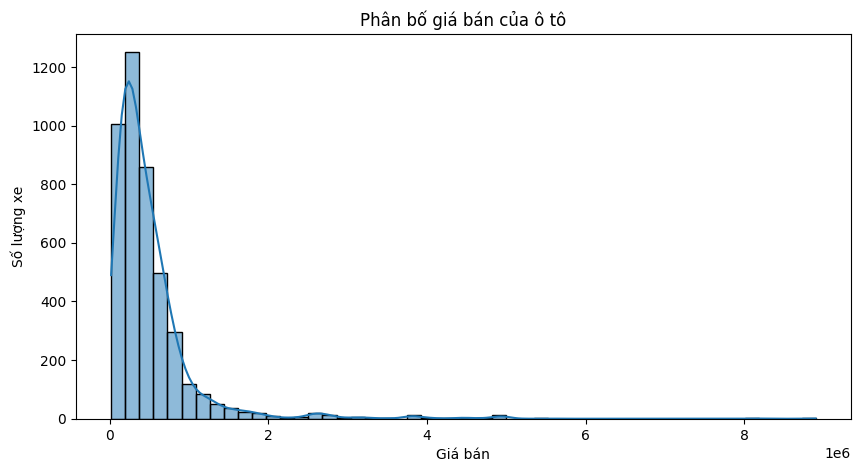


Kết quả bảng trên thể hiện độ chính xác cao của mô hình khi dự đoán giá xe ô tô cũ.

## **2.4. Trực quan hóa dữ liệu và kết quả dự đoán**

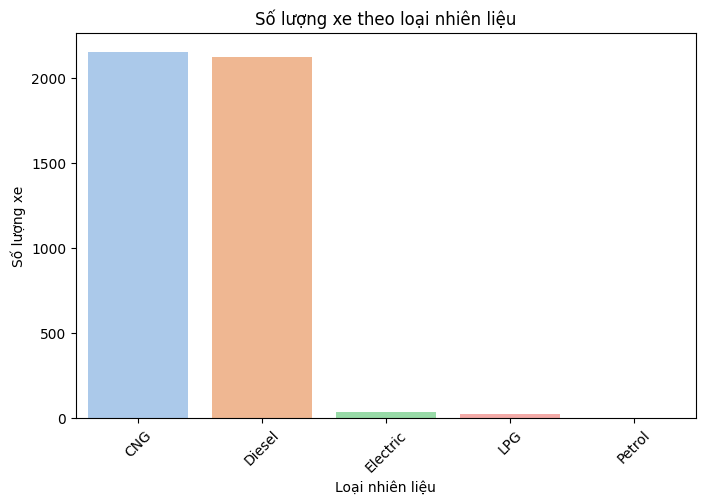
**a) Phân bố giá bán của ô tô cũ**

**Biểu đồ dưới đây cho thấy phần lớn các xe ô tô cũ được bán ở mức giá thấp (dưới 1 triệu INR), chứng tỏ thị trường tập trung chủ yếu vào các xe giá rẻ và trung bình:**

****

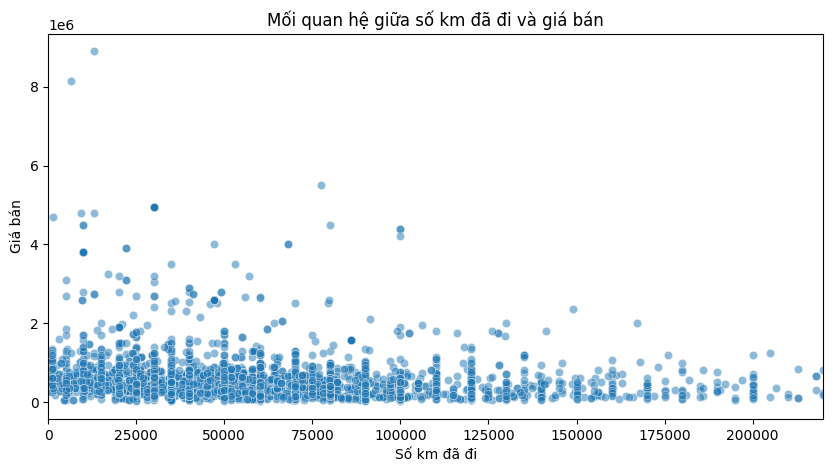
**b) Số lượng xe theo loại nhiên liệu**

Biểu đồ thể hiện rõ đa số xe sử dụng nhiên liệu Diesel và CNG, rất ít xe sử dụng nhiên liệu Electric, LPG hay Petrol, phản ánh xu hướng tiêu dùng và nguồn cung thị trường:

****

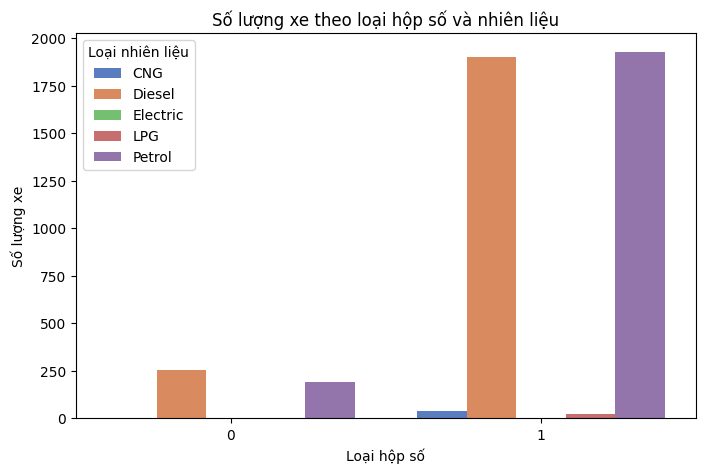
**c) Mối quan hệ giữa số km đã đi và giá bán**

Biểu đồ này thể hiện xu hướng giảm nhẹ giá xe khi số km đã đi tăng lên. Xe có số km thấp thường giữ giá tốt hơn, trong khi các xe đã đi nhiều km thường có giá thấp hơn nhiều:

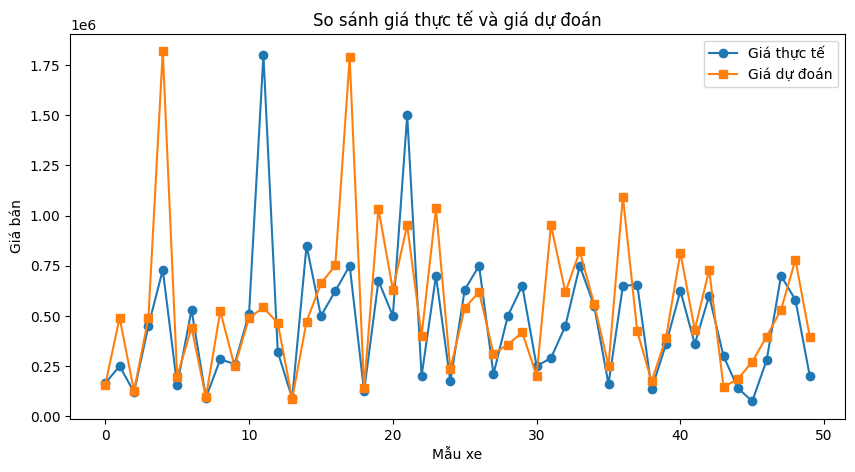
****

**d) Số lượng xe theo loại hộp số và nhiên liệu**

Biểu đồ dưới đây cho thấy xe sử dụng hộp số sàn (Manual) và nhiên liệu Diesel chiếm đa số, trong khi đó xe hộp số tự động (Automatic) sử dụng nhiên liệu Petrol cũng phổ biến, phù hợp với nhu cầu đa dạng của khách hàng:

****

**e) So sánh giá thực tế và giá dự đoán của mô hình**

****

**CHƯƠNG 3: KẾT QUẢ VÀ ĐÁNH GIÁ**

**3.1. Hiệu suất mô hình**

Để xác định hiệu quả của mô hình dự đoán giá bán xe ô tô cũ, chúng tôi đánh giá dựa trên các tiêu chí như độ chính xác, tốc độ huấn luyện, khả năng tổng quát hóa và khả năng diễn giải kết quả.

Độ chính xác: Mô hình Rừng ngẫu nhiên (Random Forest) đạt độ chính xác cao với hệ số R² = 0.94 (94%) và sai số bình phương trung bình (MSE) thấp (227,478,606.71 INR²). Điều này cho thấy mô hình có khả năng dự đoán tốt và đáng tin cậy khi áp dụng vào dữ liệu thực tế.

**Tốc độ huấn luyện:**

****

**3.2. Đánh giá khả năng tổng quát hóa**

Mô hình Random Forest đã thể hiện khả năng tổng quát hóa tốt trên tập dữ liệu thử nghiệm. Kết quả dự đoán khá sát với giá thực tế, cho thấy rằng mô hình ít bị hiện tượng quá khớp (overfitting).

**3.3. Phân tích và so sánh kết quả**

Qua so sánh với các mô hình đơn giản hơn như cây quyết định, chúng tôi nhận thấy mô hình Rừng ngẫu nhiên có độ ổn định cao hơn và dự đoán chính xác hơn. Mặc dù thời gian huấn luyện của Rừng ngẫu nhiên lâu hơn, nhưng độ chính xác cao và khả năng giảm hiện tượng quá khớp vượt trội hơn đáng kể, cho thấy hiệu quả của mô hình này rất phù hợp với bài toán dự đoán giá bán xe ô tô cũ.

**3.4. Các yếu tố ảnh hưởng đến kết quả dự đoán**

Qua phân tích dữ liệu, chúng tôi nhận thấy một số yếu tố quan trọng ảnh hưởng đáng kể đến độ chính xác của mô hình bao gồm:

Số km đã đi (km\_driven): Xe chạy nhiều km hơn thường có giá thấp hơn rõ rệt.

Năm sản xuất (year): Xe sản xuất càng gần đây càng giữ giá cao.

Loại nhiên liệu (fuel): Xe sử dụng Diesel và CNG phổ biến hơn trên thị trường, dễ dự đoán chính xác hơn.

Loại hộp số (transmission): Xe hộp số tự động thường có mức giá cao hơn và ổn định hơn so với xe số sàn.

**3.5. Tổng kết và đề xuất**

Với kết quả đánh giá trên, mô hình Rừng ngẫu nhiên là lựa chọn tối ưu cho bài toán dự đoán giá bán xe ô tô cũ do cân bằng được giữa độ chính xác cao và khả năng tổng quát tốt. Tuy nhiên, cần cân nhắc kỹ hơn về yếu tố tốc độ huấn luyện khi áp dụng trong các tình huống yêu cầu cập nhật dự đoán liên tục.

Chúng tôi đề xuất tiếp tục nghiên cứu bổ sung dữ liệu với các yếu tố như lịch sử bảo trì, tình trạng xe thực tế, và các đặc điểm thị trường cụ thể để cải thiện thêm hiệu suất dự đoán của mô hình.

**KẾT LUẬN**

Qua nghiên cứu và thực nghiệm, mô hình dự đoán giá bán xe ô tô cũ sử dụng phương pháp Rừng ngẫu nhiên đã chứng minh được tính hiệu quả và độ tin cậy cao trong việc hỗ trợ người dùng và doanh nghiệp đưa ra quyết định về giá bán xe. Kết quả đạt được không chỉ giúp nâng cao độ chính xác trong việc định giá mà còn góp phần tối ưu hóa lợi ích kinh tế, tiết kiệm thời gian và công sức.

Bên cạnh đó, nghiên cứu cũng đã xác định rõ những yếu tố quan trọng ảnh hưởng trực tiếp đến giá bán xe, làm cơ sở khoa học giúp các bên liên quan hiểu rõ thị trường và điều chỉnh chiến lược kinh doanh phù hợp. Đề tài không chỉ đáp ứng yêu cầu học thuật mà còn mang tính ứng dụng thực tiễn cao.

Trong tương lai, chúng tôi hướng tới việc phát triển và mở rộng phạm vi nghiên cứu bằng cách bổ sung thêm các đặc trưng dữ liệu mới, áp dụng các kỹ thuật hiện đại khác như Deep Learning, và kiểm thử mô hình trên nhiều tập dữ liệu khác nhau nhằm tăng khả năng tổng quát hóa và tính ứng dụng rộng rãi của mô hình dự đoán này.