|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG HÀ NỘI  **KHOA CÔNG NGHỆ THÔNG TIN**   **BÁO CÁO TỔNG KẾT** **ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN**  **NĂM HỌC 2020 - 2021**  **NGHIÊN CỨU THƯ VIỆN MÃ NGUỒN MỞ SKLEARN**  **XÂY DỰNG ỨNG DỤNG DỰ BÁO NHANH**  **DOANH THU CỦA DOANH NGHIỆP**  Thuộc lĩnh vực khoa học và công nghệ: **KHOA HỌC TỰ NHIÊN** **HÀ NỘI – 5/2021** | |
| TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG HÀ NỘI  **KHOA CÔNG NGHỆ THÔNG TIN**   **BÁO CÁO TỔNG KẾT** **ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN**  **NĂM HỌC 2020 - 2021**  **NGHIÊN CỨU THƯ VIỆN MÃ NGUỒN MỞ SKLEARN**  **XÂY DỰNG ỨNG DỤNG DỰ BÁO NHANH**  **DOANH THU CỦA DOANH NGHIỆP**  Thuộc lĩnh vực khoa học và công nghệ: **KHOA HỌC TỰ NHIÊN**  Sinh viên chịu trách nhiệm chính thực hiện: **Trần Việt Hoàng**  Nam, Nữ: **Nam** Dân tộc: **Kinh**  Lớp, khoa: **ĐH8C4 - Khoa CNTT** Năm thứ:**3** /Số năm đào tạo**: 4**  Ngành học: **Công nghệ Thông tin**  Người hướng dẫn: **ThS. Vũ Ngọc Phan**    **HÀ NỘI – 5/2021** | |
| TRƯỜNG ĐẠI HỌC  TÀI NGUYÊN VÀ MÔI TRƯỜNG HÀ NỘI  **KHOA CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM Độc lập - Tự do - Hạnh phúc** |

**THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI**

**1. Thông tin chung:**

**-** Tên đề tài: Nghiên cứu thư viện mã nguồn mở Sklearn xây dựng ứng dụng dự báo nhanh doanh thu của doanh nghiệp

- Sinh viên thực hiện: **Bùi Minh Anh**

**Đặng Việt Hoàng**

**Trần Việt Hoàng**

**Nguyễn Như Ý**

- Lớp: **ĐH8C4**  Khoa: **Công nghệ Thông tin**

- Năm thứ: **3** Số năm đào tạo: **4**

- Người hướng dẫn: **ThS. Vũ Ngọc Phan**

**2. Mục tiêu đề tài:**

Áp dụng những kiến thức đã học tập tại trường, kết hợp với nghiên cứu các công nghệ mới để xây dựng ứng dụng có thể áp dụng giải quyết các bài toán thực tế.

Nghiên cứu khả năng ứng dụng học máy trong lĩnh vực kinh doanh

Nghiên cứu cách sử dụng thư viện Sklearn và các thuật toán máy học phổ biến như nhận dạng và phân cụm dữ liệu.

Sử dụng ngôn ngữ lập trình Python xây dựng ứng dụng dự báo

**3. Tính mới và sáng tạo:**

Thế giới đang trải qua cuộc cách mạng công nghiệp 4.0, trong đó, những yếu tố cốt lõi của kỹ thuật số được tác động trực tiếp hay còn gọi là bộ khung của cuôc cách mạng này trí tuệ nhân tạo, Internet of Things và Big Data. Machine Learning là một hướng nghiên cứu trong AIvà đối với thế giới kinh doanh, những tiến bộ của Machine Learning đã tạo ra một tác động sâu sắc. Việc đưa vào Machine Learning có thể chứng minh giá trị trong việc tăng hiệu quả, năng suất và tốc độ, đồng thời góp phần đưa ra các quyết định dựa trên dữ liệu quan trọng để thúc đẩy sự phát triển của một công ty.

**4. Kết quả nghiên cứu:**

Nghiên cứu sẽ khái quát hóa khiến thức về Machine Learning, nghiên cứu về các thuật toán học máy có trong thư viện Sklearn và trình bày phương pháp sử dụng thư viện này trong việc đưa ra các dự báo

**5. Đóng góp về mặt kinh tế - xã hội,** **giáo dục và đào tạo, an ninh, quốc phòng và khả năng áp dụng của đề tài:**

Đưa ra một hướng dự báo về kết quả kinh doanh của các doanh nghiệp

**6.** **Công bố khoa học của sinh viên từ kết quả nghiên cứu của đề tài** *(ghi rõ tên tạp chí nếu có)* hoặc nhận xét, đánh giá của cơ sở đã áp dụng các kết quả nghiên cứu *(nếu có)*:

|  |
| --- |
| *Ngày….tháng….năm 20….*  **Sinh viên chịu trách nhiệm chính thực hiện đề tài**  *(Ký, ghi rõ họ và tên)* |

**Nhận xét của người hướng dẫn về những đóng góp khoa học của sinh viên thực hiện đề tài** *(phần này do người hướng dẫn ghi):*

*………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………*

|  |  |
| --- | --- |
| **Xác nhận của trường đại học**  *(ký tên và đóng dấu)* | *Ngày……..tháng……..năm 201…*  **Người hướng dẫn**  *(Ký, ghi rõ họ và tên)* |

# MỤC LỤC

[MỤC LỤC](#_Toc72695662)

[DANH MỤC CÁC TỪ VIẾT TẮT](#_Toc72695663)

[DANH MỤC CÁC HÌNH](#_Toc72695664)

[MỞ ĐẦU 1](#_Toc72695665)

[CHƯƠNG 1. HỌC MÁY VÀ ỨNG DỤNG HỌC MÁY TRONG MÔ HÌNH KINH DOANH CỦA DOANH NGHIỆP 2](#_Toc72695666)

[1.1 Tổng quan về học máy 2](#_Toc72695667)

[1.1.1 Lịch sử của học máy 2](#_Toc72695668)

[1.1.2 Xu hướng phát triển của học máy trên thế giới và Việt Nam 4](#_Toc72695669)

[1.2 Ứng dụng học máy trong kinh doanh của doanh nghiệp 5](#_Toc72695670)

[1.2.1 Machine Learning biến đổi doanh nghiệp 6](#_Toc72695671)

[1.2.2 Các trường hợp sử dụng máy học trong doanh nghiệp 9](#_Toc72695672)

[1.2.3 Tích hợp Machine Learning vào kinh doanh của doanh nghiệp 11](#_Toc72695673)

[Chương 2. NGÔN NGỮ PYTHON, THƯ VIỆN SCIKIT-LEARN VÀ CÁC THUẬT TOÁN LIÊN QUAN 13](#_Toc72695674)

[2.1 Ngôn ngữ lập trình Python 13](#_Toc72695675)

[2.1.1 Ưu, nhược điểm của ngôn ngữ Python 13](#_Toc72695676)

[2.1.2 Python trong Machine Learning 15](#_Toc72695677)

[2.2 Thư viện Sklearn và các thuật toán máy học được hỗ trợ 17](#_Toc72695678)

[2.2.1 Thư viện Sklearn 17](#_Toc72695679)

[2.2.2 Một số thuật toán được hỗ trợ từ thư viện SkLearn 19](#_Toc72695680)

[Chương 3. PHƯƠNG PHÁP THỰC HIỆN VÀ ĐÁNH GIÁ KẾT QUẢ 25](#_Toc72695681)

[3.1 Khai báo thư viện và dữ liệu cần phân tích 26](#_Toc72695682)

[3.2 Trực quan hóa dữ liệu 27](#_Toc72695683)

[3.3 Tiền xử lý dữ liệu 28](#_Toc72695684)

[3.4 Training and Test 29](#_Toc72695685)

[3.5 Xây dựng mô hình 30](#_Toc72695686)

[3.5.1 Xây dựng mô hình với thuật toán Naive-Bayes 30](#_Toc72695687)

[3.5.2 Xây dựng mô hình với thuật toán LinearSVC 31](#_Toc72695688)

[3.5.3 Xây dựng mô hình với thuật toán K-Neighbors Classifier 32](#_Toc72695689)

[3.5.4 Xây dựng mô hình với thuật toán Random Forest, Logistic Regression, Perceptron 32](#_Toc72695690)

[3.6 So sánh hiệu suất 33](#_Toc72695691)

[3.7 Phân tích và đưa ra kết quả 35](#_Toc72695692)

[KẾT LUẬN VÀ KIẾN NGHỊ 41](#_Toc72695693)

[TÀI LIỆU THAM KHẢO 43](#_Toc72695694)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **TỪ VIẾT TẮT** | **TIẾNG ANH** | **TIẾNG VIỆT** |
| AI | Artificial intelligence | Trí tuệ nhân tạo |
| API | Application Programming Interface | Phương thức trung gian kêt nối ứng dụng và thư viện |
| ML | Machine Learning | Học máy |

# DANH MỤC CÁC HÌNH

[*Hình 2 1 Mức độ phổ biến của các ngôn ngữ lập trình theo Stack Overflow 16*](#_Toc72693714)

[*Hình 2 2 Tỷ lệ sử dụng ngôn ngữ Python trong các lĩnh vực 17*](#_Toc72693715)

[*Hình 2 3 Sơ đồ hoạt động Naive-Bayes 20*](#_Toc72693716)

[*Hình 2 4 Sơ đồ hoạt động LinearSVC. 21*](#_Toc72693717)

[*Hình 2 5 Sơ đồ hoạt động K-Neightbors Classifier. 21*](#_Toc72693718)

[*Hình 2 6 Sơ đồ hoạt động Support Vector Machines. 22*](#_Toc72693719)

[*Hình 2 7. Sơ đồ hoạt động Random Forest. 23*](#_Toc72693720)

[*Hình 2 8 Sơ đồ hoạt động Logistic Resgression. 25*](#_Toc72693721)

[*Hình 3 1 Khai báo thư viện. 26*](#_Toc72742254)

[*Hình 3 2 Hình ảnh bản ghi của dữ liệu mẫu. 27*](#_Toc72742255)

[*Hình 3 3 Mã lệnh trực quan hóa dữ liệu sử dụng Seaborn. 27*](#_Toc72742256)

[*Hình 3 4 Trực quan hóa dữ liệu với biểu đồ cột. 28*](#_Toc72742257)

[*Hình 3 5 Bản ghi dữ liệu khi đã sử dụng phương pháp LabelEncoder. 29*](#_Toc72742258)

[*Hình 3 6 Bản ghi dữ liệu khi đã lọc và chọn trường để training. 29*](#_Toc72742259)

[*Hình 3 7 Mã lệnh chia nhỏ khung dữ liệu. 30*](#_Toc72742260)

[*Hình 3 8 Mã lệnh và kết quả độ chính xác thuật toán Naive-Bayes 31*](#_Toc72742261)

[*Hình 3 9 Mã lệnh và kết quả độ chính xác thuật toán LinearSVC. 31*](#_Toc72742262)

[*Hình 3 10 Mã lệnh và kết quả độ chính xác thuật toán KneighborsClassifier. 32*](#_Toc72742263)

[*Hình 3 11 Mã lệnh và kết quả độ chính xác thuật toán RandomForest. 32*](#_Toc72742264)

[*Hình 3 12 : Mã lệnh và kết quả độ chính xác thuật toán Logistic Regression. 33*](#_Toc72742265)

[*Hình 3 13 Mã lệnh và kết quả độ chính xác thuật toán Perceptron. 33*](#_Toc72742266)

[*Hình 3 14 Mã lệnh và kết quả trả về độ chính xác của tất cả thuật toán. 34*](#_Toc72742267)

[*Hình 3 15 Tỉ lệ của các trường khi đã training. 34*](#_Toc72742268)

[*Hình 3 16 Độ chính xác thuật toán sau khi xử lý. 35*](#_Toc72742269)

[*Hình 3 17 Độ chính xác trung bình cả thuật toán 35*](#_Toc72742270)

[*Hình 3 18 Độ chính xác trung bình khi tiếp tục training. 35*](#_Toc72742271)

[*Hình 3 19 Mã lệnh xây dựng biều đồ đường cong sử dụng precision\_recall\_curve. 36*](#_Toc72742272)

[*Hình 3 20 Biểu đồ đường cong của precision, recall và threshold. 36*](#_Toc72742273)

[*Hình 3 21 Biểu đồ precision và recall. 37*](#_Toc72742274)

[*Hình 3 22 Mã lệnh vẽ biểu đồ tỉ lệ tích cực và tiêu cực. 37*](#_Toc72742275)

[*Hình 3 23 Biểu đồ tỉ lệ tích cực và tiêu cực. 38*](#_Toc72742276)

[*Hình 3 24 Kết quả hiệu suất của mô hình thuật toán. 38*](#_Toc72742277)

[*Hình 3 25 Kết quả đánh giá của toàn bộ mô hình. 39*](#_Toc72742278)

[*Hình 3 26 Mã lệnh dự đoán kết quả 40*](#_Toc72742279)

[*Hình 3 27 Biểu đồ và kết quả 40*](#_Toc72742280)

# MỞ ĐẦU

1. **Lý do lựa chọn đề tài**

Thế giới đang trải qua cuộc cách mạng công nghiệp 4.0, cuộc cách mạng này sẽ diễn ra trên 3 lĩnh vực chính đó là công nghệ sinh học, kỹ thuật số và vật lý. Trong đó, những yếu tố cốt lõi của kỹ thuật số được tác động trực tiếp hay còn gọi là bộ khung của cách mạng công nghiệp 4.0 đó là trí tuệ nhân tạo, Internet of Things và Big Data. Trí tuệ nhân tạo (Artificial intelligence -AI) là trí tuệ do con người lập trình tạo nên với mục tiêu giúp máy tính có thể tự động hóa các hành vi thông minh như con người. Tuy nhiên trí tuệ nhân tạo khác với việc lập trình logic ở chỗ AI ứng dụng các hệ thống học máy (machine learning) để mô phỏng trí tuệ của con người trong các xử lý mà con người làm tốt hơn máy tính.

Machine Learning là một hướng nghiên cứu trong AI, tập trung vào việc tạo ra các cỗ máy có khả năng học hỏi mà không cần phải được lập trình một cách cụ thể. Học hỏi là một kỹ năng quan trọng bậc nhất đối với sự hình thành trí tuệ con người, do đó nếu chúng ta muốn xây dựng các hệ thống AI có trí tuệ như con người thì chúng ta trước hết phải tạo ra các cỗ máy có khả năng tự học hỏi dựa trên các kinh nghiệm mà chúng thu thập được trong suốt quá trình hoạt động. Hiện Machine Learning đã đạt được những thành công vang dội khiến nó trở thành một hướng nghiên cứu chính trong AI và được áp dụng trên toàn thế giới.

Đối với thế giới kinh doanh, những tiến bộ của Machine Learning đã tạo ra một tác động sâu sắc. Với các hoạt động và quy trình kinh doanh trải rộng ở các cấp độ khác nhau, việc đưa vào Machine Learning có thể chứng minh giá trị trong việc tăng hiệu quả, năng suất và tốc độ. Machine Learning sử dụng các tập dữ liệu lớn được tạo và lưu trữ trong doanh nghiệp để phân tích qua đó xây dựng cho một cách tiếp cận và chiến lược mới cho hoạt động kinh doanh, giúp đưa ra các quyết định dựa trên dữ liệu quan trọng để thúc đẩy sự phát triển của một công ty.

**Scikit-learn** (viết tắt là Sklearn) là một thư viện mã nguồn mở dành cho Machine Learning và cũng được sử dụng trong Data Science. Đây là công cụ rất mạnh mẽ và thông dụng với cộng đồng Python, được thiết kế trên nền NumPy và SciPy. Scikit-learn chứa hầu hết các thuật toán Machine Learning hiện đại nhất, đi kèm với documentations và được cập nhật thường xuyên. Công cụ này cung cấp việc sử dụng API và tìm kiếm ngẫu nhiên dễ dàng. Nhưng lợi thế chính trong việc sử dụng Scikit-Learn, là tốc độ trong khi thực hiện các đánh giá khác nhau trong bộ dataset.

Với những phân tích ở trên, nhóm tác giả đã lựa chọn “*Nghiên cứu thư viện mã nguồn mở Sklearn xây dựng ứng dụng dự báo nhanh doanh thu của doanh nghiệp”* làm đề tài nghiên cứu khoa học của mình.

1. **Mục tiêu đề tài**

2.1 Mục tiêu tổng quát:

Áp dụng những kiến thức đã học tập tại trường, kết hợp với nghiên cứu các công nghệ mới để xây dựng ứng dụng có thể áp dụng giải quyết các bài toán thực tế.

2.2 Mục tiêu cụ thể:

- Nghiên cứu khả năng ứng dụng học máy trong lĩnh vực kinh doanh

- Nghiên cứu cách sử dụng thư viện Sklearn và các thuật toán máy học phổ biến như nhận dạng và phân cụm dữ liệu.

- Sử dụng ngôn ngữ lập trình Python xây dựng ứng dụng dự báo.

1. **Phương pháp nghiên cứu**

- Phương pháp phân tích và tổng hợp lý thuyết: nghiên cứu các tài liệu học thuật, phân tích, tìm hiểu chuyên sâu và tổng hợp kiến thức về học máy và các thuật toán học máy phổ biến.

- Phương pháp chuyên gia: sử dụng trí tuệ của đội ngũ chuyên gia, của thầy cô giảng viên để xem xét, tìm ra giải pháp nghiên cứu tối ưu nhất.

- Phương pháp điều tra, thu thập thông tin: thu thập các nguồn dữ liệu về kinh doanh của doanh nghiệp bán phụ tùng ô tô nhằm phục vụ xây dựng chương trình ứng dụng.

- Phương pháp thực nghiệm: xây dựng các kịch bản chạy thử của phần mềm để kiểm tra khả năng hoạt động chính xác theo thiết kế.

# CHƯƠNG 1. HỌC MÁY VÀ ỨNG DỤNG HỌC MÁY TRONG MÔ HÌNH KINH DOANH CỦA DOANH NGHIỆP

## 1.1 Tổng quan về học máy

### 1.1.1 Lịch sử của học máy

Lịch sử của học máy là một quá trình dài dựa trên các thuật toán từ cả trăm năm trước, cũng như những phát triển về công nghệ máy tính trong 7 thập kỷ qua.

**Trước 1940:** Rất nhiều thuật toán đằng sau phép học máy ngày nay đã xuất hiện từ trước khi chiếc máy tính đầu tiên ra đời và bắt nguồn từ phép thống kê. Trong đó các định lý Bayes (1812) của Pierre-Simon Laplace, phép bình phương nhỏ nhất cho dữ liệu hồi quy (1805) của Adrien-Marie Legendre, kỹ thuật phân tích “Chuỗi Markov” (1913) của Andrey Markov đã góp vai trò to lớn trong sự phát triển của học máy hiện đại.

**1948:** Vào cuối những năm 1940, các máy tính với chương trình được lưu trữ sẵn đã có thể lưu trữ chương trình và dữ liệu trong cùng một bộ nhớ. Những chiếc máy tính này đã mở đầu cách mạng máy tính hiện đại đầu tiên, với chiếc Manchester Small-Scale Experimental Machine được tạo ra bởi EDSAC, Đại học Cambridge vào năm 1948, và chiếc Manchester Mark 1 tạo ra bởi EDVAC, Đại học Pennsylvania vào năm 1949.

**1950:** Nghiên cứu “Computing Machinery and Intelligence” (Máy tính và Trí thông minh), được xuất bản năm 1950 bởi Alan Turing đã đưa ra một câu hỏi mà cho tới hiện tại vẫn chưa được giải đáp: “Liệu máy móc có thể suy nghĩ?” Nghiên cứu đã đi vào sức mạnh của máy tính, và là một trong những bài viết học thuật đầu tiên tập trung mô tả cách phát triển trí tuệ nhân tạo. Một trong những thử nghiệm nổi tiếng của nghiên cứu là “trò chơi mô phỏng” (imitation game), trong đó mức độ thông minh của máy tính sẽ được đánh giá bằng khả năng phân biệt người thật và máy tính khác qua tin nhắn.

**1951:** Marvin Minsky và Dean Edmonds đã xây dựng mạng nơ-ron thần kinh đầu tiên (một hệ thống máy tính mô phỏng hoạt động của não bộ). Với khả năng học hỏi và rút kinh nghiệm qua nhiều lần thử, máy SNARC có thể di chuyển trong các mê cung giống như chuột thí nghiệm. Máy tính này được xây dựng dựa trên các nguyên lý liên kết – mô phỏng một mạng lưới các đơn vị cơ bản, rồi từ đó xây dựng “trí thông minh”.

**1974:** Sự thất bại trong dịch máy (Machine Translation), cũng như những kỳ vọng quá xa vời về AI, học máy đã dẫn tới ngân sách cho nghiên cứu bị cắt giảm nghiêm trọng. Chỉ đến cuối những năm 1980, với những phương pháp tiếp cận mới, việc nghiên cứu đã có những tiến triển, các thuật ngữ mới như Tin học (Informatics), Học máy (Machine Learning), và Trí thông minh máy tính (Computational intelligence) ra đời.

**1996:** Thuật ngữ AI dần trở nên phổ biến với công chúng qua sự kiện máy tính Deep Blue của IBM đánh bại kiện tướng cờ vua thế giới Garry Kasparov. Deep Blue đã sử dụng 480 con chip đặc biệt chỉ dùng để chơi cờ, giúp cỗ máy này tìm kiếm từ 6-20 nước cờ trước cho mỗi vị trí, “học hỏi” được qua quan sát hàng nghìn ván cờ.

**2006:** Truyền ngược (Backpropagation) là một yếu tố cốt lõi trong Học máy, được sử dụng để đào tạo các hệ nơ-ron thần kinh sâu. Truyền ngược chỉ được ứng dụng sau khi Geoff Hinton và một số nhà nghiên cứu phát hiện ra hiệu quả của nó thông qua các bộ xử lý hiện đại. Tuy nhiên, vào năm 2017, Hinton lại bày tỏ lo ngại về hạn chế của truyền ngược trong xây dựng hệ thống Học máy, và chỉ ra sự cần thiết của một hệ thống mới.

**2014:** DeepMind trở nên đặc biệt được chú ý sau khi thể hiện khả năng chơi game dựa vào hành vi của các đơn vị pixel trên màn hình, cũng như khả năng phát triển hệ nơ-ron thần kinh có thể truy cập bộ nhớ ngoài – Máy nơ-ron Turing (Neural Turing Machine).

**2016:** Sau chiến thắng của DeepBlue, AlphaGo- một robot sử dụng DeepMind đã đánh bại kỳ thủ cờ vây số 2 thế giới Lee Sedol vào tháng 3/2016, và kiện tướng cờ vây Ke Jie vào năm 2017. Để có thể đạt được những chiến thắng lịch sử này, AlphaGo đã trải qua huấn luyện với cả con người và máy móc, ứng dụng thuật toán sơ đồ cây Monte Carlo.

### 1.1.2 Xu hướng phát triển của học máy trên thế giới và Việt Nam

Việc ứng dụng Machine Learning như là một quy trình ngày càng phổ biến và được chấp nhận trong một loạt các ngành công nghiệp và ứng dụng. Hầu hết các doanh nghiệp đều có xu hướng sử dụng các ứng dụng mã nguồn mở và phần mềm quản lý dữ liệu để phân tích các mạng nơron hệ thống quan trọng, xúc tiến các quy trình chuỗi cung ứng hoặc xác định kỳ vọng của khách hàng.

Theo báo cáo về các mối đe dọa của McAfee Labs năm 2018, trong tương lai, Machine learning sẽ được triển khai để phát hiện xâm nhập, lừa đảo và spam. Nó cũng có thể được sử dụng để phát hiện phần mềm độc hại trong lĩnh vực an ninh mạng để tốc độ máy đạt cường độ cao trong môi trường không có máy chủ. Với số lượng ngày càng tăng của các cuộc tấn công mạng, AI và Machine Learning đều giúp các công ty cải thiện các phương pháp bảo mật. Các nhà phát triển có thể triển khai Blockchain như một cách khả thi để chống lại sự xâm nhập hệ thống và đảm bảo việc duy trì an toàn dữ liệu.

Trong bản báo cáo Nghiên cứu thị trường khoa học dữ liệu và [học máy](https://www.thegioimaychu.vn/blog/thuat-ngu/machine-learning/) lần thứ 6 năm 2019 của Dresner Advisory Services được công bố cho thấy các sáng kiến ​​tiên tiến liên quan đến khoa học dữ liệu và học máy, bao gồm khai thác dữ liệu, thuật toán tiên tiến và phân tích dự đoán được xếp hạng ưu tiên thứ 8 trong số 37 công nghệ và sáng kiến ​​được khảo sát trong nghiên cứu.

Tại Việt Nam, Viện Nghiên cứu Trí tuệ nhân tạo VinAI Research có ba công trình nghiên cứu được công bố tại Hội nghị Quốc tế về máy học (ICML) 2020. Ba công trình này tập trung vào các vấn đề quan trọng trong nghiên cứu AI hiện tại, gồm: Phát triển phương pháp tính toán tối ưu để so sánh các phân bố từ dữ liệu lớn; Học sâu các biểu diễn quan trọng từ dữ liệu ảnh, video cho bài toán điều khiển tối ưu và Đề xuất phương pháp suy diễn hiệu quả cho các hệ thống động nơron phi tuyến tính phức tạp.

Nghiên cứu về so sánh giữa các phân bố từ dữ liệu lớn là nền tảng cho nhiều thuật toán học máy, góp phần thúc đẩy các thuật toán học máy không giám sát - là một trong những vấn đề được quan tâm nhất trong trí tuệ nhân tạo, thị giác máy tính, xử lý và hiểu ngôn ngữ tự nhiên. Trong khi đó, các nghiên cứu về biểu diễn của dữ liệu và hệ thống động phi tuyến tính là nền tảng cho việc đột phá trong phát triển robot tự động hóa hoặc xe hơi tự lái.

## 1.2 Ứng dụng học máy trong kinh doanh của doanh nghiệp

Các doanh nghiệp luôn đi đầu trong việc áp dụng sớm các công nghệ mới. Những tiến bộ trong điện toán như Machine Learning đã tạo ra một tác động đáng chú ý đối với thế giới kinh doanh. Với các hoạt động và quy trình kinh doanh trải rộng ở các cấp độ khác nhau, việc đưa Machine Learning có thể chứng minh giá trị trong việc tăng hiệu quả, năng suất và tốc độ. Machine Learning đã tìm thấy sự chấp nhận rộng rãi giữa các doanh nghiệp. Tạp chí [MIT Technology Review (Viện công nghệ Massachusetts- Mỹ) và Google Cloud](https://s3.amazonaws.com/files.technologyreview.com/whitepapers/MITTR_GoogleforWork_Survey.pdf) gần đây đã xuất bản một báo cáo dựa trên các nghiên cứu của họ về Machine Learning và việc áp dụng nó. Báo cáo nói rằng khoảng 60 phần trăm số người được hỏi đã triển khai Machine Learning vào hoạt động kinh doanh của họ và dự đoán thị trường Machine Learning sẽ có giá trị khoảng 8,81 tỷ USD vào năm 2022. Nghiên cứu đã cho thấy thị trường mở rộng cho Machine Learning và những lợi ích rõ ràng mà nó mang cho các doanh nghiệp.

Machine Learning biến dữ liệu thô thành những hiểu biết có ý nghĩa để giải quyết các thách thức kinh doanh phức tạp. Kể từ khi được giới thiệu, rất nhiều công ty trên khắp thế giới đã được hưởng lợi từ việc sử dụng ML với những cải tiến lớn trong hoạt động kinh doanh của họ. Google là một trong những công nghệ lớn sử dụng điện toán tiên tiến như Machine và Deep Learning để làm cho công cụ tìm kiếm của nó tinh tế và mạnh mẽ hơn.

Các doanh nghiệp có lợi thế lớn nhất khi nói đến Machine Learning vì sự phụ thuộc ngày càng tăng và áp dụng công nghệ sớm. Tính khả dụng của điện toán nhanh hơn, lưu trữ có thể mở rộng và dữ liệu dễ truy cập đã dẫn đến các nền tảng Machine Learning trở nên phổ biến hơn. Với số lượng dữ liệu khổng lồ được lưu trữ và tạo bởi các doanh nghiệp, một hệ thống hoặc nền tảng máy tính nhúng Machine Learning có thể giúp khai thác dữ liệu này để tự tìm giải pháp do đó vẫn là một phần không thể thiếu.

### 1.2.1 Machine Learning biến đổi doanh nghiệp

Trong các doanh nghiệp, dữ liệu có thể tồn tại cả ở dạng có cấu trúc và không có cấu trúc với ước tính chiếm khoảng 90%. Machine Learning rất quan trọng để xử lý dữ liệu phi cấu trúc, việc này tốn nhiều thời gian nếu được thực hiện thủ công. Bên cạnh đó, nó có thể tạo ra những hiểu biết mới bằng cách sử dụng dữ liệu có sẵn và biến nó thành một hình thức có cấu trúc và được gắn nhãn để dễ dàng truy cập. Một số cách mà Machine Learning sẽ thay đổi doanh nghiệp là:

1. **Cung cấp trực quan hóa dữ liệu hợp lý**

Khả năng trực quan hóa dữ liệu là một trong những lợi thế chính khi sử dụng Machine Learning. Với toàn bộ thế giới sẽ trực quan, việc áp dụng một phương pháp trực quan hóa các kho dữ liệu rộng lớn có thể giúp nhân viên trong một công ty biết sâu về một dự án hoặc thông tin. Trực quan hóa dữ liệu có thể rút ra ý nghĩa từ một chồng dữ liệu lớn bằng cách sử dụng kết hợp các yếu tố tĩnh và tương tác.

Machine Learning có thể giúp cải thiện phát hiện dữ liệu bằng cách tìm hiểu những hiểu biết có thể hành động từ các tập dữ liệu lớn. Các nhân viên có thể hiểu rõ hơn thông qua những hình dung của các bộ dữ liệu khổng lồ. Điều này sẽ kích thích họ thực hiện hành động đúng đắn, điều này sẽ thúc đẩy hiệu quả và đổi mới trong tổ chức. Hơn nữa, Machine Learning có thể mang lại lợi thế bổ sung vào trực quan hóa dữ liệu bằng cách:

- Cho phép phân tích dự đoán thông qua việc sử dụng các thuật toán tìm hiểu và phát hiện các mẫu để cải thiện độ chính xác và tốc độ dữ liệu.

- Xây dựng sự hiểu biết tốt hơn về khách hàng bằng cách sử dụng dữ liệu được thu thập để cung cấp các dịch vụ được cá nhân hóa và tạo ra trải nghiệm nổi bật.

- Cải thiện tính năng động của trực quan hóa dữ liệu với kết quả thời gian thực, cung cấp chi tiết về các quy trình kinh doanh tổng thể.

Một hệ thống Machine Learning có thể tạo ra một chi tiết trực quan có ý nghĩa từ cả dữ liệu có cấu trúc và không cấu trúc.

1. **Tạo dữ liệu có giá trị về hành vi của khách hàng**

Dữ liệu được tạo ra liên tục khi khách hàng tương tác hoặc mua sản phẩm hoặc sử dụng dịch vụ từ các tổ chức. Việc thu thập dữ liệu này tiếp tục nếu các dịch vụ được cung cấp trong các lĩnh vực kỹ thuật số thông qua các bản nâng cấp và các hình thức khác. Tận dụng những dữ liệu này có thể tạo ra một cái nhìn sâu sắc đáng chú ý về hành vi của khách hàng, tạo thành nền tảng của mọi doanh nghiệp ngày nay.

Machine Learning có thể thu thập dữ liệu người dùng, từ đó tạo ra các giải pháp và trải nghiệm phù hợp. Khi một doanh nghiệp sử dụng thuật toán Machine Learning, nó sẽ trải qua quá trình đào tạo và thử nghiệm trong các tình huống khác nhau để rút ra ý nghĩa từ dữ liệu thu được của khách hàng.

Ngoài ra, dữ liệu thu được từ Machine Learning về hành vi của khách hàng có thể ứng dụng có những ứng dụng khác như:

- Xác định các mẫu mua bằng cách xem lại lịch sử duyệt và mua hàng: Điều này sẽ lần lượt giúp cung cấp các dịch vụ tốt nhất cho khách hàng, giúp tăng khả năng có nhiều lượt truy cập và chuyển đổi hơn trong tương lai.

- Đề xuất đề xuất sản phẩm cho khách hàng để thuyết phục họ mua: Machine Learning đã được sử dụng để đề xuất các đề xuất sản phẩm của các công ty thương mại điện tử hàng đầu như Amazon và để quản lý nội dung của công ty ứng dụng web và di động Pinterest…

- Machine Learning thúc đẩy sự hài lòng và trải nghiệm của khách hàng tốt hơn: Machine Learning có thể phân tích bản ghi các cuộc gọi được thực hiện với khách hàng, từ đó cung cấp thông tin hữu ích cho các nhân viên. Điều này góp phần xây dựng “lòng trung thành” của khách hàng với các sản phẩm và dịch vụ của doanh nghiệp cùng với việc giảm chi phí và thời gian chăm sóc khách hàng.

1. **Thay thế lao động của con người với tự động hóa**

Nhập dữ liệu thủ công luôn có thể dẫn đến một số lỗi nhất định, điều này sẽ ảnh hưởng đáng kể đến quy trình kinh doanh. Và với số lượng dữ liệu khổng lồ, công việc cần thiết này rất tốn kém thời gian và công sức, hơn thế luôn tiềm ẩn khả năng không chính xác hoặc sai lầm trong quá trình nhập liệu. [Machine Learning sử dụng thuật toán mô hình dự đoán](https://whatis.techtarget.com/definition/predictive-maintenance-PdM) hoàn toàn có thể ngăn chặn các lỗi và sự không chính xác như vậy trong dữ liệu đầu vào. Bằng cách làm việc với dữ liệu hiện có, một hệ thống Machine Learning có thể xử lý và giúp giải phóng nhân viên chịu trách nhiệm nhập dữ liệu, giúp doanh nghiệp có thể sử dụng những nhân viên này trong các nhiệm vụ cần thiết khác.

Ý tưởng tự động hóa được bằng Machine Learning là điều mà hầu hết các công ty đã ứng dụng trong nhiều năm. Tự động hóa các công việc quan trọng mang lại khoản tiết kiệm đáng kể cho một công ty, đồng thời cải thiện hiệu quả các quy trình bằng cách tăng tốc và xử lý chính xác.

1. **Tăng cường bảo mật**

Với các mối đe dọa ngày càng tăng của phần mềm độc hại và đánh cắp thông tin, an ninh mạng đã trở thành một phần quan trọng của bất kỳ tổ chức nào. Nhiều biện pháp kiểm soát mối đe dọa được triển khai tại chỗ để che chắn các hệ thống máy tính và dữ liệu được lưu trữ. Chúng có thể nhanh chóng trở nên lỗi thời trong việc cung cấp sự bảo vệ liên tục vì phần mềm độc hại và các dạng virus máy tính khác đang phát triển để vượt qua các hệ thống phòng thủ không gian mạng này.

Machine Learning có thể giúp tăng cường tính toán và bảo mật dữ liệu tổng thể của một tổ chức. Machine Learning sẽ đánh giá tình trạng bảo mật, sau đó sẽ xây dựng các chiến lược và mô hình dựa trên rủi ro để bảo vệ các hệ thống khỏi các can thiệp bên ngoài. Hơn nữa, nó có thể phân tích dữ liệu bảo mật trước đó để dự đoán các mối đe dọa và xác suất của chúng trong tương lai.

Sử dụng một mô hình Machine Learning mạnh mẽ sẽ giúp đưa ra các quyết định dựa trên dữ liệu để tăng cường an ninh mạng. Nó có thể liên tục theo dõi mạng và hệ thống xem có bất kỳ sự bất thường nào trong lưu lượng truy cập và các khía cạnh khác để ngăn chặn truy cập trái phép hoặc vi phạm vào dữ liệu nhạy cảm được lưu trữ.

1. **Sử dụng bảo trì dự đoán để giảm thiểu rủi ro**

Bảo trì là phần quan trọng của bất kỳ doanh nghiệp nào, đặc biệt là những doanh nghiệp sản xuất. Hầu hết các công ty này thường xuyên sử dụng một quy trình bảo trì để sớm xác định điểm yếu và thực hiện các biện pháp khắc phục cần thiết. Tuy nhiên, họ có xu hướng tụt lại phía sau trong việc xác định bất kỳ vấn đề nào trước đó, dẫn đến thiệt hại không thể khắc phục trong thời gian dài.

Machine Learning sử dụng dữ liệu từ các cảm biến và các đơn vị khác để phát hiện các mẫu thông báo cho bất kỳ rủi ro hoặc điểm yếu nào, có thể được giải quyết sớm để ngăn ngừa các lỗi và do đó giảm thiểu rủi ro. Mục đích là để dự đoán thời gian một máy có thể thất bại dựa trên đánh giá dữ liệu. Nó có hiệu quả cao trong việc chỉ ra thiết bị hoặc đơn vị cụ thể nào cần thay thế hoặc sửa chữa.

### 1.2.2 Các trường hợp sử dụng máy học trong doanh nghiệp

1. **Xử lý ngôn ngữ tự nhiên**

Với lượng dữ liệu khổng lồ, hệ thống máy tính sử dụng Machine Learning có thể giúp ích rất nhiều trong việc xử lý ngôn ngữ tự nhiên. Bằng cách sử dụng Deep Learning, một phần quan trọng của Machine Learning, máy tính sẽ có khả năng hiểu và xử lý văn bản và lời nói như con người.

Sự gia tăng của các trợ lý kỹ thuật số như Siri, Google Assistant, Cortana... là một số ví dụ về xử lý ngôn ngữ tự nhiên đang được sử dụng. Việc sử dụng hệ thống Machine Learning có khả năng xử lý ngôn ngữ tự nhiên có giá trị lớn đối với một doanh nghiệp, đặc biệt là trong lĩnh vực quản lý khách hàng.

Trong kinh doanh, xử lý ngôn ngữ tự nhiên tìm thấy ứng dụng dưới dạng chatbot tương tác với khách hàng và cung cấp thông tin cho bất kỳ truy vấn nào của họ. Bên cạnh đó, nó cũng có thể hỗ trợ cải thiện hiệu quả trong lĩnh vực tuyển dụng việc làm.

1. **Hệ thống quản lý đào tạo**

Việc đào tạo nhân viên mới và lực lượng lao động hiện có là ưu tiên và nhu cầu của hầu hết các công ty, tổ chức. Nền tảng của phần mềm LMS (Phần mềm quản lý học tập, đào tạo) sử dụng Machine Learning có thể phân tích dữ liệu người dùng để mang lại trải nghiệm học tập được cá nhân hóa và dựa trên giá trị.

Các nền tảng đào tạo dựa trên năng lực như UpSkill đã được sử dụng để hỗ trợ các công ty tối đa hóa năng suất lao động của họ bằng cách xác định tiềm năng của họ và truyền đạt các khóa đào tạo cần thiết. Bằng cách kết hợp nó với HR, một hệ thống nhúng Machine Learning có thể giúp xác định xu hướng và sử dụng nó để cải thiện sự hài lòng và kinh nghiệm của nhân viên tại nơi làm việc. Ngoài ra, LMS sử dụng Machine Learning còn có nhiều ứng dụng khác như:

- Tạo các mô đun đào tạo được cá nhân hóa cho mỗi người dùng bằng cách sử dụng dữ liệu thu được từ hệ thống nhân sự và LMS.

- Chỉ đạo các nhân viên tập trung vào các phần quan trọng của việc học và phát triển vì Machine Learning sẽ đảm nhiệm quá trình khai thác dữ liệu.

- Giúp các tổ chức phát triển các phương pháp đào tạo định hướng mục tiêu để cung cấp mô hình lấy người học làm trung tâm cho người dùng.

1. **Nhận dạng hình ảnh**

Nhận dạng hình ảnh là một lĩnh vực khác trong đó Machine Learning đóng vai trò quan trọng. Hầu hết các dữ liệu được lưu trữ trong các hệ thống thuộc sở hữu của công ty là trực quan, nên nhu cầu về một phương pháp tự động để xử lý chúng đã trở nên cần thiết. Machine Learning có thể hỗ trợ rất nhiều trong nhiều ứng dụng như nhận dạng khuôn mặt, nhận dạng hình ảnh thế giới thực…sau đó phân tích để tạo ra những thông tin quan trọng… Machine Learning có thể khai thác tiềm năng từ dữ liệu hình ảnh bằng cách phân loại chúng để dễ dàng truy cập.

1. **Công cụ đề xuất sản phẩm**

Đối với các doanh nghiệp thương mại điện tử, Machine Learning có chức năng chủ yếu là đề xuất các đề xuất sản phẩm [cho khách hàng dựa trên sở thích cá nhân hóa](https://www.fingent.com/blog/machine-learning-boosts-customer-experience) . Dữ liệu cụ thể của từng người dùng sẽ được xử lý bởi thuật toán của Machine Learning. Nó hoạt động theo phương pháp được gọi là lọc cộng tác nhằm đánh giá hồ sơ người dùng và lịch sử tìm kiếm sản phẩm của họ để dự đoán và từ đó đề xuất các sản phẩm phù hợp. Bên cạnh các doanh nghiệp thương mại điện tử, khuyến nghị sản phẩm cũng được sử dụng bởi nhiều doanh nghiệp khác, đặc biệt là những doanh nghiệp cung cấp để truyền phát phương tiện kỹ thuật số, chẳng hạn như Netflix.

1. **Chăm sóc khách hàng**

Các thuật toán Machine Learning ngày càng đang trở nên tốt hơn trong việc hỗ trợ khách hàng, nó đã được chứng minh là cải thiện sự hài lòng của họ và cung cấp những thông tin liên quan mà họ đang tìm kiếm. Chatbots sử dụng thuật toán tiên tiến có thể quản lý các truy vấn của khách hàng hiệu quả hơn con người. Việc đưa ra các quyết định dựa trên dữ liệu khiến cho việc phân loại khách hàng tốt hơn mang lại lợi ích trong việc đẩy nhanh quá trình làm việc trong khi cắt giảm chi phí hoạt động kinh doanh.

### 1.2.3 Tích hợp Machine Learning vào kinh doanh của doanh nghiệp

[Kinh doanh thông minh thường giống như Machine Learning](https://towardsdatascience.com/how-enterprise-software-is-getting-intelligent-through-machine-learning-4cafc2d7367e): chuyển đổi dữ liệu thành những những thông tin để quyết định hành động là điều cả hai có thể thực hiện. Phương pháp kinh doanh thông minh thường chỉ giúp quản lý các chỉ số hiệu suất chính (KPIs), phân tích dữ liệu trực quan và tăng cường hợp tác. Một số doanh nghiệp phụ thuộc vào một số hình thức công cụ kinh doanh thông minh để khai thác thông tin ý nghĩa từ dữ liệu mà họ sở hữu. Tuy nhiên, nó đã nhanh chóng được xem là một mô hình không phù hợp do tồn tại nhược điểm: các công cụ kinh doanh thông minh được xem là không đủ trong việc xác định những hiểu biết cốt lõi từ một tập dữ liệu. Để giải quyết những vấn đề này, các nhà cung cấp đã thử nghiệm tích hợp Machine Learning vào các công cụ kinh doanh thông minh để tạo ra ý nghĩa từ khối lượng dữ liệu lớn.

Machine Learning cải tiến nhanh chóng cách phân tích dữ liệu để khám phá những hiểu biết và mô hình ẩn. Nó tự động hóa các quy trình, nhanh chóng đánh giá thông tin từ đó cho phép các chủ doanh nghiệp hành động và đưa ra các quyết định liên quan. Hơn nữa, các dự đoán chỉ ra rằng, trong tương lai Machine Learning sẽ thay đổi các mô hình kinh doanh của doanh nghiệp thành một hệ thống mạnh mẽ và chính xác hơn.

Việc Machine Learning được ứng dụng ngày càng rộng rãi trên toàn thế giới đã mở ra một cách tiếp cận và chiến lược mới trong hoạt động kinh doanh. Các tập hợp dữ liệu lớn được tạo và lưu trữ trong doanh nghiệp sẽ trở nên ý nghĩa nếu mô hình Machine Learning được tích hợp vào các hệ thống. Điều này sẽ mở đường cho những hiểu biết và mô hình tốt hơn từ dữ liệu, giúp đưa ra các quyết định dựa trên dữ liệu quan trọng có thể thúc đẩy sự phát triển của một công ty.

Đối với các nhà phát triển phần mềm, Machine Learning cho phép họ xây dựng một cách tiếp cận mới mà họ có thể đưa vào trong quy trình xây dựng phần mềm cho các doanh nghiệp giúp tăng hiệu quả, tốc độ và khả năng cạnh tranh. Machine Learning có thể tích hợp vào một loạt các lĩnh vực điện toán như xử lý ngôn ngữ tự nhiên và nhận dạng hình ảnh, điều này đặc biệt có lợi trong việc tạo ra ý nghĩa từ khối lượng dữ liệu khổng lồ. Cách tiếp cận như vậy chắc chắn sẽ thúc đẩy sự đổi mới và cải thiện các sản phẩm và dịch vụ được cung cấp cùng với trải nghiệm của khách hàng.

# Chương 2. NGÔN NGỮ PYTHON, THƯ VIỆN SCIKIT-LEARN VÀ CÁC THUẬT TOÁN LIÊN QUAN

## 2.1 Ngôn ngữ lập trình Python

Python là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) bậc cao cho các mục đích lập trình đa năng, do [Guido van Rossum](https://vi.wikipedia.org/w/index.php?title=Guido_van_Rossum&action=edit&redlink=1) tạo ra và lần đầu ra mắt vào năm [1991](https://vi.wikipedia.org/wiki/1990). Python được thiết kế với ưu điểm mạnh là dễ đọc, dễ học và dễ nhớ. Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình. Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu.

Python hoàn toàn [tạo kiểu động](https://vi.wikipedia.org/w/index.php?title=T%E1%BA%A1o_ki%E1%BB%83u_%C4%91%E1%BB%99ng&action=edit&redlink=1) và dùng cơ chế [cấp phát bộ nhớ tự động](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD_b%E1%BB%99_nh%E1%BB%9B), do vậy nó tương tự như [Perl](https://vi.wikipedia.org/wiki/Perl), [Ruby](https://vi.wikipedia.org/wiki/Ruby_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), [Scheme](https://vi.wikipedia.org/wiki/Scheme), [Smalltalk](https://vi.wikipedia.org/wiki/Smalltalk), và [Tcl](https://vi.wikipedia.org/wiki/Tcl). Python được phát triển trong một dự án mã mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý. Ban đầu, Python được phát triển để chạy trên nền [Unix](https://vi.wikipedia.org/wiki/Unix). Nhưng rồi theo thời gian, Python dần mở rộng sang mọi [hệ điều hành](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh) từ [MS-DOS](https://vi.wikipedia.org/wiki/MS-DOS) đến [Mac OS](https://vi.wikipedia.org/wiki/Mac_OS),

S/2, [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux) và [các hệ điều hành khác thuộc họ Unix](https://vi.wikipedia.org/wiki/T%C6%B0%C6%A1ng_t%E1%BB%B1_Unix). Mặc dù sự phát triển của Python có sự đóng góp của rất nhiều cá nhân, nhưng Guido van Rossum hiện nay vẫn là tác giả chủ yếu của Python. Ông giữ vai trò chủ chốt trong việc quyết định hướng phát triển của Python.

### 2.1.1 Ưu, nhược điểm của ngôn ngữ Python

Đơn giản, dễ học: Python có cú pháp rất đơn giản, rõ ràng. Nó dễ đọc và viết hơn rất nhiều khi so sánh với những ngôn ngữ lập trình khác như C++, Java, C#. Python làm cho việc lập trình trở nên thú vị, cho phép lập trình viên tập trung vào những giải pháp chứ không phải cú pháp.

Miễn phí, mã nguồn mở: lập trình viên có thể tự do sử dụng và phân phối Python, thậm chí là dùng cho mục đích thương mại. Vì là mã nguồn mở, lập trình viên không những có thể sử dụng các phần mềm, chương trình được viết trong Python mà còn có thể thay đổi mã nguồn của nó. Python có một cộng đồng rộng lớn, không ngừng cải thiện nó mỗi lần cập nhật.

Khả năng di chuyển: Các chương trình Python có thể di chuyển từ nền tảng này sang nền tảng khác và chạy nó mà không có bất kỳ thay đổi nào. Nó chạy liền mạch trên hầu hết tất cả các nền tảng như Windows, macOS, Linux.

Khả năng mở rộng và có thể nhúng: Giả sử một ứng dụng đòi hỏi sự phức tạp rất lớn, lập trình viên có thể dễ dàng kết hợp các phần code bằng C, C++ và những ngôn ngữ khác (có thể gọi được từ C) vào code Python. Điều này sẽ cung cấp cho ứng dụng của lập trình viên những tính năng tốt hơn cũng như khả năng scripting mà những ngôn ngữ lập trình khác khó có thể làm được.

Ngôn ngữ thông dịch cấp cao: Không giống như C/C++, với Python, bạn không phải lo lắng những nhiệm vụ khó khăn như quản lý bộ nhớ, dọn dẹp những dữ liệu vô nghĩa,... Khi chạy code Python, nó sẽ tự động chuyển đổi code sang ngôn ngữ máy tính có thể hiểu. Lập trình viên không cần lo lắng về bất kỳ hoạt động ở cấp thấp nào.

Thư viện tiêu chuẩn lớn để giải quyết những tác vụ phổ biến: Python có một số lượng lớn thư viện tiêu chuẩn giúp cho công việc lập trình của lập trình viên trở nên dễ thở hơn rất nhiều, đơn giản vì không phải tự viết tất cả code. Những thư viện này được kiểm tra kỹ lưỡng và được sử dụng bởi hàng trăm người. Vì vậy, có thể chắc chắn rằng nó sẽ không làm hỏng code hay ứng dụng của lập trình viên.

Hướng đối tượng: Mọi thứ trong Python đều là hướng đối tượng. Lập trình hướng đối tượng (OOP) giúp giải quyết những vấn đề phức tạp một cách trực quan. Với OOP, lập trình viên có thể phân chia những vấn đề phức tạp thành những tập nhỏ hơn bằng cách tạo ra các đối tượng.

Tốc độ xử lý khá nhanh: Với tốc độ xử lý nhanh, Python có thể tạo ra những chương trình từ những script siêu nhỏ tới những phần mềm cực lớn như Biender 3D.

GUI Programming: Giúp cho việc thực hiện ảnh minh họa di động một cách tự nhiên và sống động hơn.

Với những tính năng kể trên, Python mang nhiều ưu điểm để những người mới bắt đầu học ngôn ngữ lập trình hoặc những lập trình viên lâu năm lựa chọn. Tuy nhiên Python cũng có một số nhược điểm, có thể kể đến:

Python không có các thuộc tính như: protected, private hay public, không có vòng lặp do…while và switch….case.

Python mặc dù nhanh hơn so với PHP, nhưng lại không nhanh hơn so với C++, Java.

### 2.1.2 Python trong Machine Learning

Ngôn ngữ lập trình Python có rất nhiều thư viện và framework lớn thuận lợi cho việc viết code và phát triển khoa học máy tính. Python vốn là ngôn ngữ nổi tiếng về sự đơn giản không cầu kỳ, code dễ học, dễ đọc, cú pháp logic và ngắn gọn, còn Machine Learning liên quan đến các thuật toán cực kỳ phức tạp và quy trình làm việc nhiều giai đoạn nên ở đây, sự logic ngắn gọn và dễ dàng của Python đóng vai trò quan trọng trong việc tiết kiệm thời gian của các nhà phát triển. Những lý do giúp Python được sử dụng phổ biến nhất trong các dự án học máy bao gồm:

* 1. **Tính đơn giản và nhất quán**

Trong khi các thuật toán phức tạp và quy trình làm việc linh hoạt của Học máy dễ làm hệ thống trở nên phức tạp thì, tính đơn giản của Python là giải pháp cho phép các lập trình viên viết ra các hệ thống đáng tin cậy hơn. Các lập trình viên sẽ tập trung được thời gian, trí lực vào giải quyết bài toán của học máy thay vì gặp rắc rối với kỹ thuật của ngôn ngữ.

Python là ngôn ngữ có mục đích chung, nó có thể thực hiện một tập hợp các tác vụ Học máy phức tạp và cho phép bạn xây dựng các nguyên mẫu nhanh chóng, kiểm thử sản phẩm của mình cho mục đích học máy tốt hơn.

* 1. **Cho phép truy cập vào các thư viện và framework cho học máy.**

Việc thực hiện các thuật toán của Machine Learning có thể rất khó và mất nhiều thời gian, nên một môi trường có cấu trúc và được thử nghiệm tốt là rất quan trọng cho các lập trình viên.

Để giảm thời gian phát triển dự án, các lập trình viên chuyển sang một số Framework và Thư viện của Python. Python có một kho công nghệ phong phú bao gồm rất nhiều thư viện cho trí tuệ nhân tạo và học máy. Một số thư viện, framework phổ biến có thể kể đến như:

* Keras, TensorFlow, and Scikit-learn cho Học máy
* NumPy cho phân tích dữ liệu và tính toán khoa học hiệu năng cao
* SciPy cho advanced computing
* Pandas để phân tích dữ liệu mục đích chung
* Seaborn cho trực quan hóa dữ liệu (Data Visualization)
  1. **Độc lập với nền tảng**

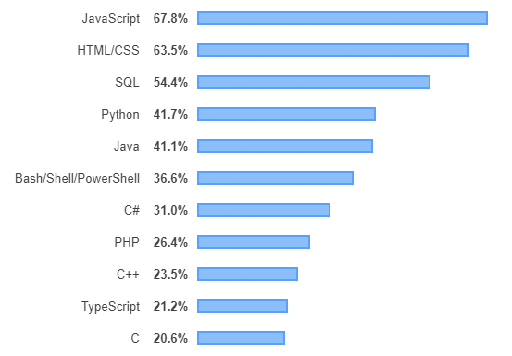
Độc lập với nền tảng tức là một ngôn ngữ lập trình hoặc framework cho phép các lập trình viên triển khai mọi thứ trên một máy và sử dụng chúng trên một máy khác mà không có bất kỳ thay đổi nào (hoặc chỉ thay đổi tối thiểu).

Độc lập với nền tảng cũng là một lý do chính khiến Python trở nên phổ biến.Python được hỗ trợ bởi nhiều nền tảng bao gồm Linux, Windows và macOS. Code Python có thể được sử dụng để tạo các chương trình thực thi độc lập cho hầu hết các hệ điều hành phổ biến, điều này có nghĩa là phần mềm Python có thể dễ dàng phân phối và sử dụng trên các hệ điều hành đó mà không cần trình thông dịch Python.

Các lập trình viên thường sử dụng các dịch vụ như Google hoặc Amazon cho nhu cầu điện toán của họ. Tuy nhiên, chúng ta vẫn có thể tìm thấy các công ty và nhà khoa học dữ liệu sử dụng máy của riêng họ với Bộ xử lý đồ họa (GPU) mạnh mẽ để đào tạo các mô hình Machine Learning của họ. Và thực tế là Python là nền tảng độc lập làm cho việc đào tạo này rẻ hơn và dễ dàng hơn rất nhiều.

* 1. **Cộng đồng rộng lớn**

Trong Khảo sát Lập trình viên năm 2019 của Stack Overflow, Python đứng thứ 4 trong top 10 ngôn ngữ lập trình phổ biến nhất, điều này có nghĩa là chúng ta có thể tìm thấy các tài liệu hướng dẫn Python một cách dễ dàng.



Hình 2 1 Mức độ phổ biến của các ngôn ngữ lập trình theo Stack Overflow



Hình 2 2 Tỷ lệ sử dụng ngôn ngữ Python trong các lĩnh vực

Hiện nay, trên các kho lưu trữ trực tuyến cũng chứa hơn 140.000 package python. Các package Python dành cho khoa học như Numpy, Scipym, Sk-Learn và Matplotlib phục vụ cho Học máy và giúp các lập trình viên phát hiện các mẫu trong các tập dữ liệu lớn.

## 2.2 Thư viện Sklearn và các thuật toán máyhọcđược hỗ trợ

### 2.2.1 Thư viện Sklearn

**a. Tổng quan về thư viện Sklearn**

Sklearn là thư viện mạnh mẽ nhất dành cho các thuật toán học máy được viết trên ngôn ngữ Python. Thư viện cung cấp một tập các công cụ xử lý các bài toán machine learning và statistical modeling gồm: classification, regression, clustering và dimensionality reduction.Thư viện được cấp phép bản quyền chuẩn FreeBDS và chạy được trên nhiều nền tảng. Sklearn được sử dụng nhiều trong học tập và nghiên cứu

Để cài đặt Sklearn trước tiên phải cài thư viện SciPy (Scientific Python). Những thành phần gồm:

* [Numpy](https://codelearn.io/sharing/tim-hieu-thu-vien-numpy-trong-python): Gói thư viện xử lý dãy số và ma trận nhiều chiều
* SciPy: Gói các hàm tính toán logic khoa học
* [Matplotlib](https://codelearn.io/sharing/ve-bieu-do-voi-thu-vien-matplotlib-p1): Biểu diễn dữ liệu dưới dạng đồ thị 2 chiều, 3 chiều
* IPython: Notebook dùng để tương tác trực quan với Python
* SymPy: Gói thư viện các kí tự toán học
* [Pandas](https://codelearn.io/sharing/xu-ly-du-lieu-voi-pandas-trong-python): Xử lý, phân tích dữ liệu dưới dạng bảng

Những thư viện mở rộng của SciPy thường được đặt tên dạng SciKits. Như thư viện này là gói các lớp, hàm sử dụng trong thuật toán học máy thì được đặt tên là scikit-learn.

Sklearn hỗ trợ mạnh mẽ trong việc xây dựng các sản phẩm. Nghĩa là thư viện này tập trung sâu trong việc xây dựng các yếu tố: dễ sử dụng, dễ code, dễ tham khảo, dễ làm việc, hiệu quả cao.Mặc dù được viết cho Python nhưng thực ra các thư viện nền tảng của Sklearn lại được viết dưới các thư viện của C để tăng hiệu suất làm việc. Ví dụ như: Numpy(Tính toán ma trận), LAPACK, LibSVM và Cython.

**b. Xử lý dữ liệu trước khi sử dụng thư viện Sklearn**

Tiền xử lý dữ liệu là một bước rất quan trọng trong việc giải quyết bất kỳ vấn đề nào trong lĩnh vực học máy. Hầu hết các bộ dữ liệu được sử dụng trong các vấn đề liên quan đến học máy cần được xử lý, làm sạch và biến đổi trước khi một thuật toán học máy có thể được huấn luyện trên những bộ dữ liệu này. Các kỹ thuật tiền xử lý dữ liệu phổ biến hiện nay bao gồm: xử lý dữ liệu bị khuyết, mã hóa các biến nhóm, chuẩn hóa dữ liệu, co giãn dữ liệu… Những kỹ thuật này tương đối dễ hiểu nhưng sẽ có nhiều vấn đề phát sinh khi chúng ta áp dụng vào các dữ liệu thực tế.

Bởi lẽ các bộ dữ liệu ứng với các bài toán trong thực tế rất khác nhau và mỗi bài toàn thì đối mặt với những thách thức khác nhau về mặt dữ liệu. Trong đề tài nghiên cứu cụ thể này, với tệp dữ liệu đã thu thập được từ doanh nghiệp bán phụ tùng ô tô, chúng tôi nghiên cứu kỹ thuật chuẩn hóa dữ liệu với LabelEncoder() và sử dụng train\_test\_split()để chia tệp dữ liệu thành các mẫu sử dụng

LabelEncoder(): Đề cập đến việc chuyển đổi các nhãn thành dạng số để chuyển đổi nó thành dạng máy có thể đọc được. Sau đó, các thuật toán học máy có thể quyết định theo cách tốt hơn về cách các nhãn đó phải được vận hành.

Train\_test\_split là một chức năng trong lựa chọn mô hình Sklearn để chia mảng dữ liệu thành các tập dữ liệu huấn luyện và dữ liệu thử nghiệm. Với chức năng này, chúng ta không cần phải chia tập dữ liệu theo cách thủ công. Khai báo chức năng này như sau:

*train\_test\_split(x, y, train\_size=0.\*,test\_size=0.\*, random\_state=\*)*

x, y: Tham số đầu tiên là tập dữ liệu bạn đang chọn sử dụng.

train\_size: Tham số này đặt kích thước của tập dữ liệu đào tạo . Có ba tùy chọn:, None là mặc định, Int yêu cầu số lượng mẫu chính xác, và float trong khoảng từ 0,1 đến 1,0.

test\_size: Tham số này chỉ định kích thước của tập dữ liệu thử nghiệm . Trạng thái mặc định phù hợp với kích thước đào tạo. Nó sẽ được đặt thành 0,25 nếu kích thước đào tạo được đặt thành mặc định.

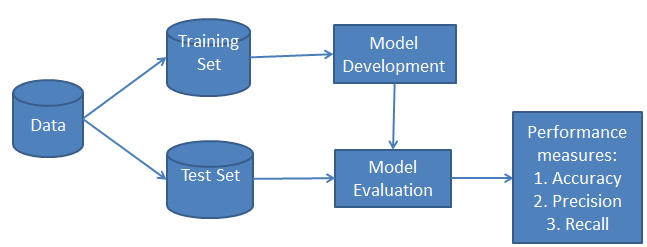
random\_state: Chế độ mặc định thực hiện phân chia ngẫu nhiên bằng cách sử dụng np.random. Ngoài ra, có thể thêm một số nguyên bằng cách sử dụng một số chính xác.

### 2.2.2 Một số thuật toán được hỗ trợ từ thư viện SkLearn

**a. Naive-Bayes**

Naive Bayes là một thuật toán học máy xác suất có thể được sử dụng trong nhiều nhiệm vụ phân loại. Các ứng dụng điển hình bao gồm lọc thư rác, phân loại tài liệu, dự đoán tình cảm... Nó dựa trên các tác phẩm của Rev. Thomas Bayes

Vì sự đơn giản và hiệu suất tốt Naive-Bayes trở thành một kỹ thuật phổ biến, có một lợi thế đáng kể với NB. Nó là một mô hình xác suất, nên thuật toán có thể được mã hóa dễ dàng và các dự đoán được thực hiện nhanh chóng. Thời gian thực nhanh chóng. Do đó, nó có thể dễ dàng mở rộng và là thuật toán được lựa chọn theo điều kiện cho các ứng dụng trong thế giới thực (ứng dụng) được yêu cầu để đáp ứng yêu cầu của người dùng ngay lập tức.



Hình 2 3 Sơ đồ hoạt động Naive-Bayes

Bộ phân lớp Naive bayes hay bộ phân lớp Bayes (simple byes classifier) hoạt động như sau:

Gọi D là tập dữ liệu huấn luyện, trong đó mỗi phần tử dữ liệu X được biểu diễn bằng một vector chứa n giá trị thuộc tính A1, A2,...,An = {x1,x2,...,xn}

Giả sử có m lớp C1, C2,..,Cm. Cho một phần tử dữ liệu X, bộ phân lớp sẽ gán nhãn cho X là lớp có xác suất hậu nghiệm lớn nhất. Cụ thể, bộ phân lớp Bayes sẽ dự đoán X thuộc vào lớp Ci nếu và chỉ nếu:

P(Ci|X) > P(Cj|X) (1<= i, j <=m, i != j)

Giá trị này sẽ tính dựa trên định lý Bayes.

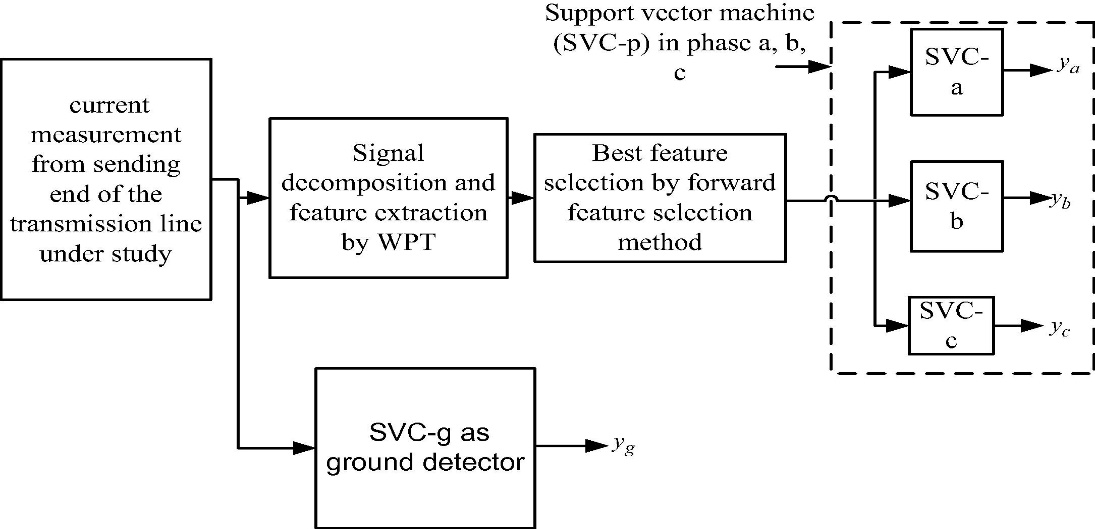
Để tìm xác suất lớn nhất, ta nhận thấy các giá trị P(X) là giống nhau với mọi lớp nên không cần tính. Do đó ta chỉ cần tìm giá trị lớn nhất của P(X|Ci) \* P(Ci). Chú ý rằng P(Ci) được ước lượng bằng |Di|/|D|, trong đó Di là tập các phần tử dữ liệu thuộc lớp Ci. Nếu xác suất tiền nghiệm P(Ci) cũng không xác định được thì ta coi chúng bằng nhau P(C1) = P(C2) = ... = P(Cm), khi đó ta chỉ cần tìm giá trị P(X|Ci) lớn nhất.

Khi số lượng các thuộc tính mô tả dữ liệu là lớn thì chi phí tính toàn P(X|Ci) là rất lớn, dó đó có thể giảm độ phức tạp của thuật toán Naive Bayes giả thiết các thuộc tính độc lập nhau. Khi đó ta có thể tính:

P(X|Ci) = P(x1|Ci)...P(xn|Ci)

1. **LinearSVC**

Mục tiêu của Linear SVC (Support Vector Classifier) trả về dữ liệu "phù hợp nhất" để phân chia hoặc phân loại dữ liệu. Từ đó, sau khi nhận được dữ liệu, bạn có thể cung cấp một số tính năng cho bộ phân loại của mình để xem lớp dự đoán là gì. Điều này làm cho thuật toán cụ thể này khá phù hợp với mục đích sử dụng của chúng tôi, mặc dù bạn có thể sử dụng thuật toán này cho nhiều trường hợp.

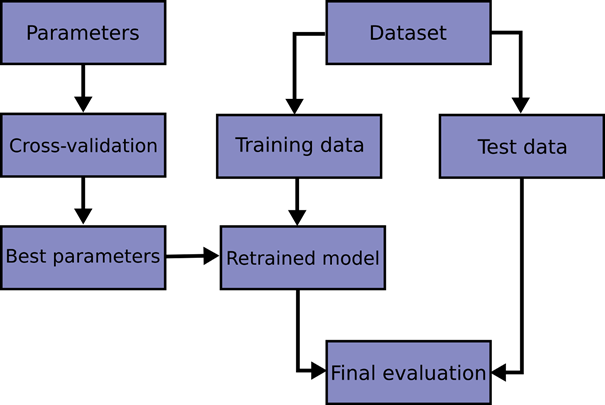
******

Hình 2 4 Sơ đồ hoạt động LinearSVC.

**c. K-Neighbors Classifier**

K-Nearest Neighbors là một trong những thuật toán phân loại cơ bản nhưng thiết yếu nhất trong Học máy. Nó thuộc về miền học tập có giám sát và tìm thấy ứng dụng mạnh mẽ trong nhận dạng mẫu, khai thác dữ liệu.

Thuật toán KNN cũng có thể được sử dụng cho các bài toán hồi quy. Sự khác biệt duy nhất sẽ là sử dụng giá trị trung bình của các giá trị gần nhất thay vì bỏ phiếu từ các giá trịgần nhất.

******

Hình 2 5 Sơ đồ hoạt động K-Neightbors Classifier.

**d. Support Vector Machines**

Là một tập hợp các phương pháp học có giám sát được sử dụng để[phân loại](https://scikit-learn.org/stable/modules/svm.html#svm-classification) , [hồi quy](https://scikit-learn.org/stable/modules/svm.html#svm-regression) và [phát hiện ngoại lệ](https://scikit-learn.org/stable/modules/svm.html#svm-outlier-detection) .Ưu điểm support vector machines là:

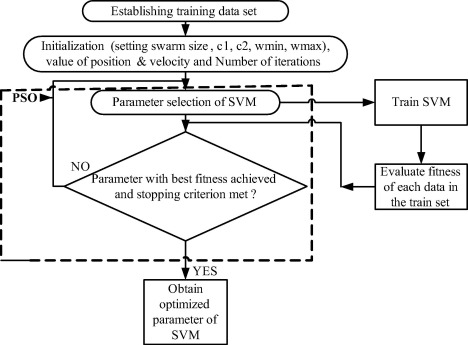
- Sử dụng một tập hợp con của các điểm huấn luyện trong hàm quyết định (được gọi là vectơ hỗ trợ), vì vậy nó cũng hiệu quả về bộ nhớ.

- Đa năng: các [chức năng Kernel](https://scikit-learn.org/stable/modules/svm.html#svm-kernels) khác nhau có thể được chỉ định cho chức năng quyết định. Các nhân chung được cung cấp, cũng có thể chỉ định các nhân tùy chỉnh.

- Xử lý trên không gian số chiều cao: SVM là một công cụ tính toán hiệu quả trong không gian chiều cao, trong đó đặc biệt áp dụng cho các bài toán phân loại văn bản và phân tích quan điểm nơi chiều có thể cực kỳ lớn.

- Tiết kiệm bộ nhớ: Do chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới nên chỉ có những điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết định.

- Tính linh hoạt - phân lớp thường là phi tuyến tính. Khả năng áp dụng Kernel mới cho phép linh động giữa các phương pháp tuyến tính và phi tuyến tính từ đó khiến cho hiệu suất phân loại lớn hơn.

******

Hình 2 6 Sơ đồ hoạt động Support Vector Machines.

1. ***Random Forest***

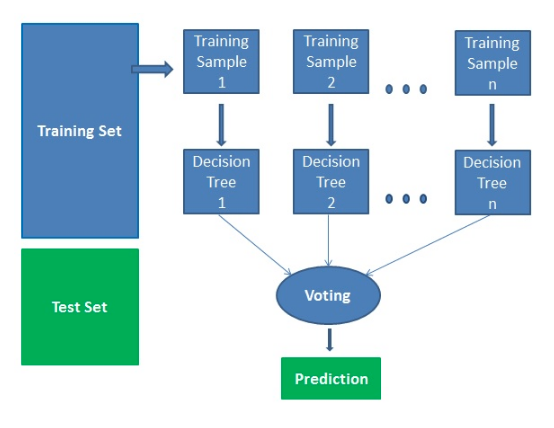
Random Forest là một [thuật toán học máy](https://builtin.com/data-science/introduction-to-machine-learning) linh hoạt, dễ sử dụng , tạo ra một kết quả tuyệt vời ngay cả khi không điều chỉnh siêu tham số. Nó cũng là một trong những thuật toán được sử dụng nhiều nhất, vì tính đơn giản và đa dạng của nó (nó có thể được sử dụng cho cả nhiệm vụ phân loại và hồi quy). Thuật toán hoạt động theo bốn bước:

- Chọn các mẫu ngẫu nhiên từ tập dữ liệu đã cho.

- Thiết lập cây quyết định cho từng mẫu và nhận kết quả dự đoán từ mỗi quyết định cây.

- Hãy bỏ phiếu cho mỗi kết quả dự đoán.

- Chọn kết quả được dự đoán nhiều nhất là dự đoán cuối cùng.

******

Hình 2 7. Sơ đồ hoạt động Random Forest.

Các tính năng quan trọng:

- Random forests cũng cung cấp một chỉ số lựa chọn tính năng tốt. Scikit-learn cung cấp thêm một biến với mô hình, cho thấy tầm quan trọng hoặc đóng góp tương đối của từng tính năng trong dự đoán. Nó tự động tính toán điểm liên quan của từng tính năng trong giai đoạn đào tạo. Sau đó, nó cân đối mức độ liên quan xuống sao cho tổng của tất cả các điểm là 1.

- Điểm số này sẽ giúp bạn chọn các tính năng quan trọng nhất và thả các tính năng quan trọng nhất để xây dựng mô hình.

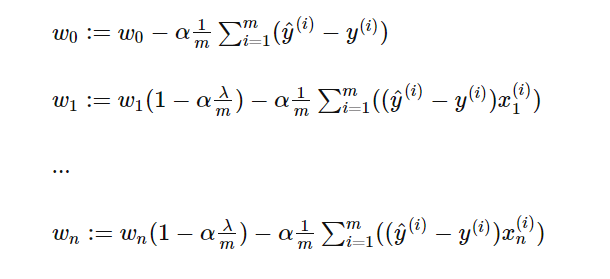
- Random forests sử dụng tầm quan trọng của gini hoặc giảm tạp chất trung bình (MDI) để tính toán tầm quan trọng của từng tính năng. Gini tầm quan trọng còn được gọi là tổng giảm trong tạp chất nút. Đây là mức độ phù hợp hoặc độ chính xác của mô hình giảm khi bạn thả biến. Độ lớn càng lớn thì biến số càng có ý nghĩa. Ở đây, giảm trung bình là một tham số quan trọng cho việc lựa chọn biến. Chỉ số Gini có thể mô tả sức mạnh giải thích tổng thể của các biến. Random Forests và cây quyết định Random Forests là một tập hợp của nhiều cây quyết định. Cây quyết định sâu có thể bị ảnh hưởng quá mức, nhưng Random forests ngăn cản việc lấp đầy bằng cách tạo cây trên các tập con ngẫu nhiên. Cây quyết định nhanh hơn tính toán. Random forests khó giải thích, trong khi cây quyết định có thể diễn giải dễ dàng và có thể chuyển đổi thành quy tắc.

1. ***Logistic Regression***

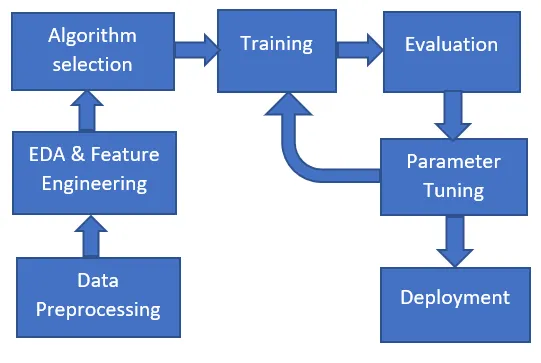
Phương pháp thống kê hồi quy logistic được sử dụng cho phân tích tập dữ liệu và tạo ra kết quả nhị phân. Một hoặc nhiều biến tự trị hơn có thể bao gồm tập dữ liệu. Kết quả được xác đinh bởi các biến này lưỡng tính trong tự nhiên. Có nghĩa là có hai kết quả là khả thi. Thuật toán là một loại hồi quy cụ thể và nó được sử dụng theo cách tốt nhất để dự đoán nhị phân và phân loại đầu ra. Các bước triển khai như sau:

Bước 1: Chọn một điểm bất kỳ (, ,…,) và một giá trị learning rate α.

Bước 2: Liên tiếp lặp lại các phép biến đổi



Bước 3: Thuật toán dừng lại khi J(w) thay đổi rất nhỏ hoặc trị tuyệt đối các đạo hàm riêng rất nhỏ. Nếu thuật toán không thể kết thúc thì chọn giá trị α nhỏ hơn rồi quay lại bước 2.



Hình 2 8 Sơ đồ hoạt động Logistic Resgression.

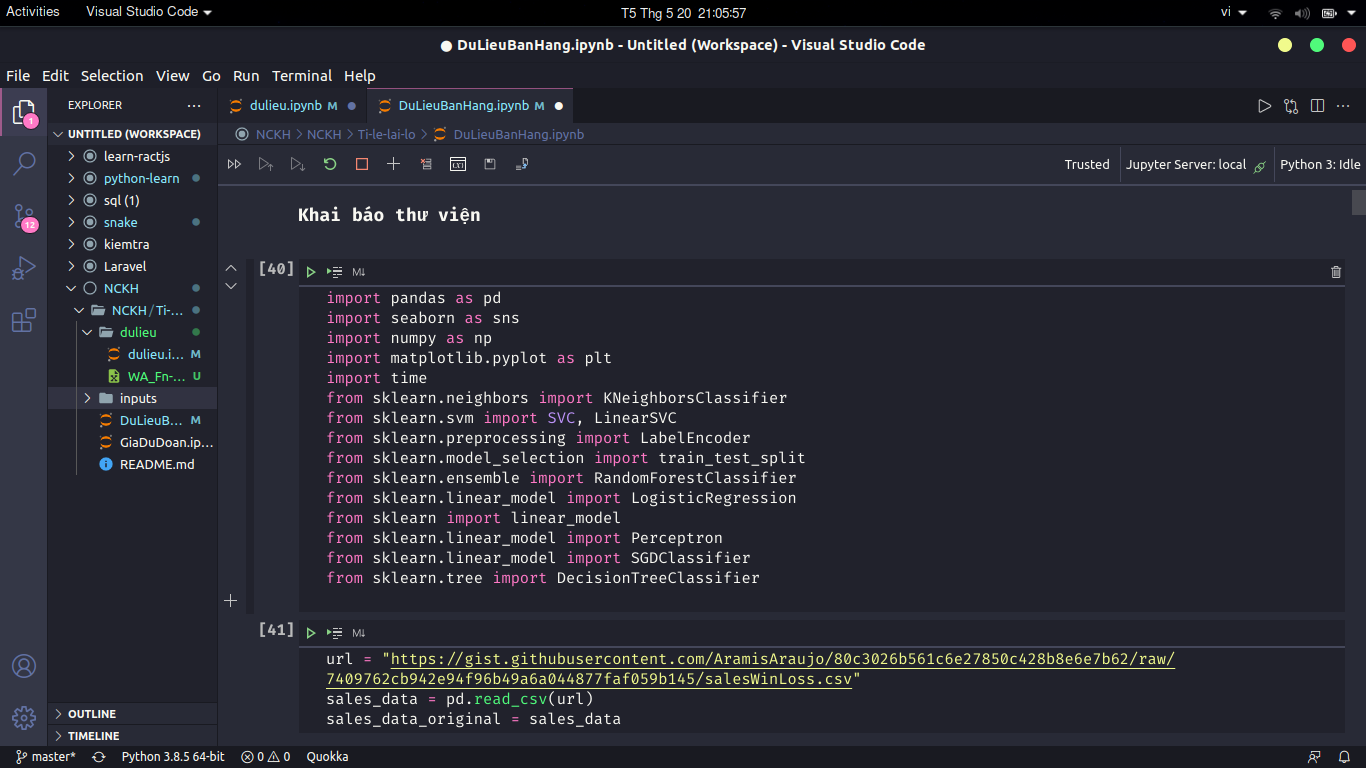
Logistic Regression phù hợp với loại dữ liệu mà hai class là gần với linearly separable. Một kiểu dữ liệu mà Logistic Regression không làm việc được là dữ liệu mà một class chứa các điểm nằm trong 1 vòng tròn, class kia chứa các điểm bên ngoài đường tròn đó. Kiểu dữ liệu này được gọi là phi tuyến (non-linear). Hạn chế của Logistic Regression là nó yêu cầu các điểm dữ liệu được tạo ra một cách *độc lập* với nhau. Trên thực tế, các điểm dữ liệu có thể bị *ảnh hưởng* bởi nhau. Mặc dù vậy, để cho đơn giản, khi xây dựng mô hình, người ta vẫn thường giả sử các điểm dữ liệu là độc lập với nhau.

# Chương 3. PHƯƠNG PHÁP THỰC HIỆN VÀ ĐÁNH GIÁ KẾT QUẢ

Với mục đích nghiên cứu khả năng sử dụng thư viện Sklearn để dự báo doanh thu của hoạt động kinh doanh, nên chúng tôi đã sử dụng tập dữ liệu bán hàng “won-loss” sẵn có trên website *https://www.ibm.com/watson/products-services*. Dữ liệu này chứa thông tin kinh doanh phụ tùng ô tô của doanh nghiệp

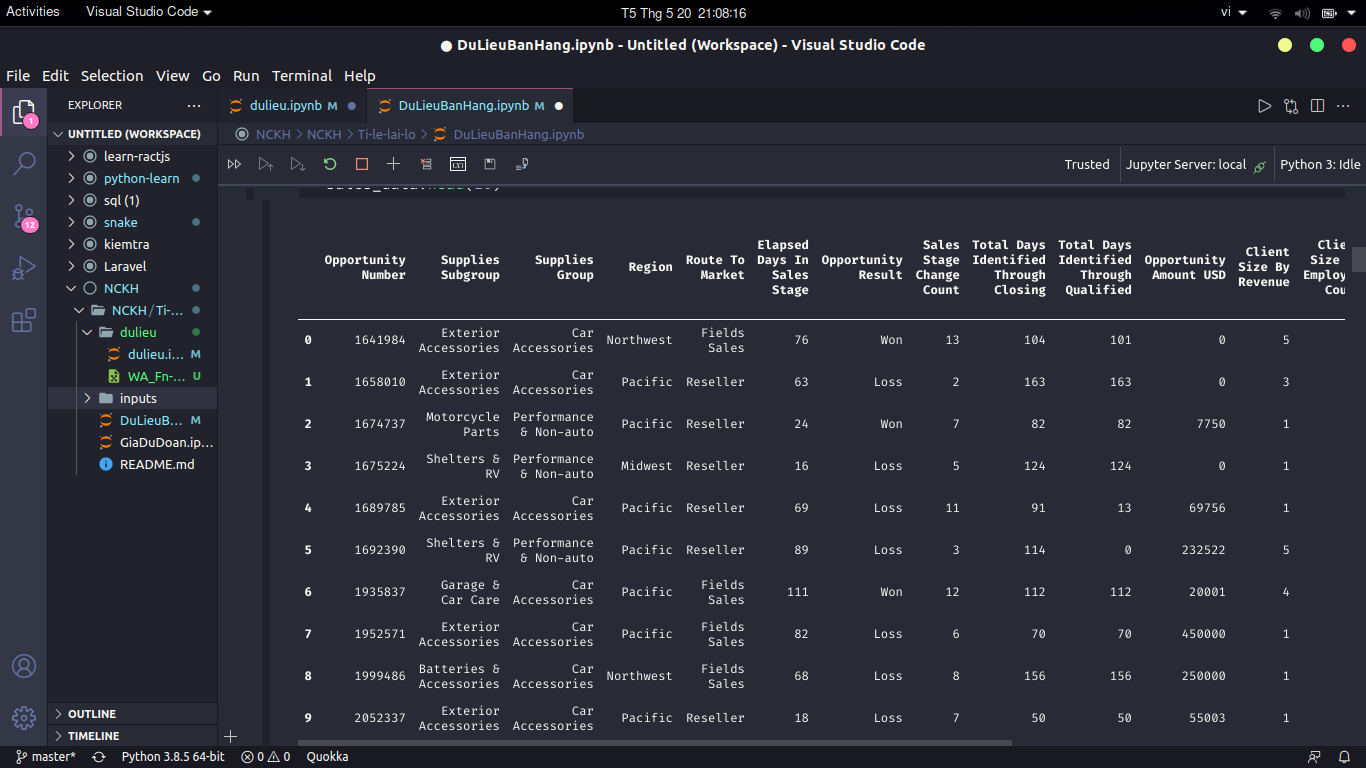
## 3.1 Khai báo thư viện và dữ liệu cần phân tích

Bước đầu tiên là import các module và thư viện cần thiết để phân tích tệp dữ liệu. Tiếp theo sử dụng read\_csv() phương thức được cung cấp bởi module pandas để đọc tệp csv chứa các dữ liệu được phân tách bằng dấu phấy và chuyển đổi thành pandas dataFame



Hình 3 1 Khai báo thư viện.

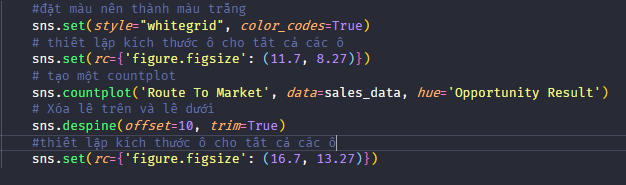
Sau khi import dữ liệu và chuyển đổi thành dataFame, hiển thị 1 vài bản ghi từ dataFame sử dụng head()



Hình 3 2 Hình ảnh bản ghi của dữ liệu mẫu.

## 3.2 Trực quan hóa dữ liệu

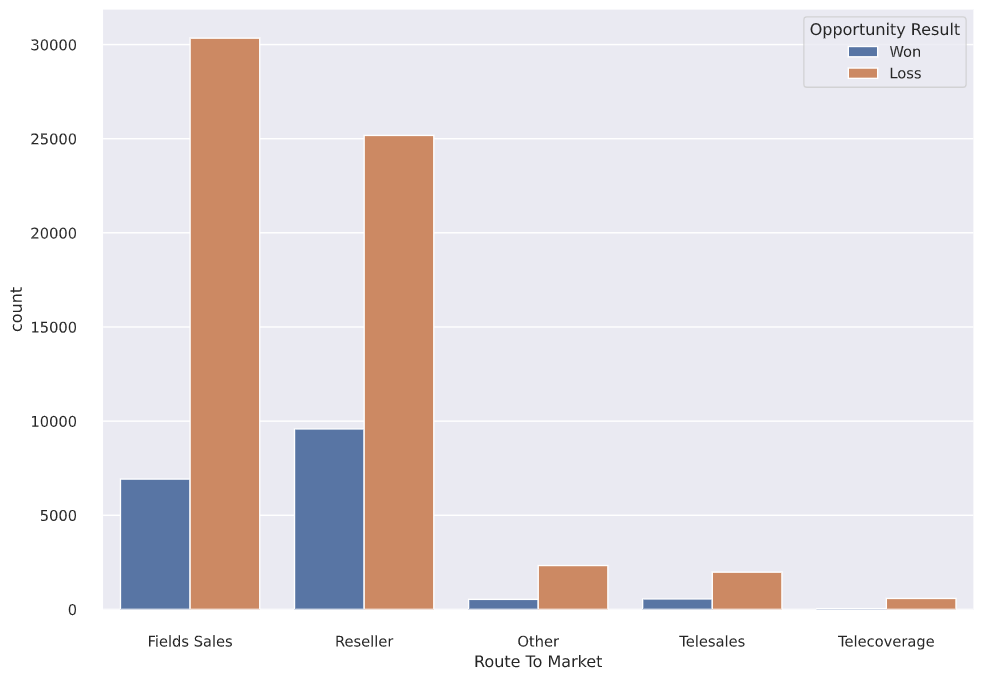
Để thực hiện trực quan hóa dữ liệu, chúng sử dụng Seaborn.



Hình 3 3 Mã lệnh trực quan hóa dữ liệu sử dụng Seaborn.

Các set() giúp thiết lập các thuộc tính khác nhau như ‘style’,‘color’ Như sử dụng sns.set(style="whitegrid", color\_codes=True) đoạn mã đặt nền là màu trăng. Sau đó đặt kích thước sns.set(rc={'figure.figsize':(11.7,8.27)}) xác định kích thước hình vẽ là 11.7px và 8.27px.

Tiếp theo tạo tạo các cột bằng countplot ('Route To Market',data=sales\_data,hue = 'Opportunity Result'). Ở đây trong đối số đầu tiên đã xác định trục X là cột ‘Route To Market’(thị trường) từ tập dữ liệu. Đối số thứ 2 là nguồn dữ liệu. Đối số thứ 3 là màu của ô vạch đã gán cho xanh lam cho ‘Won’ và nâu đỏ cho ‘Loss’.



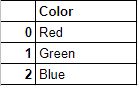
Hình 3 4 Trực quan hóa dữ liệu với biểu đồ cột.

## 3.3 Tiền xử lý dữ liệu

Khi đã phân tích và hiểu rõ dữ liệu của mình trông như thế nào, bây giờ tiến tới việc chuẩn bị nó để xây dựng các mô hình dự đoán bằng cách sử dụng scikit-learning

Trong lần phân tích ban đầu, thấy rằng hầu hết các cột trong tập dữ liệu là chuỗi, nhưng các thuật toán trong scikit-learning chỉ hiểu dữ liệu số. Nhưng trong thư viện scikit-learning cung cấp cho chúng ta nhiều phương pháp để chuyển đổi dữ liệu chuỗi thành dữ liệu số. Một trong những phương pháp đó là LabelEncoder().

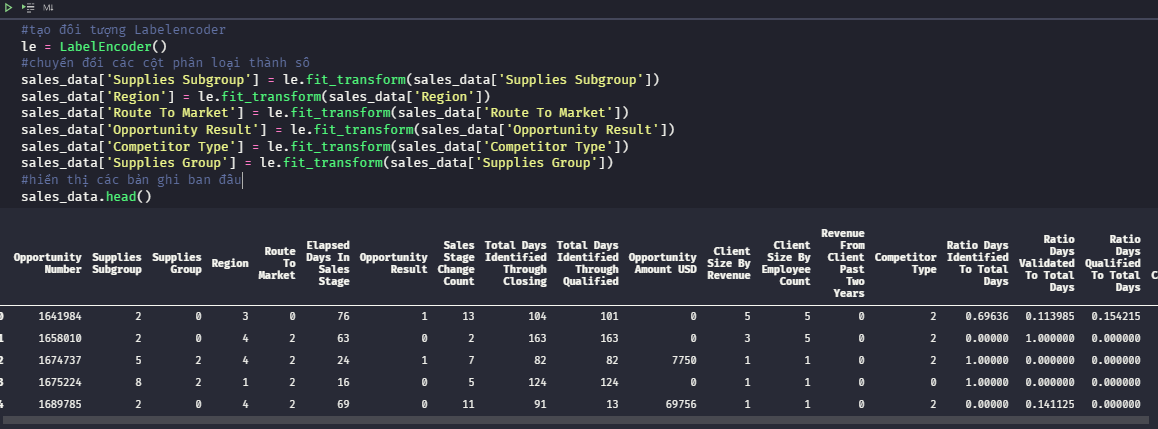
Phương pháp LabelEncoder() giúp chuyển đổi các nhãn phân loại trong tập dữ liệu như ‘won’ và ‘loss’ thành các số ví dụ như:



Kết quả ta thấy: Màu red chuyển 0, Green chuyển về 1, Blue chuyển về 2.

Sau khi phân tích song bắt đầu quá trình phân tích thực tế: Sử dụng fit\_transform() phương pháp được cung cấp bởi LabelEncoder() để mã hóa các tiêu đề trong cột phân loại, ví dụ như ‘Route To Market’ và chuyển đổi chúng thành các số tương ứng.

Mã hóa dữ liệu:



Hình 3 5 Bản ghi dữ liệu khi đã sử dụng phương pháp LabelEncoder.

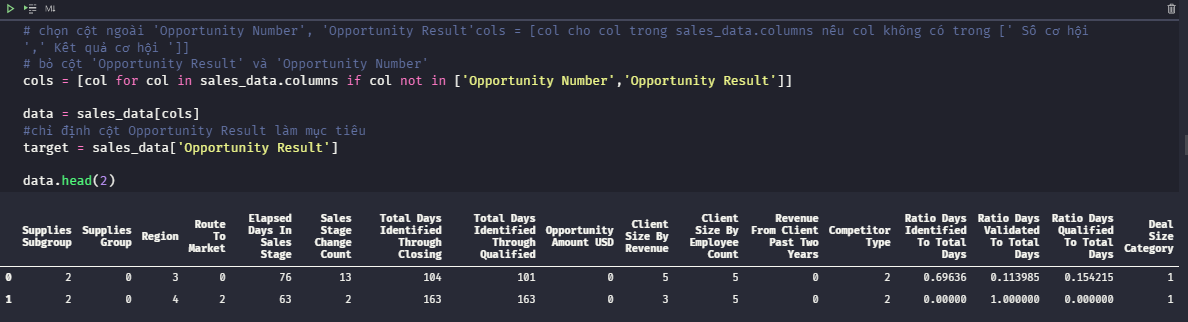
Đầu tiên tạo 1 đối tượng le của LabelEncoder(). Trong các dòng tiếp theo sử dụng fit\_transform() chức năng được cung cấp bởi LBE và chuyển đổi các nhãn phân loại của các cột khác nhau như ‘Region’, ‘Route To Market’ thành các số.

## 3.4 Training and Test

Thuật toán Machine Learning cần được đào tạo trên 1 tập dữ liệu để tìm hiểu mối quan hệ giữa các tính năng khác nhau và cách các tính năng này ảnh hưởng đển biến mục tiêu. Cần chia toàn bộ dữ liệu thành hai tập hợp:

Một là đào tạo thuật toán để xây dựng mô hình.

Phần còn lại là bộ thử nghiệm sẽ kiểm tra mô hình để xem độ chính xác của các dự đoán của nó.

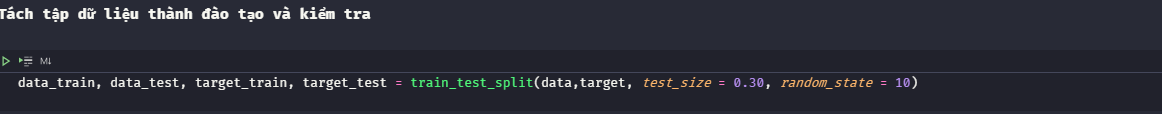


Hình 3 6 Bản ghi dữ liệu khi đã lọc và chọn trường để training.

Đầu tiên: Bỏ cột Opportunity Number vì nó chỉ là 1 mã định danh duy nhất cho mỗi bản ghi. Và Opportunity Result là mục tiêu sẽ hướng đến. Vì vâỵ Trong dòng code đầu tiên chỉ chọn các cột không khớp với Opportunity Number và Opportunity Result và gán cho chúng vào biến cols.

Tiếp theo tạo 1 khung dữ liệu mới data với các cột trong danh sách cols. Đây sẽ là bộ tính năng. Sau đó lấy cột ‘Opportunity Result’ từ dữ liệu sales\_data và tạo khung dữ liệu target.

Sau khi xác định được tính năng và mục tiêu thành 2 khung dữ liệu riêng biêt. Tiếp theo sẽ chia các khung dữ liệu data và target thành các tập training và test. Khi chia nhỏ tập dữ liệu, giữ 30% dữ liệu làm dữ liệu thử nghiệm và 70% còn lại là dữ liệu training. Sử dụng train\_test\_split() trong scikit-learn có thể sử dụng để phân chia các dữ liệu.



Hình 3 7 Mã lệnh chia nhỏ khung dữ liệu.

Chia dữ liệu thành tập training(data\_train,target\_train) và test (data\_test, data\_train). Đối số đầu tiên của train\_test\_split() là các tính năng đã tách ra ở phần trên, đối số thứ 2 là mục tiêu(Opportunity Result), đối số thứ 3 ‘test\_size’ là phần trăm dữ liệu mà chúng tôi muốn tách ra làm dữ liệu đào tạo, đối sớ thứ 4 ‘random\_state’ chỉ đảm bảo rằng nhận được kết quả có thể lặp lại mọi lúc. Vậy là mọi thứ sẵn sàng để đến xây dựng mô hình dự đoán bằng cách sử dụng thư viện thuật toán có sẵn trong scikit-learning.

## 3.5 Xây dựng mô hình

### 3.5.1 Xây dựng mô hình với thuật toán Naive-Bayes

Scikit-learn cung cấp 1 tập hợp các thuật toán phân loại ‘naively’ giả định nằm trong 1 tập dữ liệu, mọi cặp tính năng đều độc lập. Mọi giả định này là nguyên lý của Bayes theorem.

Và thuật toán được sử dụng cho dữ liệu bán hàng là Gausian Naive Bayes.

Bây giờ triển khai thuật toán từ scikit-learn để tạo mô hình dự đoán:

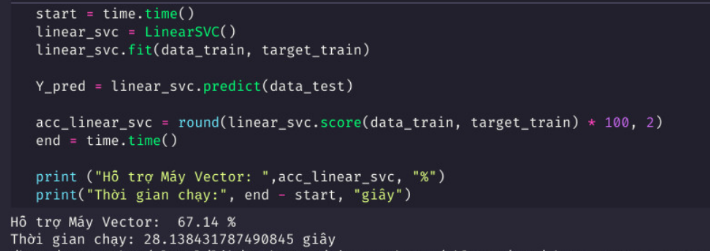


Hình 3 8 Mã lệnh và kết quả độ chính xác thuật toán Naive-Bayes

Đầu tiên nhập phương thức GaussianNB() và phương thức accuracy\_core(). Sau đó tạo 1 đội tượng gnb của GausianNB. Tiếp theo đào tạo thuật toán trên dữ liệu thử nghiệm (data\_train) và mục tiêu thử nghiệm (target\_train) bằng fit() phương pháp này, sau đó dự đoán các mục tiêu trong dữ liệu thử nghiệm bằng predict(). Cuối cùng in điểm bằng accuracy\_score() . Thành công xây dựng mô hình dự đoán.

### 3.5.2 Xây dựng mô hình với thuật toán LinearSVC

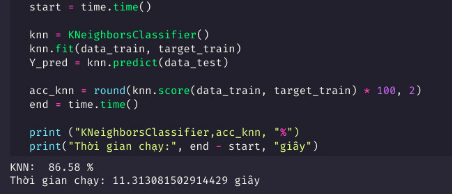
Tương tự như quá trình triển khai Naive-Bayes,đầu tiên tạo 1 đội tượng linear\_svc của LinearSVC(). Tiếp theo đào tạo thuật toán trên dữ liệu thử nghiệm (data\_train) và mục tiêu thử nghiệm (target\_train) bằng fit() phương pháp này, sau đó dự đoán các mục tiêu trong dữ liệu thử nghiệm bằng predict(). Cuối cùng in điểm bằng accuracy\_score() . Thành công xây dựng mô hình dự đoán

******

Hình 3 9 Mã lệnh và kết quả độ chính xác thuật toán LinearSVC.

### 3.5.3 Xây dựng mô hình với thuật toán K-Neighbors Classifier

Đầu tiên tạo 1 đội tượng knn của KneighborsClassifier(). Tiếp theo đào tạo thuật toán trên dữ liệu thử nghiệm (data\_train) và mục tiêu thử nghiệm (target\_train) bằng fit() phương pháp này, sau đó dự đoán các mục tiêu trong dữ liệu thử nghiệm bằng predict(). Cuối cùng in điểm bằng accuracy\_score() . Thành công xây dựng mô hình dự đoán.

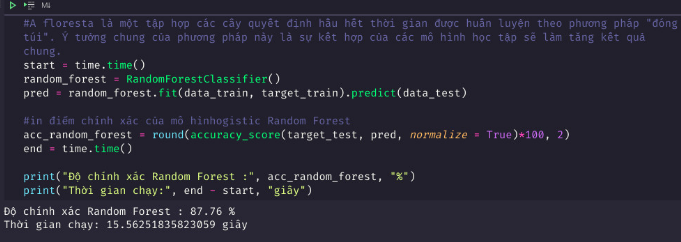


Hình 3 10 Mã lệnh và kết quả độ chính xác thuật toán KneighborsClassifier.

### 3.5.4 Xây dựng mô hình với thuật toán Random Forest, Logistic Regression, Perceptron

Tương tự như những thuật toán đã làm ở trên, ta có kết quả như sau:

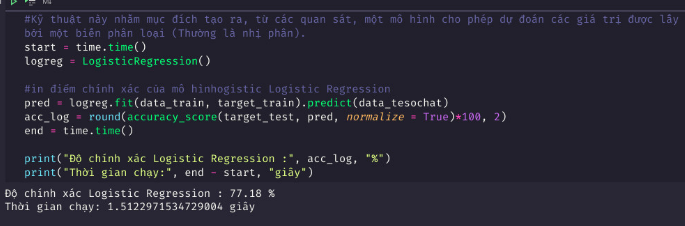
Random Forest:

******

Hình 3 11 Mã lệnh và kết quả độ chính xác thuật toán RandomForest.

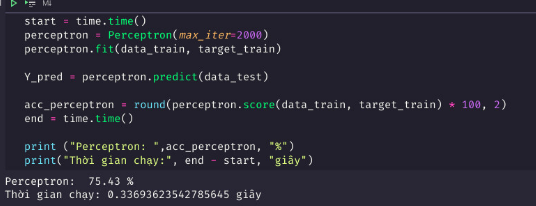
*Hình 21:*

Logistic Regression:

******

Hình 3 12 : Mã lệnh và kết quả độ chính xác thuật toán Logistic Regression.

Perceptron:

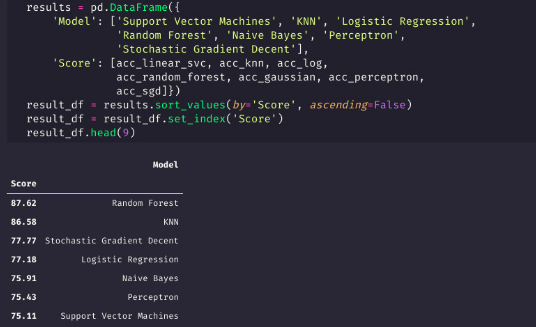
******

Hình 3 13 Mã lệnh và kết quả độ chính xác thuật toán Perceptron.

Khi đã triển khai tất cả các thuật toán trong danh sách của mình, chỉ cần so sánh điểm của tất cả các mô hình để chọn mô hình có điểm cao nhất. Vì vậy có thể sử dụng [yellowbrick](https://www.scikit-yb.org/en/latest/tutorial.html#visual-model-evaluation) thư viện trong scikit-learning, nơi cung cấp các phương pháp thể hiện trực quan các phương pháp tính điểm khác nhau.

## 3.6 So sánh hiệu suất

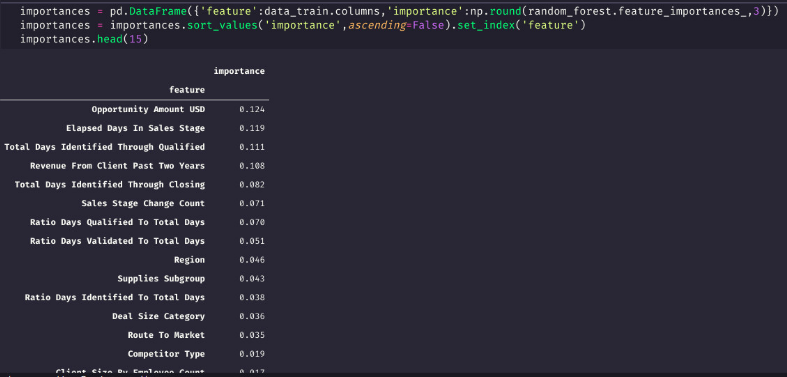
Ở đây chúng ta tiếp tục sử dụng DataFrame của pandas để phân tích dữ các điểm số của thuật toán trả về.



Hình 3 14 Mã lệnh và kết quả trả về độ chính xác của tất cả thuật toán.

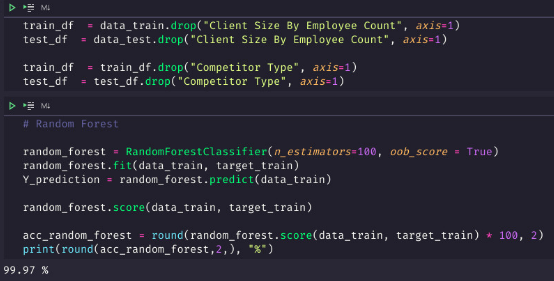
Vì điểm của thuật toán Random Forest cao nhất lên ta sẽ sử dụng Random Forest để xây dựng mô hình dự đoán.

Tiếp tục phân tích các dữ liệu để loại bỏ những cột có tỉ lệ thấp



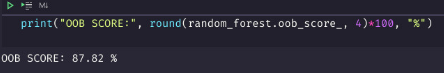
Hình 3 15 Tỉ lệ của các trường khi đã training.

Loại bỏ 2 trường ‘Client Size By Employee Count’ và ‘Competitor Type’ sau đó kiểm tra lại độ chính xác của thuật toán

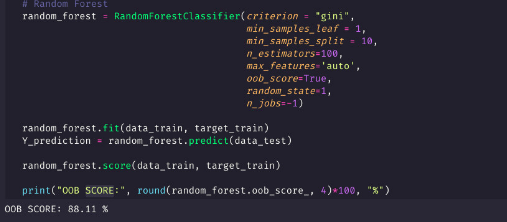


Hình 3 16 Độ chính xác thuật toán sau khi xử lý.

Kiểm tra OOB(tỉ lệ trung bình):



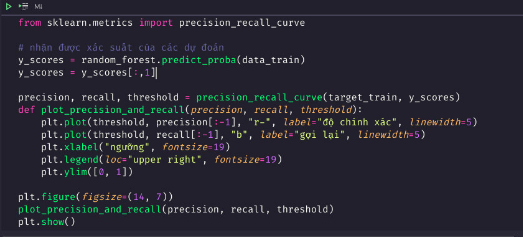
Hình 3 17 Độ chính xác trung bình cả thuật toán

Tiếp tục sử dụng RandomForestClassifier để training tăng tỉ lệ

Hình 3 18 Độ chính xác trung bình khi tiếp tục training.

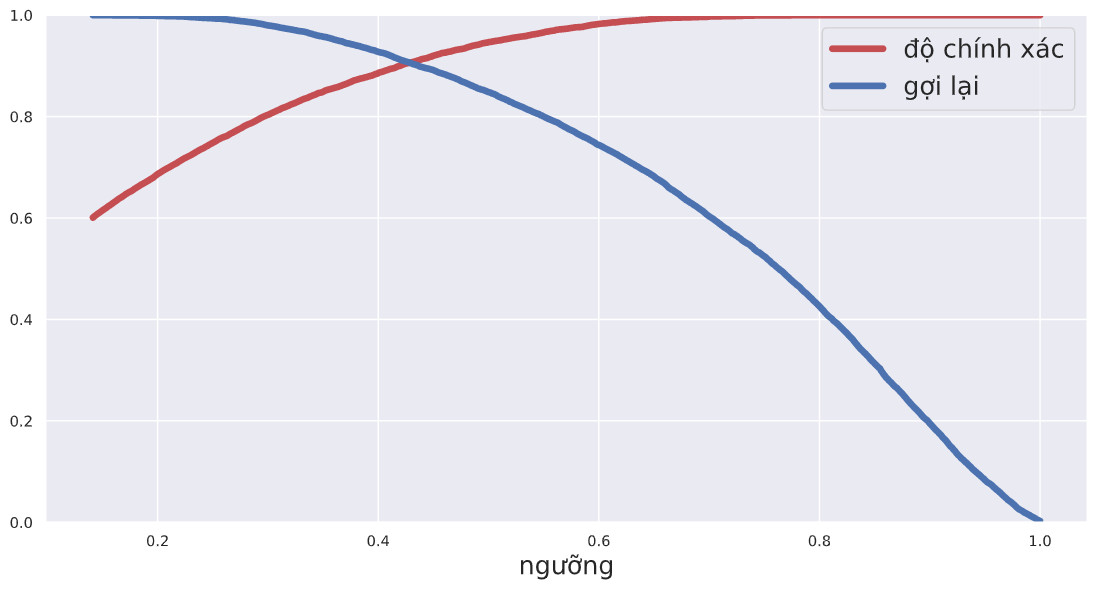
## 3.7 Phân tích và đưa ra kết quả

Sử dụng precision\_recall\_curve xây dựng biểu đồ đường cong để phân tích:



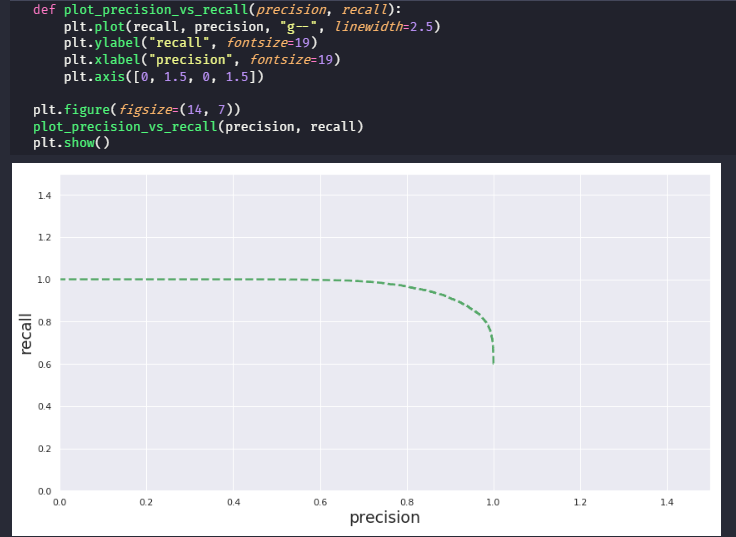
Hình 3 19 Mã lệnh xây dựng biều đồ đường cong sử dụng precision\_recall\_curve.

Biểu đồ:



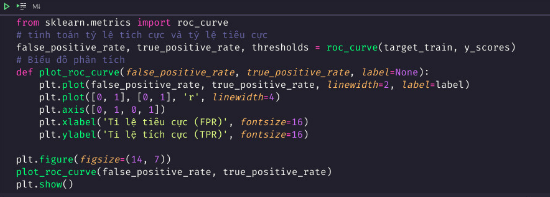
Hình 3 20 Biểu đồ đường cong của precision, recall và threshold.

Tiếp theo ta phân tích ngưỡng(precision) và recall(gợi lại):

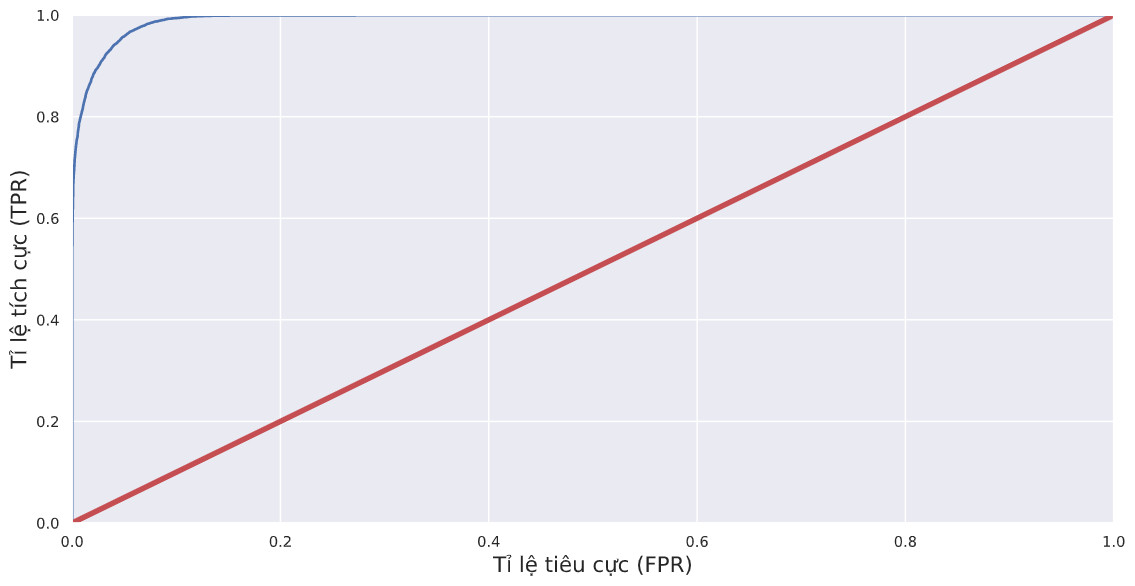


Hình 3 21 Biểu đồ precision và recall.

Cuối cùng tạo biểu đồ phân tích tỉ lệ tích cực và tiêu cực



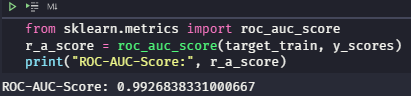
Hình 3 22 Mã lệnh vẽ biểu đồ tỉ lệ tích cực và tiêu cực.



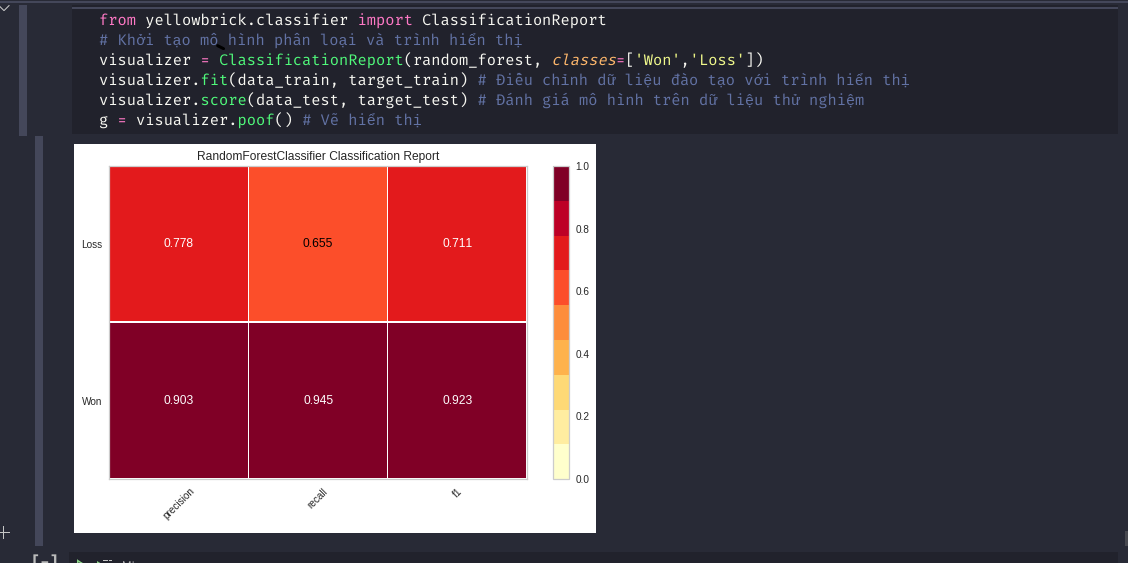
Hình 3 23 Biểu đồ tỉ lệ tích cực và tiêu cực.

*Hình 33:*

Sau khi sử dụng nhiều phương pháp phân tích dữ liệu, chuyển đổi dữ liệu training và kiểm tra, đánh giá kết quả bằng Confusion matrix đã trả về độ chính xác và độ bao quát cho từng lớp:



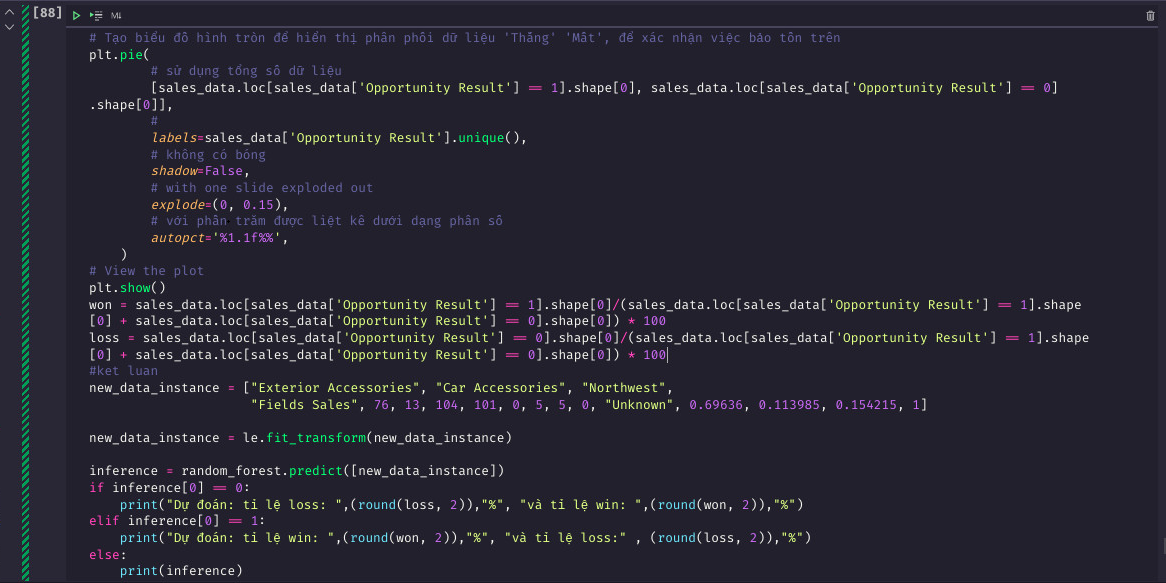
Hình 3 24 Kết quả hiệu suất của mô hình thuật toán.



Hình 3 25 Kết quả đánh giá của toàn bộ mô hình.

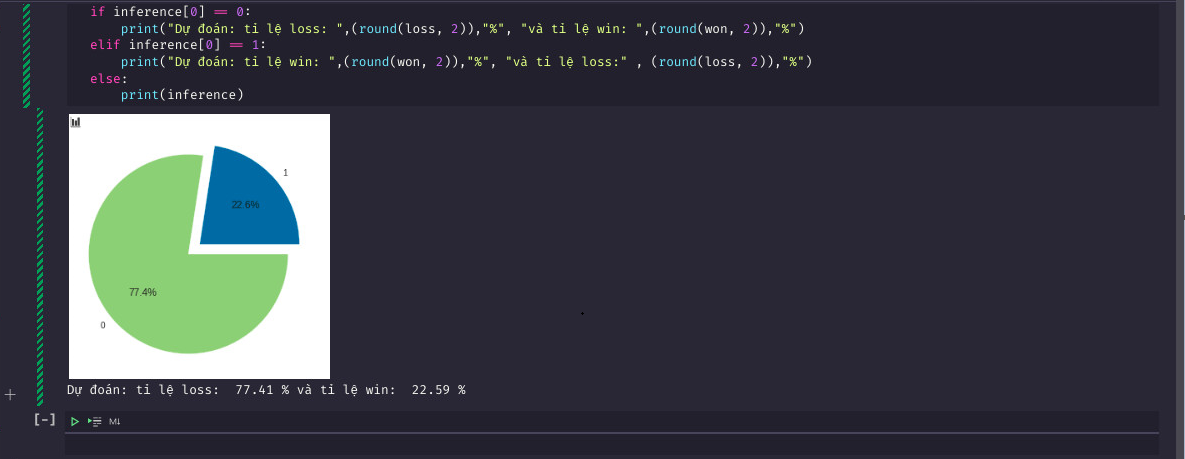
Trong đoạn mã trên, đầu tiên chúng ta nhập ClassficationReport() lớp được cung cấp bởi yellowbrick.classifer module. Tiếp theo, một đối tượng visualizer của ClassficationReport này đã được tạo ra. Ở đây, một số đầu tiên là Random Forest đối tượng random\_forest. Đối số thứ hai chứa các nhãn “won” và “loss” từ cột “Opportunity Result” từ khung sales\_data.

Sử dụng phương thức fit() để huấn luyện đối tượng 'random\_forest'. Tiếp theo là phương pháp score() sử dụng đối tượng radom\_forest để thực hiện các dự đoán theo thuật toán Random Forest và sau đó tính điểm chính xác của các dự đoán được thực hiện bởi thuật toán này. Cuối cùng, chúng tôi sử dụng phương thức poof() để vẽ biểu đồ của các điểm số khác nhau cho thuật toán Random Forest.



Hình 3 26 Mã lệnh dự đoán kết quả

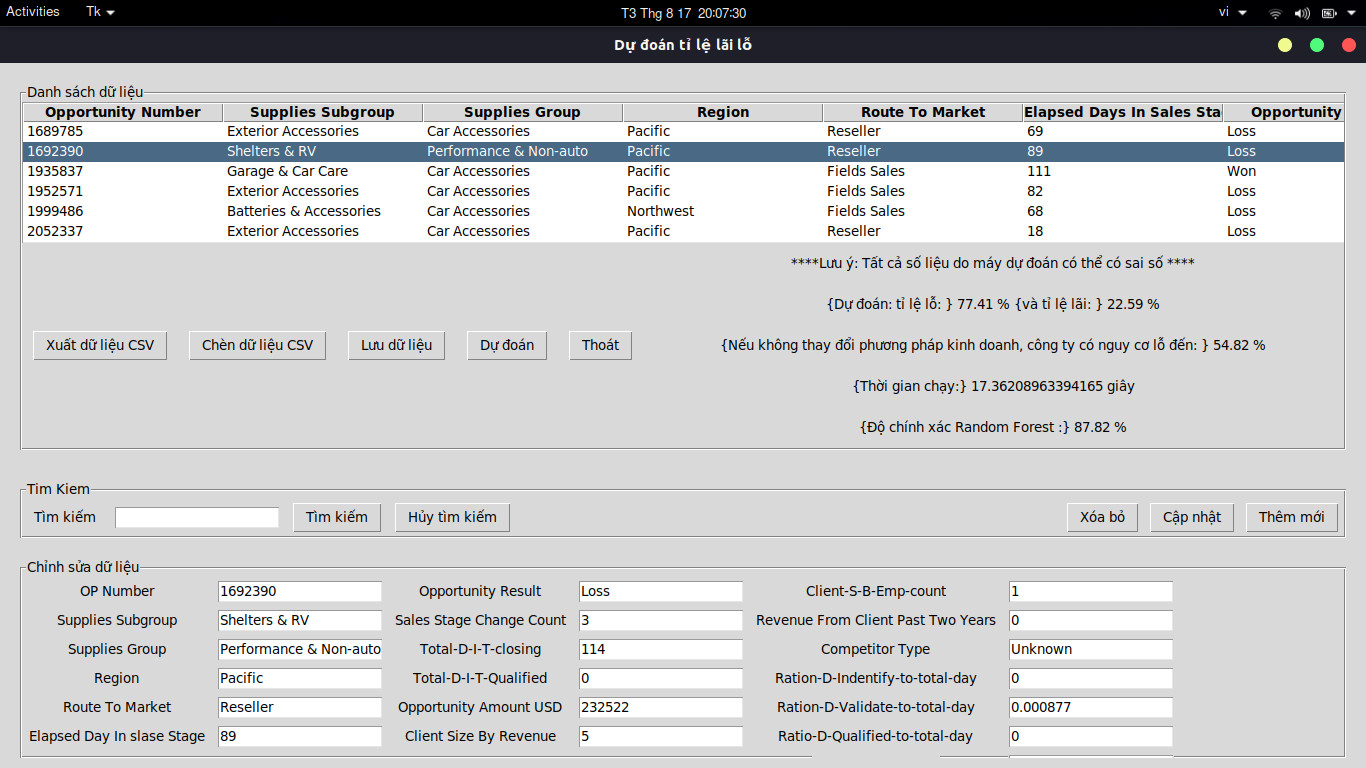
Chúng tôi sử dụng plot để hiển thị biểu đồ tổng quan tỉ lệ lãi-lỗ.



Hình 3 27 Biểu đồ và kết quả

Sau khi tìm được độ chính xác cao nhất của thuật toán Random forest. Chúng tôi đã sử dụng nó để đưa ra dự đoán về tỉ lệ lãi lỗ của doanh nghiệp. Và kết quả cuối cùng doanh nghiệp có tỉ lệ phải chịu lỗ lên tới 55%.

Cuối cùng khi mọi dữ liệu đã được chuẩn hóa và phân tích. Chúng tôi sử dụng thư viện Tkinter của python để xây dựng chương trình dự đoán tỉ lệ lãi lỗ của doanh nghiệp:



Hình 3 28 Chương trình dự đoán tỉ lệ lãi lỗ

# KẾT LUẬN VÀ KIẾN NGHỊ

**1. Kết luận**

Kết quả đạt được: Đề tài “*Nghiên cứu thư viện mã nguồn mở Sklearn xây dựng ứng dụng dự báo nhanh doanh thu của doanh nghiệp*” đã tổng quát hóa được những kiến thức về Machine Learning, nghiên cứu các thuật toán máy học bao gồm: Naive-Bayes, LinearSVC, K-Neighbors Classifier, Support Vector Machines, Random Forest ... Đồng thời nghiên cứu phương pháp ứng dụng thư viện máy học Sklearn nhằm đưa ra các dự báo trong lĩnh vực kinh doanh. Qua đó có thể ứng dụng thư viện này vào dự báo các lĩnh vực khác trong xã hội

**2. Kiến nghị**

Trong quá trình nghiên cứu, nhóm tác giả chưa hoàn toàn đưa ra được kết quả phù hợp với yêu cầu đặt ra trong đề cương nghiên cứu khoa học. Nhóm tác giả sẽ tiếp tục hoàn thiện, chỉnh sửa nhằm xây dựng được mô hình máy học xử lý tệp dữ liệu đầu vào nhằm đưa ra kết quả là dự báo doanh thu của doanh nghiệp một cách hoàn thiện hơn

# TÀI LIỆU THAM KHẢO

**TÀI LIỆU TIẾNG VIỆT**

1. Bùi Việt Hà, *“Python cơ bản”,* NXB Đại học Quốc gia Hà Nội.
2. Nguyễn Ngọc Giang, “ Đường Vào Lập Trình Python”, NXB ĐHQG HN

**TÀI LIỆU TIẾNG ANH**

1. Andreas C.Muller, *“Introduction to Machine Learning with Python:A Guide for Data Scientists”*
2. Charles Severance, “Python for Everybody”, CreateSapce Independent publishing Platform
3. Gareth Jame, Daniel Witen, Trevor Hastie, Robert Tibshirani *“An introduction to statistical learning with application in R”*, Springer
4. Jake VanderPlas*, “Dạo một vòng Python”,* O’Reilly
5. John C. Havens, “Heartificial Intelligence- Embracing Our Humanity to Maximize Machines”
6. Miroslav Kubat, *“An introduction to machine learning- second editon”,* Springe
7. Senbastian Raschka, “Python Machine Learning - Second Edition: Machine Learning with Python, scikit-learn, and TensorFlow”
8. scilit-learn.org