TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**



**BÁO CÁO THỰC TẬP NGHỀ NGHIỆP**

**NGÀNH HỆ THỐNG THÔNG TIN QUẢN LÝ**

**CHUYÊN NGÀNH QUẢN TRỊ HỆ THỐNG THÔNG TIN**

**ÁP DỤNG MÔ HÌNH HỌC MÁY ĐỂ PHÂN TÍCH VÀ DỰ ĐOÁN SỰ RỜI BỎ CỦA NHÂN VIÊN CÔNG TY IBM**

Sinh viên thực hiện : **Nguyễn Thị Bích Hà**

**Nguyễn Thị Mùi**

Lớp : **47K21.1**

Đơn vị thực tập : **Trung tâm VNPT – IT khu vực 3**

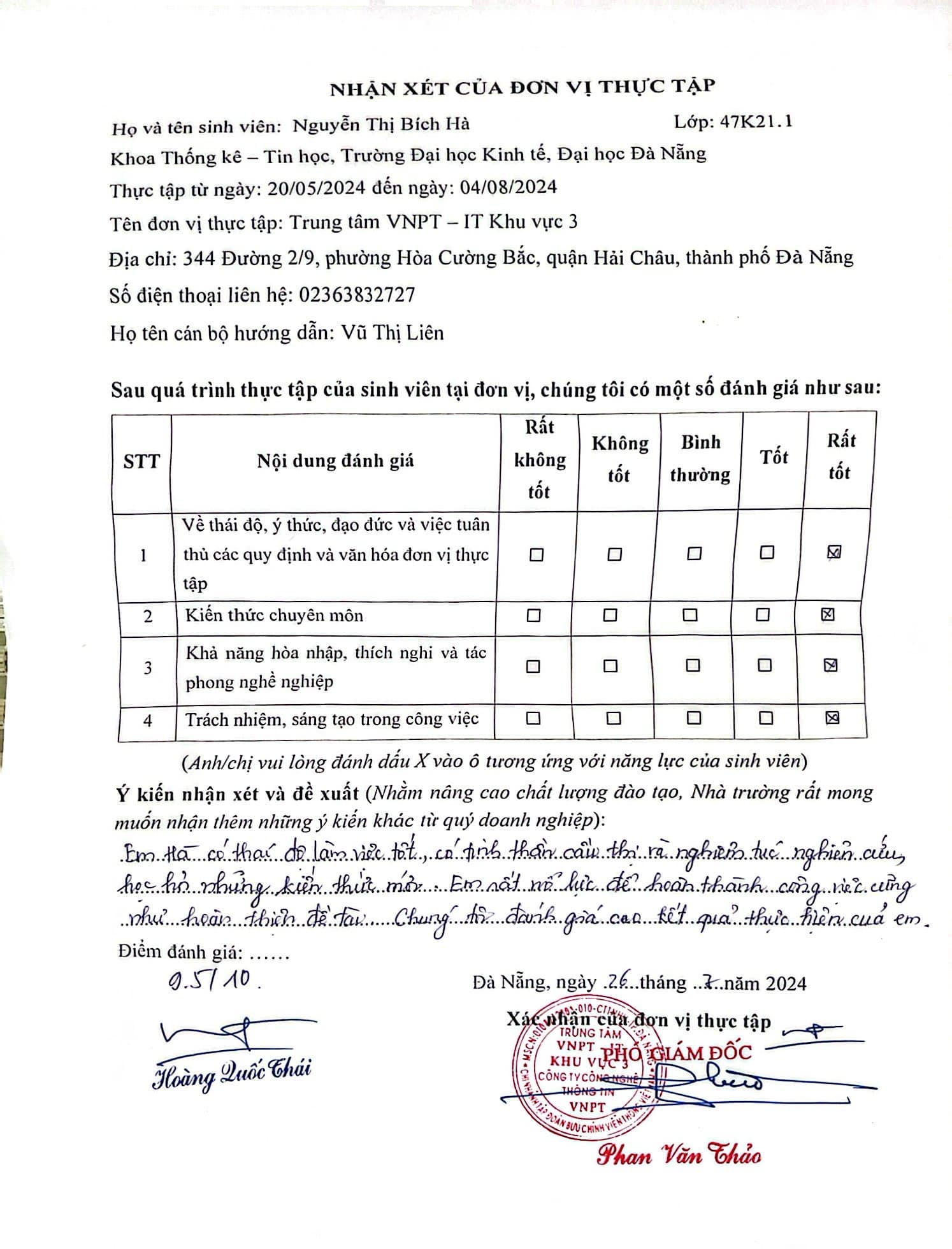
**STI Technology Solutions**

Cán bộ hướng dẫn : **Vũ Thị Liên**

**Thủy Việt Quốc**

Giảng viên hướng dẫn : **TS. Phan Đình Vấn**

***Đà Nẵng, 8/202******4***

****

**A paper with writing on it

Description automatically generated**

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin chân thành cảm ơn quý thầy cô và Ban lãnh đạo khoa Thống kê – Tin học trường Đại học Kinh tế - Đại học Đà Nẵng đã tạo điều kiện và cơ hội cho chúng em được đi thực tập tại doanh nghiệp thực tế. Những kiến thức và kỹ năng được học tập tại khoa đã giúp chúng em có nền tảng vững chắc để thực hiện tốt đề tài và các công việc tại đơn vị thực tập.

Đặc biệt, chúng em xin gửi lời cảm ơn sâu sắc đến thầy Phan Đình Vấn - người đã tận tình hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đề tài. Thầy đã dành nhiều thời gian theo sát quá trình của chúng em đưa ra các định hướng, giải đáp các thắc mắc giúp chúng em hoàn thành bài báo cáo một cách tốt nhất. Chúng em xin chân thành cảm ơn thầy.

Chúng em cũng xin gửi lời cảm ơn đến chị Vũ Thị Liên và anh Thủy Việt Quốc đã hướng dẫn và hỗ trợ chúng em tại đơn vị thực tập. Sự chỉ bảo tận tình và những kinh nghiệm quý báu từ anh/chị đã giúp chúng em học hỏi được rất nhiều điều bổ ích, bổ sung thêm kiến thức thực tế quý giá cho quá trình thực tập.

Vì kiến thức và kinh nghiệm còn nhiều hạn chế, chắc chắn bài báo cáo thực tập này không tránh khỏi những thiếu sót. Chúng em rất mong nhận được sự thông cảm và góp ý từ thầy để hoàn thiện tốt hơn bài báo cáo của mình.

Chúng em xin chân thành cảm ơn!

LỜI CAM ĐOAN

Chúng em xin cam đoan rằng đề tài “Áp dụng mô hình học máy để phân tích và dự đoán sự rời bỏ của nhân viên công ty IBM” là kết quả nghiên cứu dựa trên kiến thức của chúng em, được thực hiện dưới sự hướng dẫn tận tâm và nhiệt tình của thầy TS. Phan Đình Vấn. Chúng em đảm bảo rằng toàn bộ số liệu, kết quả và nội dung trong báo cáo này là hoàn toàn trung thực. Nếu phát hiện có gian lận, chúng em xin chịu hoàn toàn trách nhiệm.

MỤC LỤC

[LỜI CẢM ƠN iv](#_Toc173503026)

[LỜI CAM ĐOAN v](#_Toc173503027)

[MỤC LỤC vi](#_Toc173503028)

[DANH MỤC HÌNH ẢNH ix](#_Toc173503029)

[DANH MỤC BẢNG BIỂU xiii](#_Toc173503030)

[DANH MỤC CÁC TỪ VIẾT TẮT xiv](#_Toc173503031)

[LỜI MỞ ĐẦU 1](#_Toc173503032)

[CHƯƠNG 1. GIỚI THIỆU VỀ CÔNG TY VÀ NGHỀ DATA ANALYST 4](#_Toc173503033)

[1.1. Giới thiệu về công ty VNPT – IT3 4](#_Toc173503034)

[1.1.1. Thông tin chung 4](#_Toc173503035)

[1.1.2. Tầm nhìn và sứ mệnh 5](#_Toc173503036)

[1.1.3. Giá trị cốt lõi 5](#_Toc173503037)

[1.1.4. Cơ cấu tổ chức 5](#_Toc173503038)

[1.2. Giới thiệu về Công ty STI Technology Solutions 6](#_Toc173503039)

[1.2.1. Thông tin chung 6](#_Toc173503040)

[1.2.2. Tầm nhìn, Sứ mệnh 7](#_Toc173503041)

[1.2.3. Giá trị cốt lõi 7](#_Toc173503042)

[1.2.4. Lĩnh vực hoạt động 7](#_Toc173503043)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 9](#_Toc173503044)

[2.1. Các công cụ và phương pháp sử dụng 9](#_Toc173503045)

[2.1.1. Jupyter Notebook 9](#_Toc173503046)

[2.1.2. Python 10](#_Toc173503047)

[2.1.3. PowerBI 12](#_Toc173503048)

[2.2. Machine learning 13](#_Toc173503049)

[2.2.1. Thuật toán Logistic Regression 13](#_Toc173503050)

[2.2.2. Thuật toán Naïve Bayes 15](#_Toc173503051)

[2.2.3. Thuật toán Support Vector Machine 17](#_Toc173503052)

[2.2.4. Thuật toán Neural Network 18](#_Toc173503053)

[2.2.5. Confusion Matrix 21](#_Toc173503054)

[2.2.6. Ý nghĩa đường cong AUC và Roc 23](#_Toc173503055)

[CHƯƠNG 3. PHÂN TÍCH KHÁM PHÁ VÀ XỬ LÝ DỮ LIỆU 26](#_Toc173503056)

[3.1. Giới thiệu dữ liệu 26](#_Toc173503057)

[3.2. Tiền xử lý dữ liệu 28](#_Toc173503058)

[3.2.1. Import thư viện 28](#_Toc173503059)

[3.2.2. Đọc dữ liệu vào 29](#_Toc173503060)

[3.2.3. Xử lý các giá trị Null 29](#_Toc173503061)

[3.3. Phân tích khám phá dữ liệu 29](#_Toc173503062)

[3.3.1. Kiểm tra kích thước và kiểu dữ liệu 29](#_Toc173503063)

[3.3.2. Kiểm tra sự trùng lặp của dữ liệu 30](#_Toc173503064)

[3.3.3. Kiểm tra các cột có giá trị duy nhất 31](#_Toc173503065)

[3.3.4. Xử lý các cột có giá trị không đổi 31](#_Toc173503066)

[3.3.5. Phân tích thống kê mô tả 32](#_Toc173503067)

[3.4. Trực quan hóa dữ liệu 34](#_Toc173503068)

[CHƯƠNG 4. XÂY DỰNG MÔ HÌNH DỰ ĐOÁN SỰ RỜI BỎ CỦA NHÂN VIÊN CÔNG TY IBM 40](#_Toc173503069)

[4.1. Mô hình Logistic Regression 40](#_Toc173503070)

[4.1.1. Chuẩn bị dữ liệu 40](#_Toc173503071)

[4.1.2. Huấn luyện mô hình 44](#_Toc173503072)

[4.1.3. Đánh giá mô hình 44](#_Toc173503073)

[4.2. Mô hình Naïve Bayes 48](#_Toc173503074)

[4.2.1. Chuẩn bị dữ liệu 48](#_Toc173503075)

[4.2.2. Huấn luyện mô hình 52](#_Toc173503076)

[4.2.3. Đánh giá mô hình 53](#_Toc173503077)

[4.3. Mô hình Support Vector Machine 57](#_Toc173503078)

[4.3.1. Chuẩn bị dữ liệu 57](#_Toc173503079)

[4.3.2. Huấn luyện mô hình 58](#_Toc173503080)

[4.3.3. Đánh giá mô hình 61](#_Toc173503081)

[4.4. Mô hình Neural Network 63](#_Toc173503082)

[4.4.1. Chuẩn bị dữ liệu 63](#_Toc173503083)

[4.4.2. Huấn luyện mô hình 65](#_Toc173503084)

[4.4.3. Đánh giá mô hình 68](#_Toc173503085)

[4.5. Đánh giá và đưa ra kết luận 71](#_Toc173503086)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 73](#_Toc173503087)

[TÀI LIỆU THAM KHẢO 74](#_Toc173503088)

[CHECK LIST CỦA BÁO CÁO 76](#_Toc173503089)

[PHỤ LỤC 77](#_Toc173503090)

DANH MỤC HÌNH ẢNH

[Hình 1.1 Logo công ty VNPT IT 4](#_Toc173445646)

[Hình 1.2 Logo công ty STI Technology Solutions 7](#_Toc173445647)

[Hình 2.1 Logo Jupyter Notebook 9](#_Toc173445648)

[Hình 2.2 Logo Python 11](#_Toc173445649)

[Hình 2.3 Logo Power BI 12](#_Toc173445650)

[Hình 2.4 Phương trình Logistic Regression 14](#_Toc173445651)

[Hình 2.5 Thuật toán Naïve Bayes 15](#_Toc173445652)

[Hình 2.6 Công thức thuật toán Naïve Bayes 16](#_Toc173445653)

[Hình 2.7 Tham số phạt C 18](#_Toc173445654)

[Hình 2.8 Kernel trick 18](#_Toc173445655)

[Hình 2.9 Hình minh họa Neural Network 19](#_Toc173445656)

[Hình 2.10 Minh họa số lớp của Neural Network 20](#_Toc173445657)

[Hình 2.11 Confusion Matrix 22](#_Toc173445658)

[Hình 2.12 Minh họa ROC 24](#_Toc173445659)

[Hình 3.1 Bộ dữ liệu đầu vào 26](#_Toc173445660)

[Hình 3.2 Import thư viện 28](#_Toc173445661)

[Hình 3.3 Đọc dữ liệu đầu vào 29](#_Toc173445662)

[Hình 3.4 Xử lý giá trị Null 29](#_Toc173445663)

[Hình 3.5 Code kiểm tra kích thước và kiểu dữ liệu 29](#_Toc173445664)

[Hình 3.6 Kết quả kiểu dữ liệu 30](#_Toc173445665)

[Hình 3.7 Code kiểm tra sự trùng lặp 30](#_Toc173445666)

[Hình 3.8 Xem số giá trị của cột 31](#_Toc173445667)

[Hình 3.9 Xóa cột không có ý nghĩa 32](#_Toc173445668)

[Hình 3.10 Kết quả dữ liệu mới 32](#_Toc173445669)

[Hình 3.11 Chỉ số thống kê mô tả của dữ liệu 32](#_Toc173445670)

[Hình 3.12 Trực quan hóa dữ liệu 35](#_Toc173445671)

[Hình 3.13 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo phòng ban 36](#_Toc173445672)

[Hình 3.14 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo số năm làm việc 36](#_Toc173445673)

[Hình 3.15 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo mức lương 37](#_Toc173445674)

[Hình 3.16 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo giới tính 37](#_Toc173445675)

[Hình 3.17 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo tình trạng hôn nhân 38](#_Toc173445676)

[Hình 3.18 Biểu đồ xếp hạng mức độ hài lòng của nhân viên 38](#_Toc173445677)

[Hình 4.1 Chuẩn bị thư viện và đọc dữ liệu 40](#_Toc173445678)

[Hình 4.2 In số giá trị duy nhất của các cột 41](#_Toc173445679)

[Hình 4.3 In các cột có kiểu dữ liệu là object 41](#_Toc173445680)

[Hình 4.4 Chuyển đổi dữ liệu về dạng số 42](#_Toc173445681)

[Hình 4.5 Mối quan hệ tương quan 42](#_Toc173445682)

[Hình 4.6 Tạo dataframe mới 43](#_Toc173445683)

[Hình 4.7 Chia dữ liệu 43](#_Toc173445684)

[Hình 4.8 Cân bằng dữ liệu 43](#_Toc173445685)

[Hình 4.9 Chuẩn hóa dữ liệu 44](#_Toc173445686)

[Hình 4.10 Tách dữ liệu và huấn luyện mô hình 44](#_Toc173445687)

[Hình 4.11 Độ chính xác của mô hình Logistic Regression 45](#_Toc173445688)

[Hình 4.12 Các chỉ số đánh giá mô hình của Logistic Regression 45](#_Toc173445689)

[Hình 4.13 Mô hình Confusion Matrix của Logistic Regression 46](#_Toc173445690)

[Hình 4.14 Mô hình ROC của Logistic Regression 47](#_Toc173445691)

[Hình 4.15 Import thư viện và đọc dữ liệu vào 48](#_Toc173445692)

[Hình 4.16 Code chuyển đổi biến phân loại 49](#_Toc173445693)

[Hình 4.17 Dữ liệu sau khi chuyển đổi 49](#_Toc173445694)

[Hình 4.18 Kiểm tra độ tương quan 49](#_Toc173445695)

[Hình 4.19 Kết quả độ tương quan 50](#_Toc173445696)

[Hình 4.20 Tạo dữ liệu mới 50](#_Toc173445697)

[Hình 4.21 Dữ liệu mới 51](#_Toc173445698)

[Hình 4.22 Chia dữ liệu 51](#_Toc173445699)

[Hình 4.23 Dữ liệu trước khi cân bằng 51](#_Toc173445700)

[Hình 4.24 Dữ liệu sau khi cân bằng 52](#_Toc173445701)

[Hình 4.25 Chuẩn hóa dữ liệu 52](#_Toc173445702)

[Hình 4.26 Tách dữ liệu 52](#_Toc173445703)

[Hình 4.27 Huấn luyện mô hình Naïve Bayes 52](#_Toc173445704)

[Hình 4.28 Code tạo mô hình đánh giá 53](#_Toc173445705)

[Hình 4.29 Các chỉ số đánh giá mô hình Naïve Bayes 53](#_Toc173445706)

[Hình 4.30 Mô hình Confusion Matrix của Naïve Bayes 55](#_Toc173445707)

[Hình 4.31 Mô hình ROC của Naïve Bayes 56](#_Toc173445708)

[Hình 4.32 Chuyển đổi dữ liệu 57](#_Toc173445709)

[Hình 4.33 Dữ liệu sau khi chuyển đổi 58](#_Toc173445710)

[Hình 4.34 Dữ liệu sau khi chuẩn hóa 58](#_Toc173445711)

[Hình 4.35 Import thư viện SVM 59](#_Toc173445712)

[Hình 4.36 Từ điển chứa siêu tham số 59](#_Toc173445713)

[Hình 4.37 Huấn luyện mô hình với siêu tham số tốt nhất 60](#_Toc173445714)

[Hình 4.38 Độ chính xác của mô hình SVM 61](#_Toc173445715)

[Hình 4.39 Chỉ số đánh giá mô hình SVM 61](#_Toc173445716)

[Hình 4.40 Mô hình Confusion Matrix của SVM 62](#_Toc173445717)

[Hình 4.41 Mô hình ROC của SVM 63](#_Toc173445718)

[Hình 4.42 Dữ liệu sau khi chuyển đổi 64](#_Toc173445719)

[Hình 4.43 Chia dữ liệu và cân bằng dữ liệu 64](#_Toc173445720)

[Hình 4.44 Chuẩn hóa dữ liệu 64](#_Toc173445721)

[Hình 4.45 Chia dữ liệu 65](#_Toc173445722)

[Hình 4.46 Thư viện cho Neural Network 65](#_Toc173445723)

[Hình 4.47 Xây dựng mô hình ANN 66](#_Toc173445724)

[Hình 4.48 Biên dịch và huấn luyện mô hình ANN 67](#_Toc173445725)

[Hình 4.49 Đánh giá hiệu suất mô hình ANN 68](#_Toc173445726)

[Hình 4.50 Các chỉ số đánh giá mô hình ANN 68](#_Toc173445727)

[Hình 4.51 Mô hình Confusion Matrix của ANN 69](#_Toc173445728)

[Hình 4.52 Mô hình ROC 70](#_Toc173445729)

DANH MỤC BẢNG BIỂU

[Bảng 3.1 Mô tả trường dữ liệu 26](#_Toc173419575)

[Bảng 4.1 Tổng hợp các mô hình và tham số 72](#_Toc173419576)

DANH MỤC CÁC TỪ VIẾT TẮT

**DA** : Data Analyst

**SVM** : Support Vector Machine

**AUC**  : Area Under The Curve

**ROC** : Receiver Operating Characteristics.

**TPR** : True Positive Rate

**FPR** : False Positive Rate

**SMOTE** : Synthetic Minority Over-sampling

**ANN** : Artificial Neural Network

**ReLU** :Rectified Linear Unit

LỜI MỞ ĐẦU

1. **Lý do chọn đề tài**

Lý do chúng em chọn đề tài "Áp dụng mô hình học máy để phân tích và dự đoán sự rời bỏ của nhân viên công ty IBM " xuất phát từ tầm quan trọng và tính cấp thiết của việc quản lý nguồn nhân lực trong các doanh nghiệp hiện nay. Tình trạng nhân viên rời bỏ công ty đang trở thành một vấn đề ngày càng nghiêm trọng, ảnh hưởng trực tiếp đến hiệu quả hoạt động và sự phát triển bền vững của doanh nghiệp.

Đối với một công ty lớn như IBM, việc duy trì đội ngũ nhân viên ổn định và giàu kinh nghiệm là một yếu tố then chốt ảnh hưởng sâu sắc đến sự thành công và sự phát triển bền vững của công ty. Sự rời bỏ của những nhân viên có kinh nghiệm làm việc lâu năm không chỉ gây ra gián đoạn trong quy trình làm việc mà còn làm tăng chi phí tuyển dụng và đào tạo nhân viên mới. Hơn nữa, việc mất đi những nhân tài quý giá như vậy có thể ảnh hưởng nghiêm trọng đến tinh thần và hiệu suất làm việc của những nhân viên còn lại.

Đề tài này cho phép chúng ta khai thác bộ dữ liệu để khám phá những góc nhìn ẩn sâu bên trong của bộ dữ liệu và ứng dụng các kỹ thuật học máy tiên tiến như hồi quy logistic, rừng ngẫu nhiên, và học sâu để phân tích dữ liệu nhân viên của IBM. Qua đó, chúng ta có thể xác định các yếu tố chính ảnh hưởng đến quyết định rời bỏ công ty của nhân viên và dự đoán nguy cơ rời bỏ với độ chính xác cao. Kết quả từ các mô hình dự đoán này sẽ giúp IBM triển khai các biện pháp phòng ngừa kịp thời, cải thiện môi trường làm việc, và tăng cường sự hài lòng cũng như sự gắn kết của nhân viên.

Việc nghiên cứu và áp dụng mô hình học máy để quản lý sự rời bỏ của nhân viên không chỉ mang lại lợi ích thiết thực cho IBM mà còn đóng góp quan trọng vào việc cung cấp thêm kiến thức và kinh nghiệm trong lĩnh vực phân tích dữ liệu và quản lý nguồn nhân lực nói chung. Đây là một hướng nghiên cứu mang tính ứng dụng cao, có khả năng mở rộng và áp dụng cho nhiều doanh nghiệp khác, giúp nâng cao hiệu quả quản lý nhân sự và thúc đẩy sự phát triển bền vững. Việc này không chỉ giúp IBM mà còn tạo ra giá trị cho toàn ngành, góp phần xây dựng một nền tảng vững chắc cho quản lý nhân sự hiện nay

1. **Mục tiêu của đề tài**

Phân tích và khám phá các yếu tố có thể ảnh hưởng đến quyết định rời bỏ công ty của nhân viên IBM

Áp dụng các mô hình học máy để dự đoán khả năng rời bỏ của nhân viên dựa trên các yếu tố đã phân tích.

Đánh giá hiệu suất mô hình: Đánh giá độ chính xác, độ nhạy, độ đặc hiệu và các chỉ số khác của các mô hình dự đoán để chọn ra mô hình tốt nhất.

Dựa trên các kết quả phân tích và dự đoán, đưa ra các giải pháp, chiến lược giúp giảm tỷ lệ rời bỏ của nhân viên, cải thiện môi trường làm việc và giữ chân nhân tài. Từ đó giúp công ty có những biện pháp quản lý và giữ chân nhân viên hiệu quả hơn.

1. **Đối tượng và phạm vi nghiên cứu**

Đối tượng nghiên cứu của đề tài này là toàn bộ nhân viên tại IBM, bao gồm cả nhân viên chính thức và hợp đồng, trên tất cả các bộ phận và vị trí . Nghiên cứu tập trung phân tích các yếu tố ảnh hưởng đến quyết định rời bỏ công ty của nhân viên, sử dụng dữ liệu nhân sự của IBM.

Phương pháp nghiên cứu gồm tiền xử lý dữ liệu để đảm bảo chất lượng và tính nhất quán, sau đó sử dụng bộ dữ liệu đã xử lý tiến hành phân tích và tìm insight của dữ liệu. Sau phân tích xong, tiếp tục xây dựng và huấn luyện các mô hình dự đoán như hồi quy logistic, Naïve bayes và các thuật toán khác. Các mô hình sẽ được đánh giá và so sánh hiệu suất dựa trên các chỉ số như độ chính xác, độ nhạy, độ đặc hiệu và hệ số F1. Mục tiêu là dự đoán chính xác khả năng nhân viên rời bỏ công ty, từ đó đề xuất các giải pháp giảm tỷ lệ rời bỏ, cải thiện môi trường làm việc và giữ chân nhân tài. Nghiên cứu hy vọng đưa ra khuyến nghị hiệu quả để hỗ trợ IBM trong quản lý nhân sự và phát triển bền vững.

1. **Kết cấu của đề tài**

Đề tài được tổ chức gồm phần mở đầu, 4 chương nội dung và phần kết luận

* Mở đầu: Nêu lý do chọn đề tài
* **Chương 1**: Giới thiệu về đơn vị thực tập và vị trí Data Analyst
* **Chương 2**: Cơ sở lý thuyết
* **Chương 3**: Phân tích khám phá và xử lý dữ liệu
* **Chương 4**: Xây dựng mô hình dự đoán sự rời bỏ của nhân viên công ty IBM
* Kết luận và hướng phát triển

1. **Tổng quan về đề tài dự đoán sự rời bỏ công ty của nhân viên**

Tình trạng rời bỏ của nhân viên đang trở thành một thách thức lớn đối với rất nhiều doanh nghiệp. Nhiều công ty đang phải đối mặt với tỷ lệ nghỉ việc cao, gây ra những tổn thất về chi phí tuyển dụng, đào tạo và thời gian để nhân viên mới thích nghi với công việc. Các nguyên nhân chính dẫn đến tình trạng này bao gồm môi trường làm việc căng thẳng, thiếu cơ hội thăng tiến, chính sách phúc lợi không hấp dẫn, và sự mất cân bằng giữa công việc và cuộc sống cá nhân. Đặc biệt, trong bối cảnh hậu đại dịch COVID-19, nhiều nhân viên đã có thời gian đánh giá khả năng và điểm mạnh của mình và quyết định rời bỏ công việc hiện tại để tìm kiếm những cơ hội tốt hơn, có sự phát triển hơn trong sự nghiệp của họ. Điều này đặt ra yêu cầu cấp thiết cho các doanh nghiệp phải tìm ra các biện pháp hiệu quả để giữ chân nhân tài, cải thiện môi trường làm việc và tạo ra một văn hóa công ty tích cực.

Từ đó, việc sử dụng các phương pháp phân tích dữ liệu và xây dựng mô hình dự đoán sự rời bỏ của nhân viên trở nên vô cùng quan trọng. Thông qua việc phân tích các dữ liệu liên quan đến hành vi nhân viên dựa trên các yếu tố như lịch sử làm việc, hiệu suất, mức độ hài lòng, và các yếu tố khác, doanh nghiệp có thể xác định các dấu hiệu tiềm ẩn của việc rời bỏ của nhân viên trước khi nó xảy ra. Việc sử dụng các mô hình dự đoán, chẳng hạn như mô hình logistic regression, mô hình SVM (Support Vector Machine),... có thể giúp doanh nghiệp dự báo chính xác những nhân viên nào có nguy cơ nghỉ việc cao nhất, từ đó triển khai các biện pháp can thiệp kịp thời. Điều này không chỉ giúp giảm thiểu tỷ lệ nghỉ việc mà còn nâng cao sự hài lòng và gắn bó của nhân viên, tạo ra một môi trường làm việc bền vững và hiệu quả hơn. Nhờ vậy, doanh nghiệp có thể tối ưu hóa nguồn nhân lực, tiết kiệm chi phí, duy trì sự ổn định và phát triển bền vững hơn.

Bài báo cáo này sẽ tập trung vào việc phân tích và khám phá dữ liệu, trực quan hóa trên Power BI, và sử dụng các mô hình học máy từ thư viện scikit-learn của Python để phân tích và dự đoán sự rời bỏ của nhân viên IBM dựa trên các yếu tố hành vi. Từ các kết quả phân tích và dự đoán, chúng em sẽ đề xuất các giải pháp và chiến lược nhằm giảm tỷ lệ rời bỏ, cải thiện môi trường làm việc, và giữ chân nhân tài.

# GIỚI THIỆU VỀ CÔNG TY VÀ NGHỀ DATA ANALYST

## Giới thiệu về công ty VNPT – IT3

### Thông tin chung

Công ty Công nghệ thông tin VNPT (viết tắt: VNPT-IT) được thành lập ngày 01/03/2018 theo Quyết định số 39/QĐ-VNPT-HĐTV-NL của Chủ tịch Tập đoàn Bưu chính Viễn thông Việt Nam. Công ty được tái tổ chức từ các nhiệm vụ và nguồn lực công nghệ thông tin thuộc Tập đoàn. VNPT-IT tập trung vào nghiên cứu, phát triển và tích hợp các sản phẩm, dịch vụ công nghệ thông tin, không chỉ phục vụ nội bộ Tập đoàn VNPT mà còn mở rộng đến khách hàng bên ngoài, bao gồm cả thị trường quốc tế.



Hình 1.1 Logo công ty VNPT IT

Công ty đặt mục tiêu xây dựng hệ sinh thái tích hợp trọn gói các sản phẩm và dịch vụ công nghệ thông tin và Internet lớn nhất Việt Nam, từ đó mở rộng ra thị trường quốc tế. Để đạt được điều này, VNPT-IT định hướng phát triển dựa trên bốn giá trị cốt lõi: Con người là chìa khóa, khách hàng là trung tâm, sáng tạo không ngừng, và đối tác đáng tin cậy.

Về lĩnh vực kinh doanh VNPT-IT chuyên nghiên cứu, phát triển, sản xuất và kinh doanh các sản phẩm, dịch vụ công nghệ thông tin để cung cấp cho cả nội bộ VNPT và khách hàng bên ngoài. Công ty đầu tư, phát triển và quản lý các hệ thống, nền tảng công nghệ thông tin; vận hành, khai thác hệ thống điều hành sản xuất kinh doanh, đồng thời đảm bảo an toàn và bảo mật thông tin cho các sản phẩm và dịch vụ công nghệ thông tin mà VNPT cung cấp cho khách hàng. [1]

### Tầm nhìn và sứ mệnh

Tầm nhìn của VNPT-IT thể hiện khát vọng mạnh mẽ và định hướng rõ ràng trong việc trở thành đơn vị tiên phong trong lĩnh vực công nghệ thông tin. Với khát vọng tiên phong trong chiến lược đầu tư và cung cấp giải pháp, sản phẩm, dịch vụ công nghệ thông tin, VNPT-IT, công ty hàng đầu trong lĩnh vực này tại Việt Nam, cam kết phát triển theo hướng nghiên cứu, gia công và thử nghiệm. Công ty hướng tới cung cấp các sản phẩm đa dạng, uy tín và chất lượng cao, đáp ứng nhu cầu của xã hội và góp phần nâng cao chất lượng cuộc sống của người Việt, đồng thời nâng tầm vị thế của Việt Nam trên thị trường quốc tế. [1]

### Giá trị cốt lõi

Là công ty công nghệ trọng điểm quốc gia, Công ty Công nghệ thông tin VNPT-IT, thành viên của Tập đoàn Bưu chính Viễn thông Việt Nam, sẽ đóng vai trò chủ lực trong việc triển khai VNPT 4.0 và tham gia mạnh mẽ vào quá trình chuyển đổi nền kinh tế số của Việt Nam. VNPT-IT không chỉ cung cấp giải pháp xây dựng chính quyền số và nền kinh tế số mà còn hỗ trợ doanh nghiệp Việt Nam chuyển đổi thành doanh nghiệp số, cung cấp các nền tảng phát triển công nghệ số trong tương lai. VNPT-IT chính là hoài bão và khát vọng của VNPT trong hành trình khai phá lĩnh vực CNTT.

Với sứ mệnh và trọng trách cao cả, công ty cam kết cung cấp các giải pháp, dịch vụ và sản phẩm thông minh cho xã hội trong mọi lĩnh vực, từ giáo dục, quản lý, chăm sóc sức khỏe, đến ứng dụng và chuyển giao công nghệ, tất cả đều đạt chất lượng cao với thương hiệu và danh tiếng đẳng cấp khu vực và quốc tế. VNPT-IT góp phần quan trọng vào sự nghiệp công nghiệp hóa, hiện đại hóa đất nước trong bối cảnh hội nhập kinh tế toàn cầu.

### Cơ cấu tổ chức

Về Cơ cấu của VNPT-IT bao gồm Ban Tổng giám đốc, Văn phòng các Ban chức năng, các Trung tâm trực thuộc và các Trung tâm tại Hà Nội, Hồ Chí Minh, Đà Nẵng, Hải Phòng và Tiền Giang. Các Trung tâm này là đơn vị hạch toán phụ thuộc của Công ty.

Là một trong những đơn vị chủ chốt của Tập đoàn Bưu chính Viễn thông Việt Nam, VNPT-IT không ngừng phấn đấu nâng cao chất lượng sản phẩm và dịch vụ về mọi mặt để trở thành thương hiệu uy tín trong lĩnh vực công nghệ thông tin, góp phần đưa VNPT đạt mục tiêu trở thành Tập đoàn Viễn thông - CNTT hàng đầu quốc gia và giữ vai trò chủ đạo trong lĩnh vực Viễn thông và CNTT tại Việt Nam. [1]

## Giới thiệu về Công ty STI Technology Solutions

### Thông tin chung

Công ty Cổ phần Giải pháp Công nghệ STI (STI Technology Solutions) được thành lập vào năm 2021 với sự lãnh đạo của Tổng giám đốc Nguyễn Võ Quang Đông. STI với sứ mệnh mang đến những giải pháp công nghệ toàn diện và xuyên suốt, là nơi quy tụ những cá nhân có cùng niềm đam mê công nghệ giúp mang lại giá trị cho khách hàng cũng như cho xã hội. STI cung cấp nền tảng và giải pháp công nghệ thông tin cho cả thị trường trong và ngoài nước nhằm hướng đến mục tiêu quốc tế hóa trong thời gian sắp tới.

Về văn hóa của STI “Mô hình văn hóa STI được thể hiện qua chính sách thời gian làm việc, hoạt động của công ty cũng như trong kế hoạch đào tạo nhân sự.”

STI với thông điệp từ lãnh đạo “Luôn đề cao chất lượng và trải nghiệm sản phẩm, chúng tôi đặt mình vào vị trí của người dùng khi phân tích, tư vấn, thiết kế và phát triển sản phẩm. Khi gắn kết chặt chẽ với khách hàng trong quá trình phát triển sản phẩm, chúng tôi mong muốn mang niềm đam mê công nghệ của mình để hỗ trợ khách hàng. Giảm chi phí và nâng cao chất lượng là yêu cầu trong quá trình tư vấn và phát triển sản phẩm của chúng tôi. Thước đo thành công của chúng tôi là nâng cao chất lượng dịch vụ và sự thành công của khách hàng. Chúng tôi sẽ luôn nỗ lực hết mình để mang đến chất lượng sản phẩm ngày càng cao hơn cho từng công trình của khách hàng.” [2]



Hình 1.2 Logo công ty STI Technology Solutions

### Tầm nhìn, Sứ mệnh

Tầm nhìn của STI là phấn đấu trở thành người tiên phong trong lĩnh vực công nghệ mới và trở thành nơi của những người đam mê công nghệ, luôn cung cấp các giải pháp giải quyết vấn đề của khách hàng một cách tối ưu và triệt để.

STI với sứ mệnh khơi dậy niềm đam mê công nghệ, phát triển toàn diện cho nhân viên và kết nối để tạo ra giá trị cho khách hàng và xã hội. [2]

### Giá trị cốt lõi

Tại công ty, xây dựng nền văn hóa doanh nghiệp dựa trên 5 giá trị cốt lõi: Chia sẻ, Sự đổi mới, Sự tôn trọng, Sự đoàn kết và Trách nhiệm. Thông qua việc chia sẻ, công ty tạo ra môi trường làm việc cởi mở, khuyến khích sự giao lưu, hợp tác và trao đổi ý tưởng giữa các thành viên. Điều này góp phần thúc đẩy sự đổi mới, đưa các ý tưởng sáng tạo vượt trội trở thành hiện thực. Công ty tôn trọng mỗi cá nhân, công nhận và tôn vinh những đóng góp của từng thành viên. Điều này nuôi dưỡng sự gắn kết và lòng trung thành, qua đó mang lại sự đoàn kết mạnh mẽ trong toàn công ty. Cuối cùng, mỗi thành viên đều hiểu rõ và thực hiện trách nhiệm của mình, hướng tới mục tiêu chung của tổ chức. Với những giá trị này, STI không chỉ xây dựng một môi trường làm việc hiệu quả, mà còn tạo nền tảng vững chắc để tiếp tục phát triển bền vững. [2]

### Lĩnh vực hoạt động

STI là một công ty hoạt động nghiên cứu và phát triển công nghệ trên nhiều lĩnh vực khác nhau. Về lĩnh vực Web apps, STI đang phát triển các hệ thống web app như ứng dụng mua sắm thực phẩm trực tuyến, các hệ thống Fintech, và các hệ thống quản lý quy trình nội bộ. Về lĩnh vực Mobile apps, STI có các ứng dụng di động như ứng dụng mua bán tiền kỹ thuật số (Crypto), ứng dụng mua sắm trực tuyến, ứng dụng giao thực phẩm, và các ứng dụng Livestream. Trong lĩnh vực Blockchain, STI đang phát triển các hợp đồng thông minh để tạo ra tiền xu mới (Coin ICO), hệ thống StableCoin, và ứng dụng Blockchain trong lưu trữ dữ liệu. Còn ở lĩnh vực AI, STI có các hệ thống trí tuệ nhân tạo như hệ thống nhận dạng phông chữ (OCR), hệ thống nhận dạng hình ảnh camera, hệ thống xử lý âm thanh, hệ thống tạo nội dung và chú thích ảnh, cũng như hệ thống tạo nội dung theo các chủ đề nóng trên truyền thông xã hội. STI là một công ty đa năng, hoạt động trên nhiều lĩnh vực công nghệ tiên tiến như Web, Mobile, Blockchain, AI, Devops và Game. [2]

# CƠ SỞ LÝ THUYẾT

## Các công cụ và phương pháp sử dụng

### Jupyter Notebook

Jupyter Notebook - Khám Phá Thế Giới Tri Thức Qua Ngôn Ngữ Đa Dạng. Jupyter là một từ được tạo ra bằng cách gộp chung các ngôn ngữ lập trình nổi tiếng như Julia, Python và R. Đây chính là nền tảng tính toán khoa học mã nguồn mở, cho phép người dùng tạo và chia sẻ các tài liệu với đa dạng nội dung - từ mã code trực tiếp, các phương trình toán học, đến các trực quan hóa dữ liệu và văn bản giải thích. Jupyter được coi là một môi trường điện toán tương tác đa ngôn ngữ, nó hỗ trợ hơn 40 ngôn ngữ lập trình khác nhau, mang đến sự linh hoạt và sức mạnh vô tận cho người dùng. Ở đây, các nhà khoa học, lập trình viên và nghiên cứu viên có thể thỏa sức sáng tạo, chia sẻ những kiến thức và cùng hợp tác với nhau làm việc một cách hiệu quả. Với Jupyter Notebook, giới hạn của tri thức nhân loại dường như không còn bị gò bó bởi ranh giới của từng ngôn ngữ lập trình riêng lẻ. Thay vào đó, nó mở ra một thế giới đầy tiềm năng, nơi các ý tưởng và khám phá có thể được kết nối và phát triển một cách đa dạng và sáng tạo hơn bao giờ hết. [3]



Hình 2.1 Logo Jupyter Notebook

#### Các ứng dụng của Jupyter Notebook

Jupyter Notebook là một công cụ đa năng được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau. Nó đặc biệt hữu ích trong các tác vụ liên quan đến phân tích dữ liệu, bao gồm làm sạch dữ liệu, chuyển đổi dữ liệu và trực quan hóa dữ liệu. Ngoài ra, Jupyter Notebook cũng được ứng dụng trong lĩnh vực máy học, giúp các nhà nghiên cứu và chuyên gia dễ dàng xây dựng, huấn luyện và triển khai các mô hình máy học. Công cụ này cũng hỗ trợ các tác vụ mô phỏng số và xây dựng mô hình thống kê, mang lại lợi ích to lớn cho các ngành như khoa học, kỹ thuật và tài chính. Nhờ những tính năng mạnh mẽ và khả năng đa dạng ứng dụng, Jupyter Notebook đã trở thành một công cụ không thể thiếu trong thế giới khoa học dữ liệu và tính toán khoa học hiện đại ngày nay. [3]

#### Các lợi ích của Jupyter Notebook

Hỗ trợ nhiều ngôn ngữ: Hỗ trợ hơn 40 ngôn ngữ lập trình bao gồm Python, Scala, Julia và R

Khả năng chia sẻ: Dễ dàng chia sẻ notebooks của bạn với người khác bằng email, Dropbox, Github hoặc Jupyter Notebook Viewer 17

Kiểu đầu ra phong phú: Code có thể tạo ra đầu ra phong phú như HTML, LaTeX, hình ảnh và video

Tùy chỉnh: Có khả năng xây dựng các thành phần tùy chỉnh và tùy chỉnh JupyterLab để phù hợp với quy trình làm việc của bạn

Khả năng sử dụng: JupyterHub cho phép bạn chia sẻ notebooks với nhiều nhóm người dùng và Binder cho phép bạn sử dụng Jupyter trên GitHub trong trình duyệt.

Tài liệu: Các tùy chọn định dạng tuyệt vời ngay trên môi trường phát triển. Đây là điểm đặc sắc của Notebook so với các môi trường phát triển IDE khác. [3]

### Python

#### Giới thiệu về Python

Python được tạo ra bởi Guido Van Rossum vào năm 1989 tại Centrum Wiskunde & Informatica (CWI), một lập trình viên máy tính ở Hà Lan. Tên gọi của ngôn ngữ này được lấy cảm hứng từ chương trình truyền hình Monty Python's Flying Circus của đài BBC vì Guido Van Rossum là một “fan cứng” của chương trình này.



Hình 2.2 Logo Python

Python là một ngôn ngữ lập trình thông dịch, dễ đọc và dễ hiểu. Nền tảng nổi tiếng với cú pháp đơn giản và được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau. Python có cú pháp linh hoạt và cấu trúc dữ liệu mạnh mẽ, công nghệ được hỗ trợ bởi một cộng đồng lớn. Điều này đã mang đến các thư viện và framework phong phú mà người dùng có thể sử dụng để xây dựng các ứng dụng phức tạp. Python cũng là một trong những ngôn ngữ phổ biến cho người mới học lập trình nhờ vào tính linh hoạt của nó.

Thư viện là một tập hợp các mã thường xuyên được sử dụng mà các nhà phát triển có thể bao gồm trong những chương trình Python của họ để không phải lập trình từ đầu. Theo mặc định, Python đi kèm với Thư viện chuẩn, chứa rất nhiều các hàm có thể tái sử dụng. Ngoài ra, hơn 137.000 thư viện Python có sẵn cho các ứng dụng khác nhau, bao gồm phát triển web, khoa học dữ liệu và máy học (ML) [4]

#### Ứng dụng của Python

**Khoa học dữ liệu và Phân tích dữ liệu:** Python là công cụ chính cho các nhà khoa học dữ liệu. Thư viện Pandas cung cấp các công cụ mạnh mẽ để thao tác và phân tích dữ liệu. NumPy cho phép thực hiện các tính toán số học với hiệu suất cao. Matplotlib và Seaborn giúp tạo ra các biểu đồ đẹp mắt và dễ hiểu. Các nhà khoa học dữ liệu có thể dễ dàng làm việc với các tập dữ liệu lớn, thực hiện các phân tích phức tạp và trực quan hóa kết quả một cách sinh động.

**Trí tuệ Nhân tạo và Học máy:** Python là ngôn ngữ hàng đầu cho trí tuệ nhân tạo và học máy. TensorFlow và Keras giúp xây dựng các mô hình học sâu (deep learning) mạnh mẽ. PyTorch nổi bật với tính linh hoạt và dễ sử dụng, là lựa chọn ưa thích của nhiều nhà nghiên cứu. Scikit-learn cung cấp các công cụ đơn giản nhưng hiệu quả để thực hiện các thuật toán học máy truyền thống. Python giúp biến những ý tưởng phức tạp thành hiện thực với ít dòng mã hơn.

**Phát triển web:** Python thường được sử dụng để phát triển back-end của trang web hoặc ứng dụng, những phần mà người dùng không nhìn thấy. Vai trò của Python trong phát triển web có thể bao gồm gửi dữ liệu đến và đi từ máy chủ, xử lý dữ liệu và giao tiếp với cơ sở dữ liệu, định tuyến URL và đảm bảo tính bảo mật. Python cung cấp một số khuôn khổ để phát triển web. Những framework thường được sử dụng bao gồm Django và Flask. [5]

### PowerBI

#### Giới thiệu về Power BI

Power BI là một bộ tổng hợp các dịch vụ, ứng dụng và kết nối trong lĩnh vực phân tích dữ liệu giúp người dùng kết nối và chuyển đổi dữ liệu từ các nguồn không liên quan như bảng tính Excel hoặc một tập hợp các kho dữ liệu trên đám mây thành một tổ hợp thông tin trực quan và dễ tương tác, đồng thời dễ dàng chia sẻ chúng với các đối tượng khác. [6]



Hình 2.3 Logo Power BI

#### Tính năng chính của Power BI

Kết nối với nguồn dữ liệu đa dạng, từ những tệp dữ liệu phẳng (flat file), file excel, đến các cơ sở dữ liệu như SQL Server, MySQL, [Access](https://fptshop.com.vn/tin-tuc/thu-thuat/microsoft-access-la-gi-ly-do-ban-nen-su-dung-microsoft-access-145420),… hay dữ liệu trên đám mây (cloud). Ngoài ra, Power BI cũng có thể kết nối được với các nguồn dữ liệu đặc biệt như SharePoint, dữ liệu Web, PDF,… hay kết nối trực tiếp đến các hệ thống ERP, [SAP](https://fptshop.com.vn/tin-tuc/danh-gia/phan-mem-sap-la-gi-nhung-tinh-nang-va-loi-ich-khi-su-dung-146484),…

Làm sạch, biến đổi và chuẩn hóa dữ liệu đầu vào với Power Query Trước khi đưa vào báo cáo phân tích. Power BI Query có khả năng thực hiện nhiều thao tác phức tạp như lọc dữ liệu, xóa thông tin trùng lặp,...và trả về dữ liệu với độ chính xác cao, đầy đủ và phù hợp để phân tích.

Xây dựng các mô hình dữ liệu làm gọn nhẹ dữ liệu hơn so với việc phải kết hợp các dữ liệu thành một file duy nhất. Giống với các lược đồ dữ liệu trong các cơ sở dữ liệu, Power BI có khả năng kết nối các bảng để tạo thành một mô hình có tính liên kết giúp việc truy xuất dữ liệu được hiệu quả.

Trực quan hóa dữ liệu bằng các biểu đồ, dashboard mang tính tương tác một cách linh hoạt, từ đó có những phân tích theo nhiều chiều, từ tổng quan đến chi tiết để có các phát hiện, kết luận và đưa ra quyết định kinh doanh.

Hỗ trợ các hàm biểu thức tính toán phân tích dữ liệu DAX (data analysis expression) cực kì mạnh mẽ để xây dựng các công thức phân tích và hiển thị dữ liệu. [7]

## Machine learning

### Thuật toán Logistic Regression

#### Khái niệm

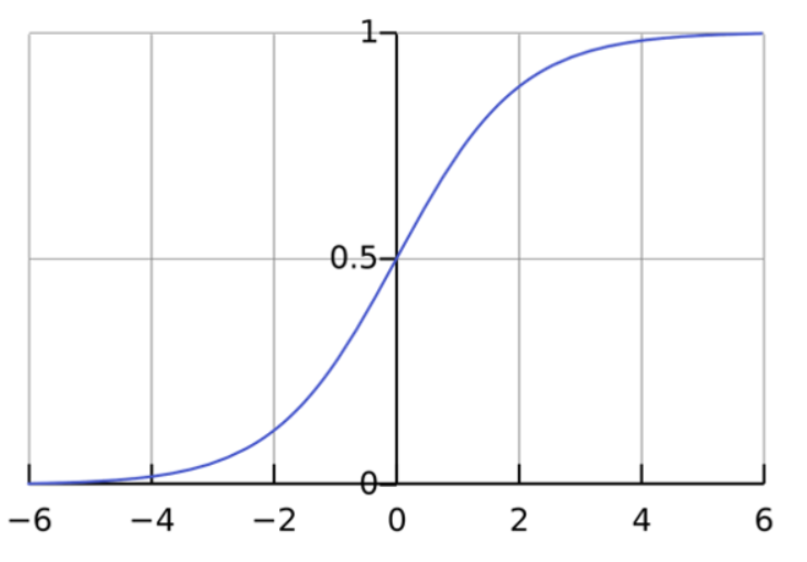
Hồi quy logistic là một thuật toán học máy được sử dụng rộng rãi cho các vấn đề liên quan đến phân loại. Ở dạng cơ bản, nó được sử dụng cho bài toán phân loại nhị phân chỉ có hai lớp để dự đoán. Tuy nhiên, với một chút mở rộng và chất xám, logistic regression có thể dễ dàng được sử dụng cho vấn đề phân loại nhiều lớp. [8]

#### Mô hình Logistic Regression

Mô hình hồi quy logistic sử dụng hàm sigmoid (logistic function) để ánh xạ tổ hợp tuyến tính của các biến đầu vào vào một giá trị nằm trong khoảng [0, 1], đại diện cho xác suất mà đầu ra thuộc lớp 1. Hàm sigmoid được biểu diễn bằng công thức như sau:

f(x) =

Nếu vẽ phương trình hồi quy logistic này, sẽ được một đường cong hình chữ S như hình dưới đây.



Hình 2.4 Phương trình Logistic Regression

Hàm logit chỉ trả về các giá trị giữa 0 và 1 cho biến phụ thuộc, dù giá trị của biến độc lập là gì. Đây là cách hồi quy logistic ước tính giá trị của biến phụ thuộc. Phương pháp hồi quy logistic cũng lập mô hình phương trình giữa nhiều biến độc lập và một biến phụ thuộc. [8]

#### Cách thức hoạt động của Logistic Regression

**Xác định câu hỏi**

Xác định mục tiêu nghiên cứu hoặc vấn đề cần giải quyết. Ví dụ, bạn có thể muốn dự đoán khả năng một khách hàng sẽ mua sản phẩm hay không dựa trên các biến số như tuổi, thu nhập, và hành vi mua sắm trước đây.

**Thu thập dữ liệu lịch sử**

Thu thập dữ liệu cần thiết bao gồm cả biến độc lập (các yếu tố dự đoán) và biến phụ thuộc (kết quả cần dự đoán). Dữ liệu này thường được thu thập từ các nguồn như cơ sở dữ liệu, khảo sát, hoặc các hệ thống quản lý thông tin.

**Chia tập dữ liệu**

Chia dữ liệu thành hai phần: tập huấn luyện và tập kiểm tra. Tập huấn luyện được sử dụng để xây dựng mô hình, trong khi tập kiểm tra được dùng để đánh giá hiệu suất của mô hình. Thông thường, dữ liệu được chia theo tỷ lệ 70-30 hoặc 80-20.

**Huấn luyện mô hình phân tích hồi quy**

Sử dụng tập huấn luyện để huấn luyện mô hình hồi quy logistic. Quá trình này bao gồm việc ước lượng các tham số của mô hình sao cho tối đa hóa khả năng dự đoán chính xác của mô hình.

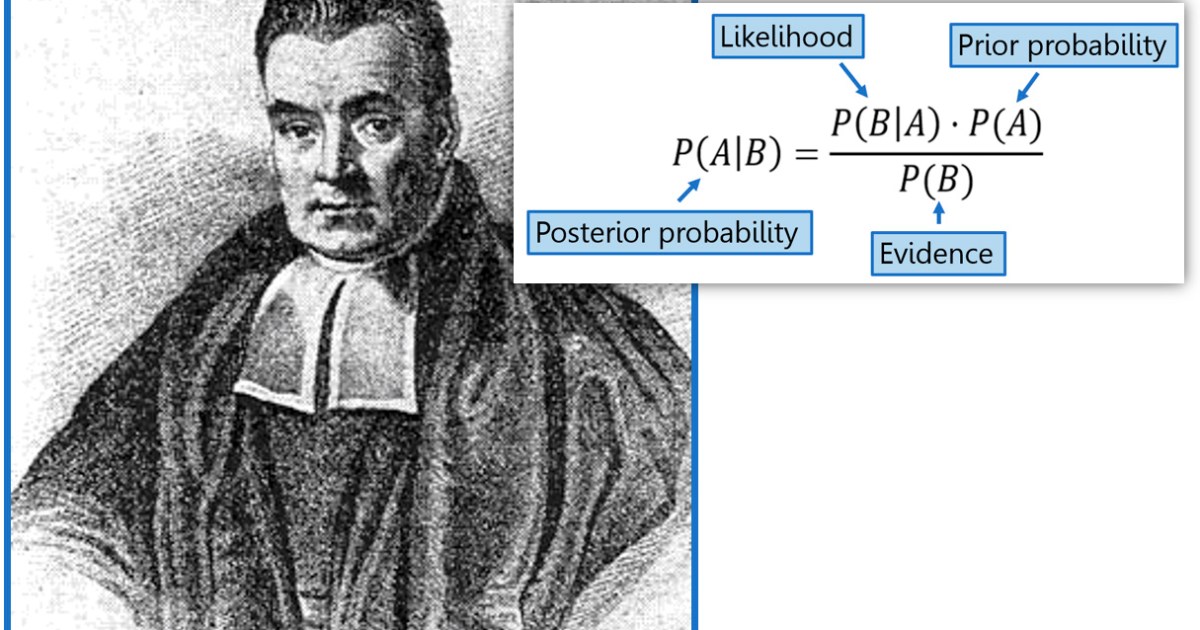
**Đưa ra dự đoán**

Sau khi mô hình được huấn luyện, sử dụng nó để dự đoán xác suất xảy ra của sự kiện trên các dữ liệu mới hoặc không xác định. Kết quả dự đoán thường là một giá trị xác suất từ 0 đến 1, và ta có thể sử dụng một ngưỡng xác định (ví dụ: 0.5) để phân loại kết quả thành các nhóm nhị phân. [8]

### Thuật toán Naïve Bayes

#### Khái niệm

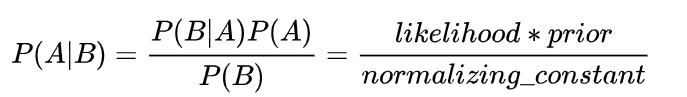
Thuật toán Naïve Bayes là một kỹ thuật phân loại thống kê dựa trên nguyên lý định lý Bayes về xác suất có điều kiện với một giả định rằng tất cả các yếu tố đặc trưng đầu vào là độc lập với nhau. Nói cách đơn giản, giả định là sự hiện diện của một đối tượng trong một lớp là độc lập với sự hiện diện của bất kỳ đối tượng nào khác trong cùng một lớp.



Hình 2.5 Thuật toán Naïve Bayes

Định lý Bayes là một kết quả của lý thuyết xác suất, phản ánh mối quan hệ giữa xác suất của một biến cố mà không quan tâm các yếu tố khác (gọi là xác suất biên hay xác suất tiền nghiệm) với xác suất của biến cố đó sau khi một biến cố khác đã xảy ra (gọi là xác suất có điều kiện hay xác suất hậu nghiệm). Định lý này cho phép tính xác suất của một sự kiện dựa trên thông tin đã biết trước.

Công thức của định lý Bayes như sau



Hình 2.6 Công thức thuật toán Naïve Bayes

Cụ thể hơn về công thức này thì Định lý Bayes cho phép tính xác suất xảy ra của một sự kiện ngẫu nhiên A khi biết sự kiện liên quan B đã xảy ra ( xác suất có điều kiện). Xác suất này được ký hiệu là P(A|B), và đọc là "xác suất của A nếu có B". Xác suất xảy ra sẽ phụ thuộc vào 3 yếu tố:

Xác suất xảy ra A của riêng nó, không quan tâm đến B. Ký hiệu là P(A) và đọc là xác suất xảy ra của A. Đây được gọi là xác suất biên duyên hay xác suất tiên nghiệm theo nghĩa rằng nó không quan tâm đến bất kỳ thông tin nào về B.

Xác suất xảy ra B của riêng nó, không quan tâm đến A. Ký hiệu là P(B) và đọc là "xác suất xảy ra của B". Đại lượng này còn gọi là Hằng số chuẩn (normalising constant), vì nó luôn giống nhau, không phụ thuộc vào sự kiện A đang muốn biết.

Xác suất xảy ra B khi biết A xảy ra. Ký hiệu là P(B|A) và đọc là "xác suất của B nếu điều kiện giả thuyết A đúng ". Đại lượng này gọi là khả năng (likelihood) xảy ra B khi biết A đã xảy ra. Chú ý không nhầm lẫn giữa khả năng xảy ra B khi biết A và xác suất xảy ra A khi biết B. [9]

#### Cách thức hoạt động của Naïve Bayes

Trước hết, chuẩn bị và làm sạch dữ liệu là bước đầu tiên. Trong quá trình này sẽ loại bỏ các từ hoặc ký hiệu không cần thiết khỏi văn bản, chuẩn bị dữ liệu đầu vào cho mô hình.

Tiếp theo, trích xuất các đặc trưng (feature) từ dữ liệu và tạo ra ma trận nhãn tương ứng. Đây là những thông tin quan trọng mà mô hình sẽ sử dụng để thực hiện quá trình học và dự đoán. Đầu tiên, tính toán xác suất cho từng lớp. Cụ thể là tính xác suất tiền nghiệm của từng lớp dựa trên tần suất xuất hiện của từng lớp trong tập dữ liệu huấn luyện. Tiếp theo, tính xác suất của từng đặc trưng khi biết lớp. Sau đó, áp dụng giả định "naive" - giả định rằng các đặc trưng đầu vào là độc lập với nhau. Điều này giúp đơn giản hóa tính toán và giảm độ phức tạp của mô hình. Dựa trên các xác suất đã tính, sử dụng công thức Bayes để tính xác suất của mẫu dữ liệu cho từng lớp. Đây là xác suất hậu nghiệm, phản ánh xác suất một mẫu dữ liệu thuộc về từng lớp.

Sử dụng thư viện sklearn Naive Bayes để training mô hình và dự đoán. Sklearn Naive Bayes cung cấp ba lựa chọn tham số cho mô hình: Gaussian, Multinomial và Bernoulli. Tùy thuộc vào đặc điểm của dữ liệu, lựa chọn tham số phù hợp nhất. Cuối cùng, đánh giá mô hình bằng cách so sánh độ chính xác của dự đoán với kết quả thực tế qua các chỉ số đánh giá mô hình. [9]

### Thuật toán Support Vector Machine

#### Khái niệm

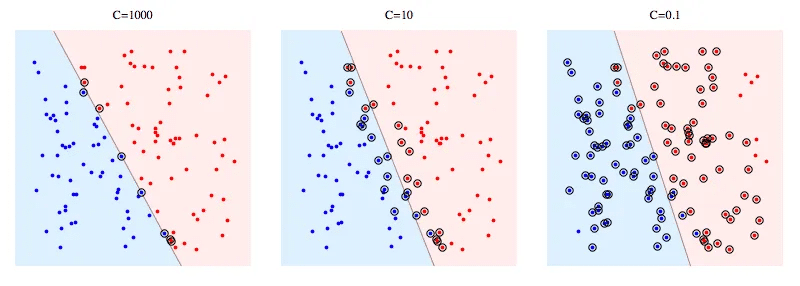
SVMlà một thuật toán học máy được sử dụng trong bài toán phân loại và hồi quy, được sử dụng nhiều nhất trong học máy trước khi mạng nơ ron nhân tạo trở lại với các mô hình deep learning. SVM chỉ có thể hoạt động trên dữ liệu tuyến tính. Mục tiêu của thuật toán này là tìm cách xác định một siêu phẳng trong không gian N chiều (ứng với N đặc trưng) để phân chia các điểm dữ liệu thành các lớp . Mục tiêu của SVM là tìm một siêu phẳng có khoảng cách lớn nhất với các điểm dữ liệu gần nhất của các lớp tương ứng với lớp của chúng. (Nghĩa là mục tiêu của chúng ta là tìm siêu phẳng có lề hay khoảng cách tới các điểm của 2 lớp là lớn nhất). [10]

#### Các mở rộng của SVM

**Soft Margin SVM**

Là một biến thể của SVM, sử dụng trong trường hợp dữ liệu được phân tách tuyến tính, khi chúng ta không thể vẽ được đường thẳng để phân loại các điểm dữ liệu. Thuật toán này cho phép SVM mắc một số lỗi nhất định với mục tiêu giữ cho lề càng rộng càng tốt và các điểm khác vẫn được phân loại chính xác. Nói cách khác, nó sẽ cân bằng giữa việc phân loại sai và tối đa hóa lề. Sự cho phép vi phạm một số điểm dữ liệu giúp giảm thiểu overfitting và làm cho mô hình SVM linh hoạt hơn trong việc xử lý các tập dữ liệu có độ phức tạp cao hơn.

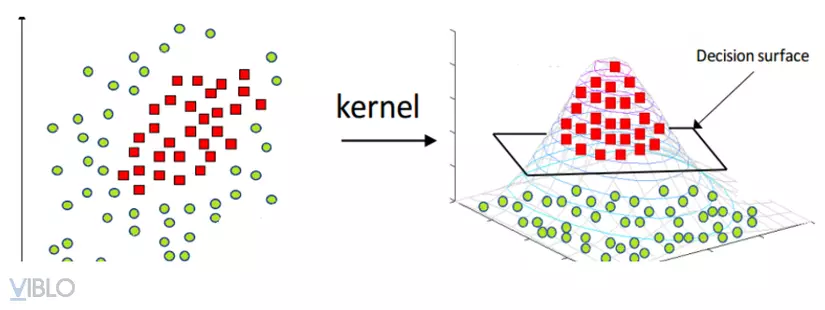
Một đại lượng quan trọng và để kiểm soát mức độ sai sót của Soft Margin SVM là mức độ chấp nhận lỗi (hay tham số phạt). Khi tham số phạt càng lớn nghĩa là SVM bị phạt càng nặng khi phân loại sai và kết quả sẽ cho ra 1 lề ( margin) hẹp và ngược lại. Hình dưới thể hiện [SVM](https://trituenhantao.io/tu-dien-thuat-ngu/svm/) với các giá trị C khác nhau. [10]



Hình 2.7 Tham số phạt C

**Kernel SVM**

Là một phiên bản mở rộng của SVM cho phép xử lý các tập dữ liệu không tuyến tính bằng cách sử dụng hàm nhân (kernel function). Thay vì tìm một siêu phẳng trong không gian ban đầu, Kernel SVM ánh xạ dữ liệu vào không gian nhiều chiều hơn thông qua hàm nhân và sau đó tìm một siêu phẳng phân chia tuyến tính trong không gian mới. Các hàm nhân phổ biến được sử dụng bao gồm hàm đa thức (polynomial kernel), hàm Radial Basis Function (RBF kernel), và hàm sigmoid. Hình dưới đây sẽ minh họa về sự tiện lợi trong việc tìm kiếm siêu phẳng (hyperplane) khi chuyển đổi dữ liệu từ 2 chiều (dimension) sang 3 chiều (dimension) sử dụng Polynomial kernel. [10]



Hình 2.8 Kernel trick

### Thuật toán Neural Network

#### Khái niệm

Neural Network đọc tiếng việt là Mạng nơ-ron nhân tạo, đây là một chuỗi những thuật toán được đưa ra để tìm kiếm các mối quan hệ cơ bản trong tập hợp các dữ liệu. Thông qua việc bắt bước cách thức hoạt động từ não bộ của con người.  Hiểu theo cách đơn giản hơn, mạng nơ-ron nhân tạo được xem là hệ thống của các tế bào thần kinh nhân tạo và mạng nơ-ron có khả năng thích ứng được với mọi thay đổi từ đầu vào.



Hình 2.9 Hình minh họa Neural Network

Các loại mạng Nơ-ron gồm các loại như:

Mạng Nơ-ron Truyền Thẳng (Feedforward Neural Networks - FNNs): Loại đơn giản nhất, nơi các kết nối không tạo thành vòng lặp.

Mạng Nơ-ron Chập (Convolutional Neural Networks - CNNs): Chuyên xử lý dữ liệu dạng lưới như hình ảnh.

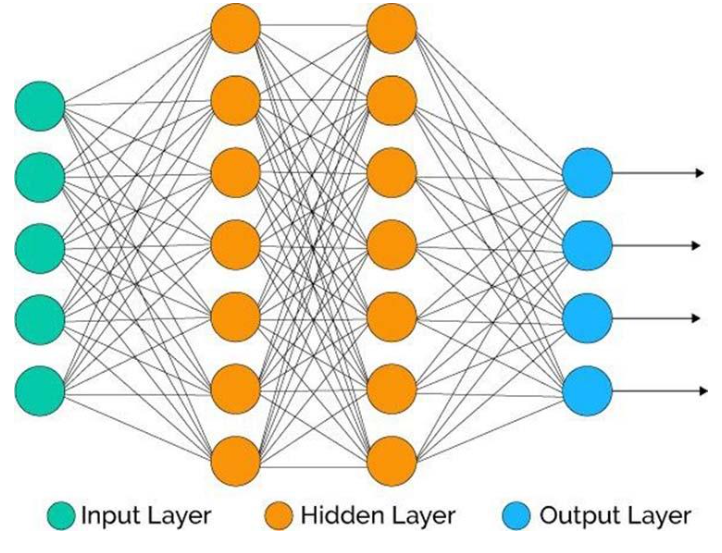
Mạng Nơ-ron Tái Hiện (Recurrent Neural Networks - RNNs): Thiết kế cho dữ liệu tuần tự, nơi các kết nối tạo thành vòng lặp có hướng.

Mạng LSTM và GRU: Các biến thể của RNNs giải quyết vấn đề gradient biến mất, làm cho chúng hiệu quả cho các phụ thuộc dài hạn.

Mạng GANs (Generative Adversarial Networks): Gồm hai mạng, một mạng sinh (generator) và một mạng phân biệt (discriminator), cạnh tranh để tạo ra dữ liệu thực tế. [11]

#### Kiến trúc của mạng Neural Network

Mạng Nơ-ron là sự kết hợp của các tầng Perceptron hay còn gọi là Perceptron đa tầng. Đơn vị cơ bản của mạng nơ-ron là nơ-ron, còn được gọi là nút hoặc đơn vị. Các nơ-ron được tổ chức thành các lớp. Mỗi nơ-ron nhận một hoặc nhiều đầu vào, xử lý chúng và tạo ra một đầu ra.



Hình 2.10 Minh họa số lớp của Neural Network

Và mỗi một mạng Neural Network thường bao gồm 3 kiểu lớp là:

Lớp input layer (Lớp vào): Tầng này nằm bên trái cùng của mạng, thể hiện cho các đầu vào của mạng.

Lớp output layer (Lớp ra): Là tầng bên phải cùng và nó thể hiện cho những đầu ra của mạng.

Lớp hidden layer (Lớp ẩn): Tầng này nằm giữa tầng vào và tầng ra nó thể hiện cho quá trình suy luận logic của mạng.

Các nơ-ron trong một lớp được kết nối với các nơ-ron trong lớp tiếp theo thông qua các kết nối có trọng số. Mỗi kết nối có một trọng số liên quan xác định độ mạnh và hướng của ảnh hưởng giữa các nơ-ron. [11]

#### Các hàm trong mạng Neural

Trong mạng Nơ-ron sẽ có những hàm phổ biến như: hàm kích hoạt, hàm mất mát, hàm tối ưu hóa, hàm khởi tạo trọng số.

Hàm kích hoạt quyết định đầu ra của một nơ-ron dựa trên đầu vào của nó. Chúng thêm tính phi tuyến tính vào mô hình, giúp mạng học được các mối quan hệ phức tạp trong dữ liệu bao gồm:

**Sigmoid:** Biến đầu vào thành giá trị giữa 0 và 1 và thường được sử dụng trong lớp đầu ra cho các bài toán phân loại nhị phân.

**Tanh:** Biến đầu vào thành giá trị giữa -1 và 1 và thường được sử dụng trong các lớp ẩn.

**ReLU:** Trả về x nếu x > 0, ngược lại trả về 0 và rất phổ biến trong các lớp ẩn vì tính hiệu quả và đơn giản.

**Leaky ReLU:** Giống ReLU nhưng cho phép một số giá trị âm nhỏ (0.01x thay vì 0).

**Softmax:** Thường được sử dụng trong lớp đầu ra cho các bài toán phân loại nhiều lớp và trả về xác suất cho mỗi lớp.

Hàm mất mát đo lường sự khác biệt giữa giá trị dự đoán và giá trị thực tế, giúp hướng dẫn quá trình học của mạng.

Các hàm mất mát (loss functions) là thành phần quan trọng trong quá trình huấn luyện mạng nơ-ron, giúp đo lường sự khác biệt giữa giá trị dự đoán và giá trị thực tế, từ đó hướng dẫn quá trình học của mạng. Trong hàm mất mát sẽ gồm:

**MSE:** Đo lường độ lệch bằng bình phương sai số, nhạy cảm với outliers, dùng cho hồi quy.

**MAE:** Đo lường độ lệch bằng giá trị tuyệt đối của sai số, ít nhạy cảm với outliers, dùng cho hồi quy.

**Cross-Entropy Loss:** Đo lường sự không khớp giữa phân phối xác suất dự đoán và thực tế trong phân loại nhị phân.

**Categorical Cross-Entropy:** Đo lường sự không khớp giữa phân phối xác suất dự đoán và thực tế trong phân loại nhiều lớp. [11]

### Confusion Matrix

Ma trận nhầm lẫn là ma trận tóm tắt hiệu suất của mô hình học máy trên một tập dữ liệu thử nghiệm. Đây là phương tiện hiển thị số lượng trường hợp chính xác và không chính xác dựa trên dự đoán của mô hình. Ma trận này thường được sử dụng để đo hiệu suất của các mô hình phân loại, nhằm mục đích dự đoán nhãn phân loại cho từng trường hợp đầu vào. [12]Confusion matrix bao gồm 4 chỉ số như sau:

True Positive (TP): Khi giá trị thực tế là Positive và được dự đoán đúng là Positive.

True Negative (TN): Khi giá trị thực tế là Negative và được dự đoán đúng là Negative.

False Positive (FP): Khi giá trị thực tế là Negative nhưng được dự đoán là Positive. Còn được gọi là Type 1 error

False Negative (FN): Khi giá trị thực tế là Positive nhưng được dự đoán là Negative. Còn được gọi là Type 2 error

A diagram of positive and negative

Description automatically generated

Hình 2.11 Confusion Matrix

Để đánh giá hiệu suất mô hình một cách toàn diện hơn, Confusion matrix sử dụng các chỉ số đánh giá sau:

Độ chính xác đo lường tỷ lệ dự đoán đúng trong tổng số dự đoán. Nó cho biết mô hình của bạn đúng bao nhiêu phần trăm khi dự đoán tất cả các trường hợp.

Accuracy =

Độ nhạy đo lường tỷ lệ các trường hợp dương tính thực sự được mô hình nhận diện chính xác. Nó cho biết khả năng mô hình phát hiện các trường hợp dương tính.

Sensitivity =

Độ đặc hiệu đo lường tỷ lệ các trường hợp âm tính thực sự được mô hình nhận diện chính xác. Nó cho biết khả năng mô hình loại trừ các trường hợp âm tính.

Specificity =

Độ chính xác của các dự đoán dương tính đo lường tỷ lệ các dự đoán dương tính thực sự đúng. Nó cho biết trong số các trường hợp được dự đoán là dương tính, có bao nhiêu trường hợp thực sự là dương tính.

Precision =

Điểm F1 là trung bình giữa độ chính xác và độ nhạy. Nó cung cấp một thước đo cân bằng giữa Precision và Recall, đặc biệt hữu ích khi bạn có một phân phối không đồng đều giữa các lớp.

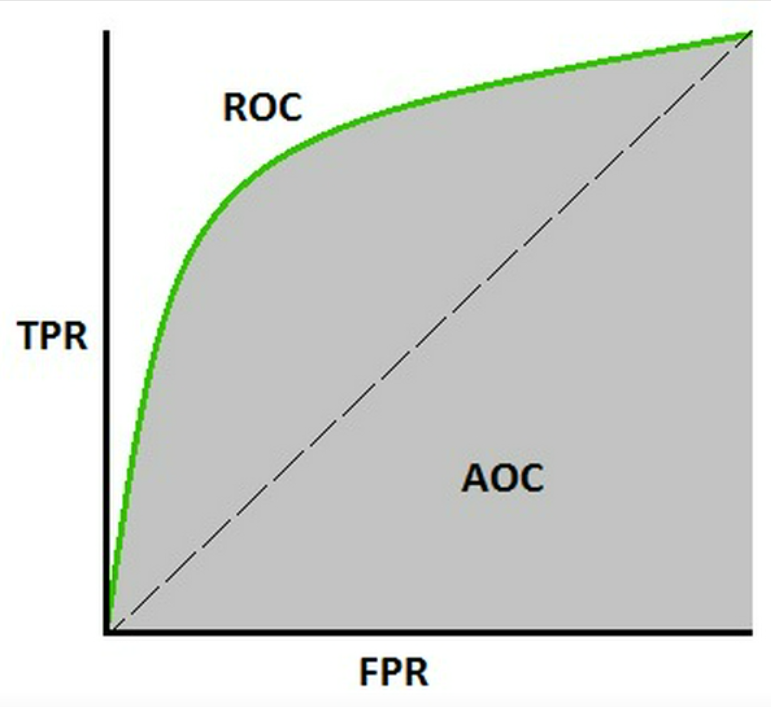
F1-score = 2 \*

### Ý nghĩa đường cong AUC và Roc

Trong các bài toán phân loại, ngoài việc sử dụng ma trận Confusion Matrix - một phương pháp phổ biến, chúng ta cũng không thể không nhắc đến một phương pháp quan trọng khác: đường cong AUC-ROC dung để tính toán hiệu suất mô hình.

AUC - ROC là một phương pháp tính toán hiệu suất của một mô hình phân loại theo các ngưỡng phân loại khác nhau. ROC là một đường cong biểu diễn xác suất và AUC biểu diễn mức độ phân loại của mô hình.  Ý nghĩa của AUC - ROC có thể diễn giải như sau: Là xác suất rằng một mẫu dương tính được lấy ngẫu nhiên sẽ được xếp hạng cao hơn một mẫu âm tính được lấy ngẫu nhiên.

Đường cong ROC biểu diễn các cặp chỉ số (TPR, FPR) tại mỗi ngưỡng với TPR là trục tục và FPR là trục hoành.



Hình 2.12 Minh họa ROC

TPR(True Positive Rate/Sentivity/Recall): là chỉ số biểu diễn tỷ lệ phân loại chính xác các mẫu dương tính trên tất cả các mẫu dương tính và TPR càng cao thì các mẫu dương tính càng được phân loại chính xác. Nó được tính theo công thức:

TPR/ Sentivity/ Recall =

Specificity: Biểu diễn tỷ lệ phân loại chính xác các mẫu âm tính trên tất cả các mâu âm tính, được tính theo công thức:

Specificity =

FPR(False Positive Rate/Fall-out): Biểu diễn tỷ lệ gắn nhãn sai các mẫu âm tính thành dương tính trên tất cả các mẫu âm tính, được tính theo công thức:

FPR = Specificity -1=

Từ đó ta thấy rằng  Specificity tỷ lệ nghịch với FPR. FPR càng cao thì Specificity càng giảm và số lượng các mẫu âm tính bị gắn nhãn sai càng lớn.

Đây chính là các chỉ số dùng để tính toán hiệu suất phân loại của mô hình. Để hợp chúng lại thành 1 chỉ số duy nhất, ta sử dụng đường cong ROC để hiển thị từng cặp (TPR, FPR) cho các ngưỡng khác nhau với mỗi điểm trên đường cong biểu diễn 1 cặp (TPR, FPR) cho 1 ngưỡng, sau đó tính chỉ số AUC cho đường cong này. Chỉ số AUC chính là con số thể hiện hiệu suất phân loại của mô hình. Chỉ số AUC càng gần 1 thì mô hình phân loại càng chính xác, còn nếu AUC càng gần 0,5 thì hiệu suất phân loại mô hình càng tệ và càng về 0 thì mô hình sẽ bị phân loại ngược với kết quả (phân loại dương tính thành âm tính và ngược lại). [13]

# PHÂN TÍCH KHÁM PHÁ VÀ XỬ LÝ DỮ LIỆU

## Giới thiệu dữ liệu

Bộ dữ liệu mẫu sử dụng được trích xuất từ tập dữ liệu nhân viên của công ty IBM từ năm 2021

A table of numbers and letters

Description automatically generated

Hình 3.1 Bộ dữ liệu đầu vào

Tập dữ liệu gồm 35 trường và 1470 bản ghi

Bảng 3.1 Mô tả trường dữ liệu

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Mô tả** | **Kiểu dữ liệu** |
| Age | Tuổi của nhân viên. | int 64 |
| Attrition | Tình trạng nghỉ việc | object |
| BusinessTravel | Mức độ và tần suất công tác của nhân viên. | object |
| DailyRate | Số tiền mà nhân viên được trả mỗi ngày. | int64 |
| Department | Phòng ban nhân viên làm việc | object |
| DistanceFromHome | Khoảng cách từ nhà đến nơi làm việc của nhân viên. | int64 |
| Education | Trình độ học vấn của nhân viên | int64 |
| EducationField | Ngành học của nhân viên | object |
| EmployeeCount | Số lượng nhân viên | int64 |
| EmployeeNumber | Mã số duy nhất của nhân viên. | int64 |
| EnvironmentSatisfaction | Mức độ hài lòng của nhân viên với môi trường làm việc | int64 |
| Gender | Giới tính | object |
| HourlyRate | Số tiền mà nhân viên được trả mỗi giờ. | int64 |
| JobInvolvement | Mức độ tham gia và cam kết của nhân viên với công việc | int64 |
| JobLevel | Cấp bậc hoặc vị trí công việc của nhân viên | int64 |
| JobRole | Vai trò hoặc chức vụ của nhân viên | object |
| JobSatisfaction | Mức độ hài lòng của nhân viên với công việc | int64 |
| MaritalStatus | Tình trạng hôn nhân | object |
| MonthlyIncome | Số tiền mà nhân viên được trả mỗi tháng. | int64 |
| MonthlyRate | Mức lương danh nghĩa hoặc tỷ lệ lương | int64 |
| NumCompaniesWorked | Số lượng công ty mà nhân viên đã từng làm việc trước đây. | int64 |
| Over18 | Trên 18 tuổi | object |
| OverTime | Làm thêm giờ | object |
| PercentSalaryHike | Tỷ lệ phần trăm tăng lương hàng năm của nhân viên. | int64 |
| PerformanceRating | Đánh giá hiệu suất | int64 |
| RelationshipSatisfaction | Mức độ hài lòng của nhân viên với các mối quan hệ trong công ty | int64 |
| StandardHours | Giờ làm việc tiêu chuẩn | int64 |
| StockOptionLevel | Cấp độ quyền chọn cổ phiếu | int64 |
| TotalWorkingYears | Tổng số năm làm việc | int64 |
| TrainingTimesLastYear | Số lần nhân viên tham gia đào tạo trong năm ngoái. | int64 |
| WorkLifeBalance | Đánh giá mức độ cân bằng giữa công việc và cuộc sống của nhân viên | int64 |
| YearsAtCompany | Số năm làm việc tại công ty hiện tại. | int64 |
| YearsInCurrentRole | Số năm làm việc trong vai trò hiện tại. | int64 |
| YearsSinceLastPromotion | Số năm kể từ lần thăng chức cuối cùng | int64 |
| YearsWithCurrManager | Số năm làm việc với quản lý hiện tại | int64 |

## Tiền xử lý dữ liệu

### Import thư viện

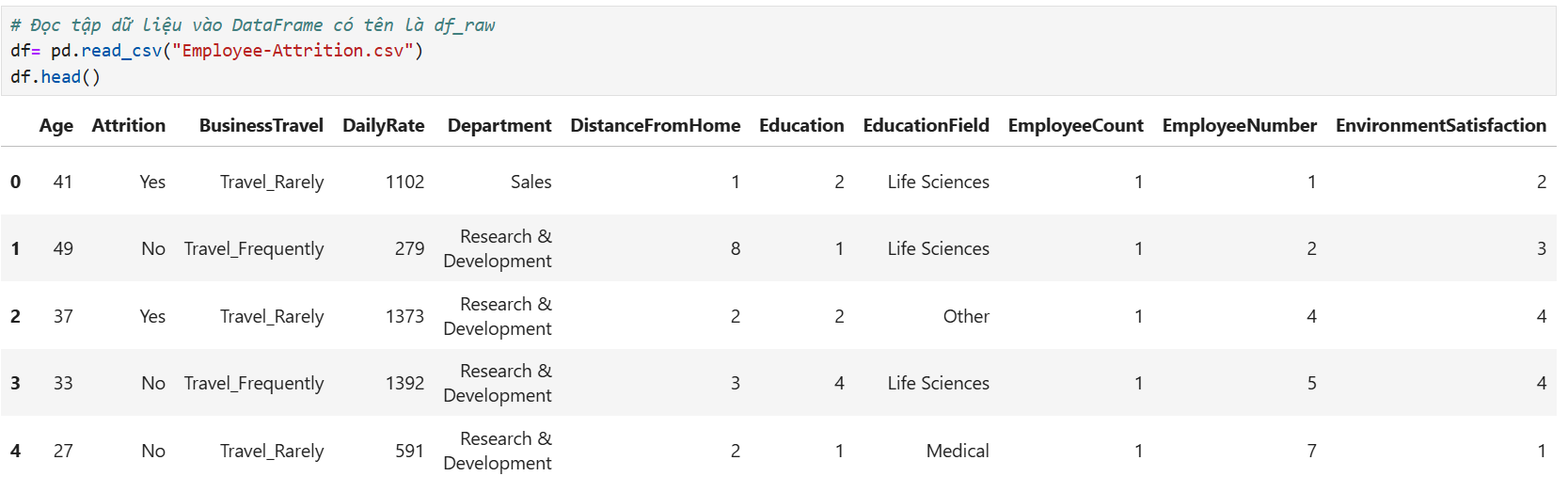
A white rectangular object with a white border

Description automatically generated

Hình 3.2 Import thư viện

Để thực hiện việc kiểm tra mẫu dữ liệu một cách chính xác và thực hiện các lệnh phân tích dữ liệu, ta thực hiện các bước bằng công cụ Jupyter Notebook và thực thi mã lập trình Python. Trước tiên, ta thực hiện thiết lập cấu hình hiển thị của pandas DataFrame. Việc thiết lập sẽ hỗ trợ việc xem toàn bộ dữ liệu mà không bị cắt ngắn khi dữ liệu lớn và nó sẽ đảm bảo việc phát hiện giá trị bất thường trong dữ liệu.

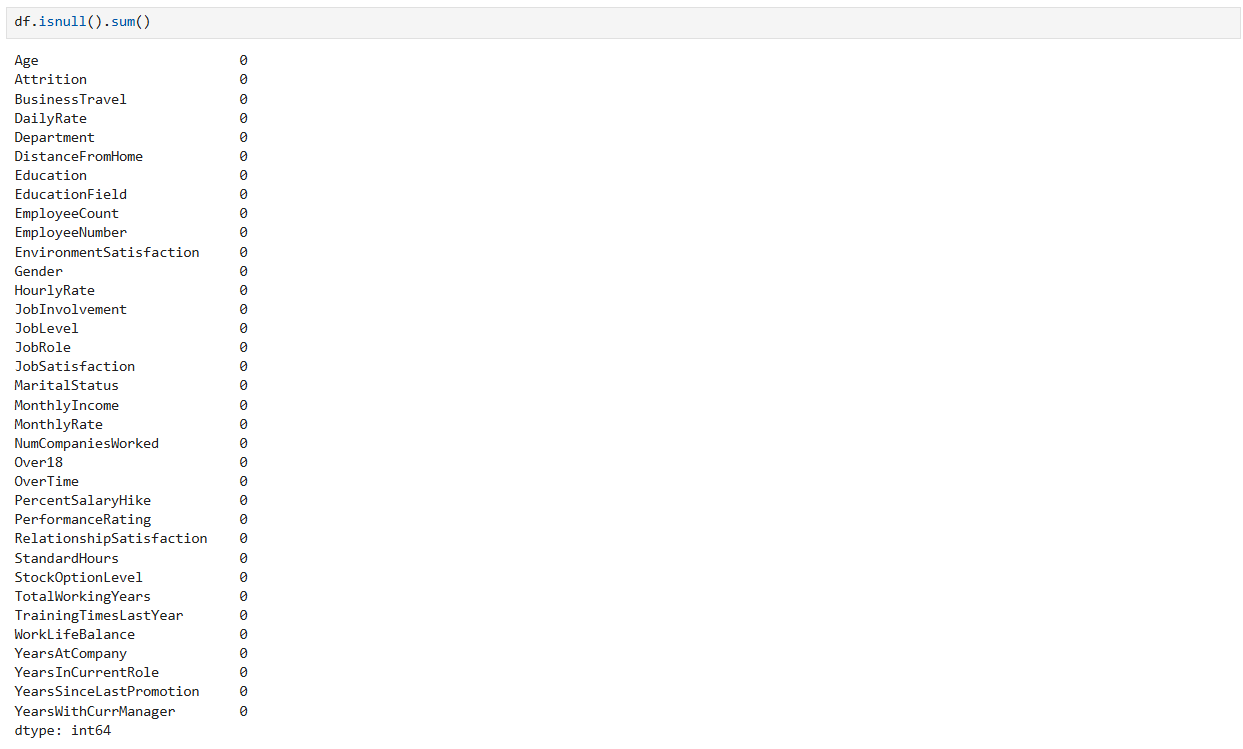
### Đọc dữ liệu vào



Hình 3.3 Đọc dữ liệu đầu vào

Ta thực hiện việc đọc tập dữ liệu vào DataFrame có tên là “df” và dùng lệnh “data.head” hiển thị các dòng đầu tiên của dữ liệu để hiểu rõ hơn về thông tin các dòng các cột của bộ dữ liệu.

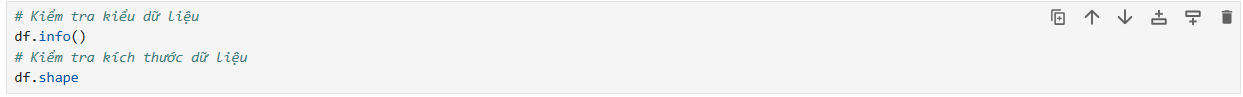
### Xử lý các giá trị Null



Hình 3.4 Xử lý giá trị Null

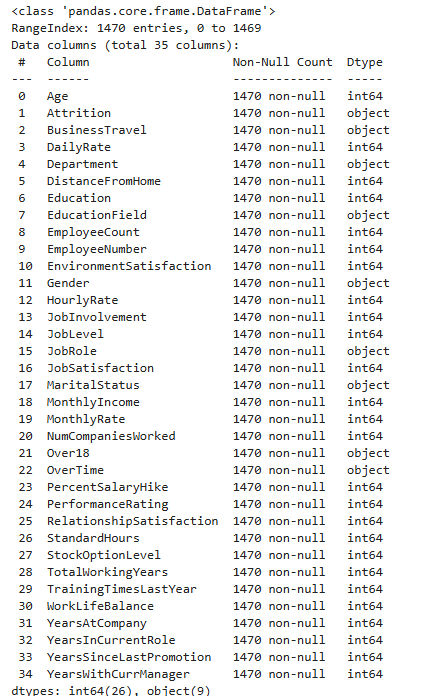
## Phân tích khám phá dữ liệu

### Kiểm tra kích thước và kiểu dữ liệu



Hình 3.5 Code kiểm tra kích thước và kiểu dữ liệu

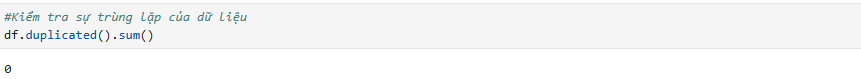
Trong quá trình phân tích khám phá dữ liệu, ta sử dụng lệnh “df.shape” kiểm tra số hàng, số cột có trong tập dữ liệu một cách chính xác nhất. Sau khi thực hiện lệnh ta thu được kết quả số hàng là 1470 và số cột là 35 tương ứng.



Hình 3.6 Kết quả kiểu dữ liệu

Để hiểu rõ hơn về bộ dữ liệu chúng ta sẽ sử dụng câu lệnh “df.info” để hiển thị các kiểu dữ liệu của các cột trong bộ dữ liệu. Kết quả sau khi sử dụng câu lệnh “df.info” cho ta biết được có 9 cột với kiểu dữ liệu là object và 26 cột có kiểu dữ liệu là int64.

### Kiểm tra sự trùng lặp của dữ liệu



Hình 3.7 Code kiểm tra sự trùng lặp

Trước khi tiến hành xử lý dữ liệu, chúng ta thực hiện việc kiểm tra các dữ liệu trùng lặp. Kết quả trả về là 0 cho thấy không có hàng nào trong DataFrame “df” bị trùng lặp. Nó là một kết quả tích cực trong việc xử lý dữ liệu và giúp đảm bảo tính toàn vẹn cho dữ liệu, tránh sai lệch và tiết kiệm thời gian phân tích dữ liệu

### Kiểm tra các cột có giá trị duy nhất

A screenshot of a computer

Description automatically generated

Hình 3.8 Xem số giá trị của cột

Sau khi kiểm tra không có giá trị trùng lặp chúng ta tiến hành kiểm tra các giá trị duy nhất của mỗi cột tức là kiểm tra xem trong 35 cột có những cột nào có giá trị không thay đổi. Nếu có sẽ tiến hành xóa bỏ các cột liệu để giảm bớt kích thước của tập dữ liệu và tăng hiệu suất cho các phân tích dữ liệu về sau.

Để kiểm tra các giá trị duy nhất ta sử dụng câu lệnh “df.unique” kết quả cho ta thấy được rằng các cột có giá trị không đổi gồm các cột “EmployeeCount, EmployeeNumber, Over18, StandardHour”

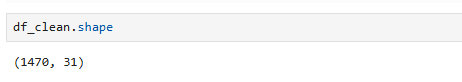
### Xử lý các cột có giá trị không đổi

Vì không có sự biến đổi giá trị nào trong các cột “EmployeeCount, EmployeeNumber, Over18, StandardHour, ta quyết định loại bỏ chúng khỏi tập dữ liệu để giảm bớt kích thước của tập dữ liệu và tăng hiệu suất cho các phân tích dữ liệu về sau.



Hình 3.9 Xóa cột không có ý nghĩa

Sau khi xóa bỏ 4 cột ở trên, thực hiện kết quả kiểm tra lại dữ liệu, ta thu được 31 cột dữ liệu và 1470 bản ghi.



Hình 3.10 Kết quả dữ liệu mới

### Phân tích thống kê mô tả

#### Thống kê mô tả

A screenshot of a computer

Description automatically generated

A screenshot of a data

Description automatically generated

A screenshot of a computer

Description automatically generated

Hình 3.11 Chỉ số thống kê mô tả của dữ liệu

Dựa vào các chỉ số thống kê mô tả có thể nhận xét sau:

Tuổi (Age): Tuổi trung bình của nhân viên là khoảng 37, với độ lệch chuẩn tương đối cao (9.14), cho thấy sự phân bố tuổi đa dạng trong công ty, từ 18 đến 60 tuổi. Các phân vị cho thấy phần lớn nhân viên nằm trong khoảng tuổi từ 30 đến 43.

Mức lương hàng ngày (DailyRate): Mức lương hàng ngày trung bình là 802.49, với độ lệch chuẩn cao (403.51), cho thấy sự chênh lệch lớn trong mức lương hàng ngày, từ 102 đến 1499.

Khoảng cách từ nhà đến công ty (DistanceFromHome): Khoảng cách trung bình từ nhà đến công ty là 9.19 km, với độ lệch chuẩn cao (8.11), cho thấy sự khác biệt lớn về khoảng cách đi làm của nhân viên, từ 1 đến 29 km.

Trình độ học vấn (Education): Trình độ học vấn trung bình là gần 3, với độ lệch chuẩn tương đối thấp (1.02), cho thấy đa số nhân viên có trình độ học vấn từ trung bình đến cao.

Số lượng nhân viên (EmployeeCount): Chỉ số này không thay đổi (giá trị duy nhất là 1), cho thấy mỗi hàng ghi nhận là một nhân viên.

Số hiệu nhân viên (EmployeeNumber): Số hiệu nhân viên cho thấy sự phân bố rộng (từ 1 đến 2068), nhưng chỉ số này thường chỉ để nhận diện và không mang nhiều ý nghĩa thống kê.

Sự hài lòng về môi trường làm việc (EnvironmentSatisfaction): Sự hài lòng trung bình là 2.72, cho thấy đa số nhân viên cảm thấy hài lòng ở mức trung bình đến cao.

Mức lương theo giờ (HourlyRate): Mức lương theo giờ trung bình là 65.89, với sự phân bố khá rộng (30 đến 100), cho thấy sự chênh lệch trong mức lương theo giờ giữa các nhân viên.

Mức độ tham gia công việc (JobInvolvement): Mức độ tham gia công việc trung bình là 2.73, cho thấy đa số nhân viên có mức độ tham gia công việc từ trung bình đến cao.

Cấp độ công việc (JobLevel): Cấp độ công việc trung bình là 2.06, cho thấy đa số nhân viên ở các cấp độ công việc từ 1 đến 3.

Mức độ hài lòng về công việc (JobSatisfaction): Mức độ hài lòng trung bình là 2.73, cho thấy đa số nhân viên cảm thấy hài lòng ở mức trung bình đến cao.

Thu nhập hàng tháng (MonthlyIncome): Thu nhập hàng tháng trung bình là 6502.93, với độ lệch chuẩn rất cao (4707.96), cho thấy sự chênh lệch lớn trong thu nhập hàng tháng của nhân viên, từ 1009 đến 19999.

Tỷ lệ hàng tháng (MonthlyRate): Tỷ lệ hàng tháng trung bình là 14313.10, với sự phân bố rộng (2094 đến 26999), cho thấy sự khác biệt lớn giữa các mức lương hàng tháng.

Số công ty đã làm việc (NumCompaniesWorked): Số công ty đã làm việc trung bình là 2.69, với độ lệch chuẩn cao (2.50), cho thấy sự khác biệt lớn về kinh nghiệm làm việc trước đây của nhân viên.

Tỷ lệ tăng lương theo phần trăm (PercentSalaryHike): Tỷ lệ tăng lương trung bình là 15.21%, cho thấy đa số nhân viên có tỷ lệ tăng lương từ 12% đến 18%.

Đánh giá hiệu suất (PerformanceRating): Đánh giá hiệu suất trung bình là 3.15, cho thấy đa số nhân viên được đánh giá ở mức 3 hoặc 4.

Sự hài lòng về mối quan hệ (RelationshipSatisfaction): Sự hài lòng trung bình là 2.71, cho thấy đa số nhân viên cảm thấy hài lòng ở mức trung bình đến cao về các mối quan hệ.

Dựa trên các chỉ số thống kê mô tả, các yếu tố chính ảnh hưởng đến việc rời bỏ của nhân viên bao gồm mức lương, khoảng cách từ nhà đến công ty, sự hài lòng về công việc và môi trường làm việc, tỷ lệ tăng lương, số công ty đã làm việc, tuổi, mối quan hệ với đồng nghiệp và cấp trên, và mức độ tham gia công việc. Sự chênh lệch lớn trong mức lương và thu nhập hàng tháng, khoảng cách đi làm xa, sự không hài lòng với công việc hoặc môi trường làm việc, và mối quan hệ không tốt đẹp với đồng nghiệp có thể dẫn đến cảm giác bất mãn và quyết định rời bỏ công ty. Công ty cần quan tâm đến những yếu tố này để cải thiện môi trường làm việc và giữ chân nhân viên hơn.

## Trực quan hóa dữ liệu

Việc trực quan hóa dữ liệu của bộ dữ liệu nhóm chúng em thực hiện trực quan trên công cụ Power BI để có cái nhìn trực quan hơn về bộ dữ liệu của công ty IBM.

A screenshot of a computer dashboard

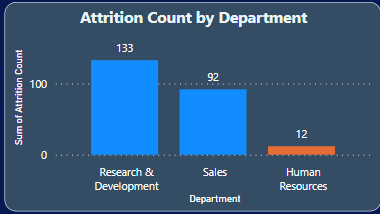
Description automatically generated

Hình 3.12 Trực quan hóa dữ liệu

Tổng quan của bộ dữ liệu sẽ có những thông tin cơ bản như tổng số phòng ban, tổng số nhân viên của công ty, số lượng rời bỏ của nhân viên, tỷ lệ rời bỏ, Tổng số nhân viên không rời bỏ, độ tuổi trung bình của nhân viên, số năm làm việc trung bình mà nhân viên làm việc ở công ty.

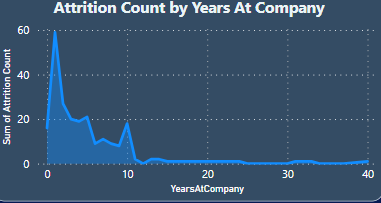
Công ty hiện có tổng số là 1470 nhân viên. Trong đó, 237 nhân viên đã rời bỏ công ty, chiếm tỷ lệ là 16.12%. Tỷ lệ rời bỏ này khá cao, điều này có thể chỉ ra rằng công ty đang gặp vấn đề trong việc giữ chân nhân viên và có thể gây ra những tổn thất lớn về mặt tài chính và hiệu suất làm việc của công ty. Số lượng nhân viên hiện đang làm việc cho công ty là 1233 người, cho thấy công ty vẫn duy trì được một lực lượng lao động đáng kể mặc dù có sự rời bỏ của các nhân viên khác. Độ tuổi trung bình của nhân viên trong công ty là 36.92 . Điều này cho chúng ta thấy, lực lượng lao động của công ty thuộc độ tuổi trưởng thành, có kinh nghiệm và có khả năng cống hiến lâu dài. Bên cạnh đó, số năm làm việc trung bình mà nhân viên làm việc ở công ty là 7.01, cho thấy một mức độ ổn định. Tuy nhiên, Nếu những nhân viên dày dặn kinh nghiệm này mà rời đi thì có thể ảnh hưởng lớn đến sự ổn định và hiệu quả làm việc của công ty.

Để phân tích rõ hơn về sự rời bỏ của nhân viên công ty IBM , chúng ta sẽ có những biểu đồ phân tích chi tiết như sau:



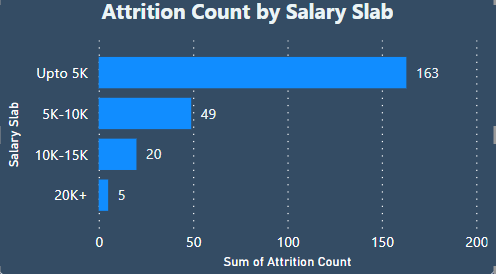
Hình 3.13 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo phòng ban

Ở biểu đồ “Attrition Count by Department” ,số lượng nhân viên rời bỏ theo phòng ban cho thấy ở Phòng Nghiên cứu & Phát triển (Research & Development) có 133 người, phòng Kinh doanh (Sales) có 92 người, và phòng Nhân sự (Human Resources) chỉ có 12 người. Điều này cho thấy rằng phòng Nghiên cứu & Phát triển có tỷ lệ nhân viên rời bỏ cao nhất, trong khi phòng Nhân sự lại có số lượng rời bỏ thấp nhất, phản ánh sự khác biệt rõ rệt trong mức độ gắn bó và giữ chân nhân viên giữa các phòng ban trong công ty.



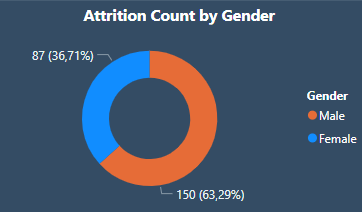
Hình 3.14 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo số năm làm việc

Biểu đồ “Attrition Count by Years at Company” cho thấy số lượng nhân viên rời bỏ theo số năm làm việc cho thấy xu hướng đáng chú ý. Giai đoạn có sự rời bỏ cao nhất là trong những năm đầu tiên, sau đó giảm mạnh và trở nên ổn định từ năm thứ 10 trở đi. Điều này cho thấy nhân viên có xu hướng rời bỏ công ty nhiều nhất trong năm đầu tiên, có thể do quá trình hội nhập chưa hiệu quả hoặc thiếu sự hỗ trợ cần thiết từ phía công ty. Tuy nhiên, nếu nhân viên vượt qua được ở những năm đầu, nhân viên có xu hướng gắn bó lâu dài hơn, cho thấy rằng những nhân viên ở lại sau năm đầu tiên thường có sự ổn định và cam kết lâu dài hơn với công ty.



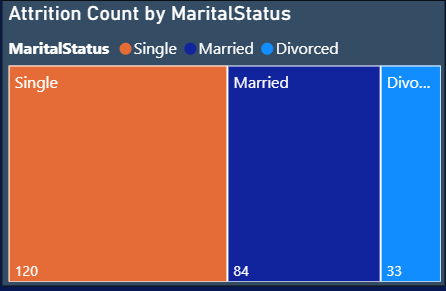
Hình 3.15 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo mức lương

Biểu đồ “ Attrition Count by Salary Slab” này thu nhập của nhân viên sẽ được phân loại thành 4 nhóm chính: "Upto 5K" cho thu nhập hàng tháng nhỏ hơn hoặc bằng 5000. "5K-10K" cho thu nhập hàng tháng lớn hơn 5000 và nhỏ hơn hoặc bằng 10000. "10K-15K" cho thu nhập hàng tháng lớn hơn 10000 và nhỏ hơn hoặc bằng 15000. Còn lại "20K+" cho thu nhập hàng tháng lớn hơn 15000. Từ việc phân loại trên cho ta thấy được, số lượng nhân viên rời bỏ theo mức lương cho thấy một xu hướng rõ rệt. Nhân viên có mức lương dưới 5K chiếm tỷ lệ rời bỏ cao nhất với 163 người, tiếp theo là nhóm lương từ 5K-10K với 49 người. Nhóm lương từ 10K-15K có 20 người rời bỏ và nhóm lương trên 20K chỉ có 5 người. Điều này cho thấy mức lương thấp có thể là một yếu tố quan trọng dẫn đến sự rời bỏ nhân viên. Do đó, công ty cần xem xét lại chính sách lương để cải thiện khả năng giữ chân nhân viên.



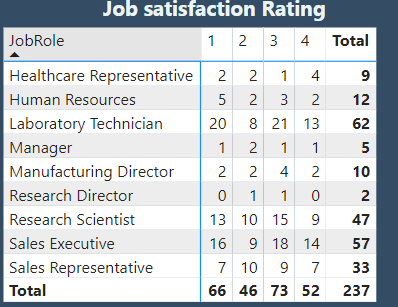
Hình 3.16 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo giới tính

Biểu đồ “ Attrition Count by Gender” này cho thấy số lượng nhân viên rời bỏ theo giới tính cho thấy tỷ lệ rời bỏ của nam giới cao hơn nữ giới, với 150 nam (chiếm 63,29%) và 87 nữ (chiếm 36,71%). Nam giới có xu hướng rời bỏ công ty nhiều hơn. Một số lý do có thể dẫn đến kết quả này bao gồm áp lực cạnh tranh cao hơn, kỳ vọng nghề nghiệp khác nhau, khó khăn trong việc cân bằng giữa công việc và cuộc sống, môi trường làm việc không phù hợp, thiếu chính sách hỗ trợ và phúc lợi công bằng, và có nhiều cơ hội việc làm khác bên ngoài công ty.



Hình 3.17 Biểu đồ thể hiện số lượng nhân viên rời bỏ theo tình trạng hôn nhân

Biểu đồ “ Attrition Count by Marital Status” cho thấy số lượng nhân viên rời bỏ theo tình trạng hôn nhân có tỷ lệ rời bỏ cao nhất, với 120 người, so với 84 người đã kết hôn và 33 người đã ly hôn. Điều này có thể xuất phát từ việc nhân viên độc thân gặp khó khăn trong việc cân bằng giữa công việc và cuộc sống cá nhân của họ. Cuối cùng sẽ là biểu đồ thể hiện mức độ hài lòng của nhân viên ở các chức vụ khác nhau.



Hình 3.18 Biểu đồ xếp hạng mức độ hài lòng của nhân viên

Trong biểu đồ này, xếp hạng mức độ hài lòng của nhân viên ở các vị trí chức vụ có sự khác biệt rõ rệt trong mức độ rời bỏ và sự hài lòng. Các vị trí như Đại diện Chăm sóc Sức khỏe với 9 nhân viên rời bỏ và Kỹ thuật viên Phòng thí nghiệm với 62 nhân viên rời bỏ đều có mức độ hài lòng thấp. Phòng Nhân sự có 12 nhân viên rời bỏ với mức độ hài lòng thấp. Các vai trò như Giám đốc Sản xuất với 10 nhân viên rời bỏ và Nhà khoa học Nghiên cứu với 47 nhân viên rời bỏ cũng phần lớn không hài lòng. Trong khi đó, các vai trò như Quản lý và Giám đốc Nghiên cứu có sự phân bố đều các mức độ hài lòng, với lần lượt 15 và 7 nhân viên rời bỏ. Từ đó, ta thấy rằng sự hài lòng công việc có thể chịu ảnh hưởng của nhiều yếu tố khác nhau, bao gồm quản lý kém hoặc thiếu cơ hội phát triển. Những chức vụ có số lượng nhân viên rời bỏ cao thường đi kèm với mức độ hài lòng khá thấp.

Sau cùng, qua những phân tích trên nhóm em rút ra được rằng Tuổi trung bình của những nhân viên đã rời công ty là 40 tuổi. Bộ phận Nghiên cứu & Phát triển ghi nhận tỷ lệ nghỉ việc cao nhất, chiếm khoảng 14% tổng số. Phần lớn những nhân viên rời công ty thuộc khoảng lương dưới 5000 đô la mỗi tháng, tổng cộng 163 người. Nhân viên nam trải qua tỷ lệ nghỉ việc là 63,29%, trong khi nhân viên nữ có tỷ lệ nghỉ việc hơi thấp hơn, là 36,71%. Ở những năm đầu của nhân viên mới tỷ lệ rời bỏ công ty khá cao, còn ở các nhân viên có số năm làm việc hơn 10 năm có độ gắn bó lâu dài với công ty hơn. Đây cũng là những yếu tố ảnh hưởng đến sự rời bỏ của nhân viên.

# XÂY DỰNG MÔ HÌNH DỰ ĐOÁN SỰ RỜI BỎ CỦA NHÂN VIÊN CÔNG TY IBM

## Mô hình Logistic Regression

### Chuẩn bị dữ liệu

Sử dụng bộ dữ liệu đã được xử lý ở trên để tiến hành xây dựng mô hình. Thực hiện import các thư viện cần thiết và đọc dữ liệu.

A screenshot of a computer

Description automatically generated

Hình 4.1 Chuẩn bị thư viện và đọc dữ liệu

Dữ liệu sau khi xử lý đưa về file csv sẽ có thêm một cột index không có tên cột. Tiến hành xóa đi cột index dư thừa.

Sử dụng hàm nunique() để xem các cột có bao nhiêu giá trị duy nhất.

A screenshot of a computer

Description automatically generated

Hình 4.2 In số giá trị duy nhất của các cột

In ra các cột có kiểu dữ liệu là object để phục vụ cho bước chuyển đổi

A white rectangular object with a black border

Description automatically generated with medium confidence

Hình 4.3 In các cột có kiểu dữ liệu là object

Mô hình Logistic Regression là mô hình dự đoán và phân loại các mẫu theo hai lớp hoặc hai nhãn có giá trị nhị phân thường là 0 và 1. Tiến hành chuyển đổi các giá trị về nhãn 0 và 1.

Sau khi xem được các cột có bao nhiêu giá trị duy nhất và cột nào có kiểu dữ liệu là object, thấy được các cột Attrition, Gender, OverTime có 2 giá trị duy nhất là ‘Yes’ và ‘No’ nên sẽ sử dụng hàm replace để chuyển giá trị ‘Yes’ về 1 và ‘No’ về 0. Các cột BusinessTravel, Department, EducationField, JobRole, MaritalStatus có nhiều hơn 2 giá trị và có kiểu dữ liệu là object nên sẽ import lớp LabelEncoder từ thư viện sklearn.preprocessing. LabelEncoder là một công cụ để mã hóa các biến phân loại sang dạng số.

A screenshot of a computer

Description automatically generated

Hình 4.4 Chuyển đổi dữ liệu về dạng số

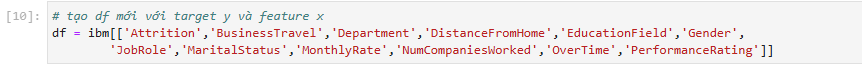
Sử dụng ma trận tương quan để chọn ra những biến độc lập có mối quan hệ tương quan cao với biến phụ thuộc Attrition. Loại bỏ những biến có mối quan hệ tương quan âm.

A screenshot of a computer

Description automatically generated

Hình 4.5 Mối quan hệ tương quan

Tạo một dataframe mới chứa biến phụ thuộc Attrition và biến độc lập được chọn từ ma trận tương quan gồm Attrition, BusinessTravel, Department, DistanceFromHome, EducationField, Gender, JobRole, MaritalStatus, MonthlyRate, NumCompaniesWorked, OverTime, PerformanceRating.



Hình 4.6 Tạo dataframe mới

Chia dữ liệu thành 2 tập biến độc lập x và biến phụ thuộc y trước khi tiến hành xây dựng mô hình

A white rectangular object with black text

Description automatically generated

Hình 4.7 Chia dữ liệu

Tỷ lệ nhân viên rời bỏ đang có sự mất cân bằng khi có 1233 nhân viên không rời bỏ và chỉ có 237 nhân viên rời bỏ. Sự mất cân bằng này sẽ ảnh hưởng đến việc xây dựng mô hình, dẫn tới đưa ra các dự đoán sai về sự rời bỏ của nhân viên.

Để giải quyết vấn đề mất cân bằng dữ liệu sẽ sử dụng kĩ thuật SMOTE một phương pháp phổ biến dùng để cân bằng dữ liệu sẽ giúp cải thiện hiệu suất của mô hình dự đoán bằng cách giảm đi dữ liệu lớp đa số hoặc tạo ra các dữ liệu mới cho lớp thiểu số. Ở dữ liệu này, chúng ta sẽ dùng SMOTE để tạo thêm dữ liệu mới cho lớp 1 (nhân viên rời bỏ)

A screenshot of a computer

Description automatically generated

Hình 4.8 Cân bằng dữ liệu

Để đảm bảo các biến có thang đo khác nhau sẽ có phạm vi giá trị tương đồng, tránh một vài biến có ảnh hưởng quá lớn so với các biến khác, chúng ta sẽ sử dụng phương pháp chuẩn hóa Min-max để chuẩn hóa các biến về khoảng 0-1.

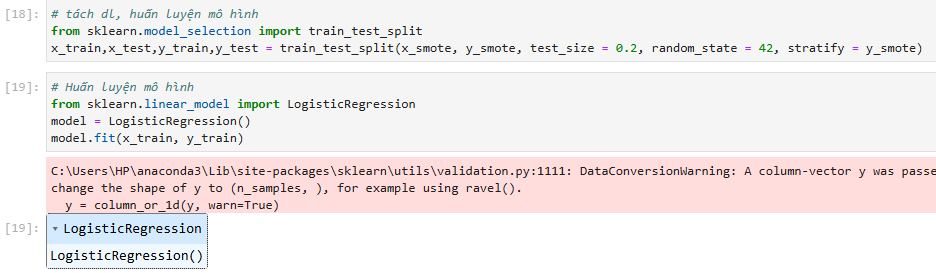
A screenshot of a computer

Description automatically generated

Hình 4.9 Chuẩn hóa dữ liệu

### Huấn luyện mô hình

Sau khi hoàn thành các bước chuẩn bị dữ liệu sẽ tiến hành tách dữ liệu thành 2 tập, 80% dữ liệu dùng để huấn luyện và 20% còn lại dùng để kiểm tra.



Hình 4.10 Tách dữ liệu và huấn luyện mô hình

### Đánh giá mô hình

Sau khi huấn luyện mô hình, import các thư viện cần thiết để in ra độ chính xác trên tập train và tập test. Có thể thấy được 2 tập có độ chính xác tương đương nhau khá cao khoảng 62%, cho thấy mô hình hoạt động khá tốt.

A screenshot of a computer program

Description automatically generated

Hình 4.11 Độ chính xác của mô hình Logistic Regression

Mô hình classification\_report cung cấp một báo cáo chi tiết về hiệu suất của một mô hình phân loại.

A screenshot of a computer

Description automatically generated

Hình 4.12 Các chỉ số đánh giá mô hình của Logistic Regression

Độ chính xác (Accuracy): Mô hình có độ chính xác là 0.61, cho thấy mô hình dự đoán đúng 61% các trường hợp. Đây là một tỷ lệ tương đối cao, cho thấy mô hình có khả năng phân biệt tốt giữa hai lớp mục tiêu.

Độ chính xác (Precision): Đối với lớp 0 nhân viên không rời bỏ có độ chính xác khoảng 60% và độ chính xác lớp 1 lớp nhân viên rời bỏ là 62%. Cho thấy mô hình dự đoán lớp 1 chính xác hơn so với lớp 0.

Độ nhạy (Recall): Đối với lớp 0 có độ nhạy là 66% và lớp 1 có độ nhạy 55%. Cho thấy mô hình có khả năng phát hiện và nhận dạng tốt hơn nhãn 0 so với nhãn 1.

Điểm F1 (F1-score): Là mức trung bình giữa độ chính xác và độ nhạy. Với điểm F1 cho lớp 0 là 0.63 và cho lớp 1 là 0.59. Điểm F1 nằm trong khoảng 0 đến 1, với 1 là hiệu suất hoàn hảo. Vì vậy, với điểm F1 trên 0.60 cho cả hai lớp, ta có thể đánh giá mô hình này có hiệu suất khá tốt.

Support: Đưa ra số lượng mẫu trong tập kiểm tra của mỗi nhãn. Cả 2 lớp đều có số lượng mẫu là 247.

A diagram of a confusion matrix

Description automatically generated

Hình 4.13 Mô hình Confusion Matrix của Logistic Regression

Trong mô hình đánh giá Confusion Matrix

True Negative (TN): Có 163 điểm dữ liệu thực tế thuộc nhóm 0 được dự đoán đúng là nhóm 0.

True Positive (TP): Có 137 điểm dữ liệu thực tế thuộc nhóm 1 được dự đoán đúng là nhóm 1.

False Positive (FP): Có 84 điểm dữ liệu thực tế thuộc nhóm 0 được dự đoán là nhóm 1.

False Negative (FN) : Có 110 điểm dữ liệu thực tế thuộc nhóm 1 được dự đoán là nhóm 0.

Nhận xét mô hình:

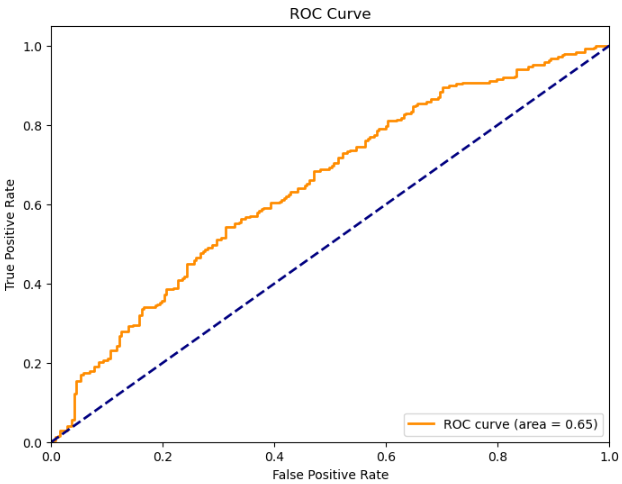
Precision của mô hình khoảng 61.9%, tức là trong số những nhân viên được dự đoán sẽ rời bỏ, có khoảng 61.9% là đúng. Thông số này cho thấy mô hình khá chính xác khi dự đoán nhân viên sẽ rời bỏ, nhưng vẫn còn khá nhiều dự đoán sai (False Positives).

Recall là 55,4% tức là mô hình chỉ phát hiện được khoảng 55,4% số nhân viên thực sự rời bỏ. Đồng nghĩa với việc mô hình bỏ sót khá nhiều nhân viên rời bỏ (False Negatives nhiều).

Specificity là khoảng 65.9%, cho thấy mô hình phát hiện đúng khoảng 66% số nhân viên không rời bỏ. Điều này khá tốt nhưng vẫn còn nhiều nhân viên không rời bỏ bị dự đoán sai là rời bỏ (False Positives).

F1-Score của mô hình là 58.5%, nghĩa là mô hình có mức độ cân bằng tương đối giữa Precision và Recall. Tuy nhiên, F1-Score thấp hơn Precision cho thấy mặc dù mô hình khá tốt trong việc xác định những nhân viên sẽ rời bỏ, nhưng khả năng phát hiện tất cả những nhân viên thực sự rời bỏ Recall vẫn chưa cao.

Có thể thấy số lượng FN khá nhiều, đây là lớp đại diện cho những nhân viên có nguy cơ rời bỏ nhưng mô hình không phát hiện ra. Nếu không can thiệp kịp thời, công ty sẽ mất đi những nhân viên này. Việc mất đi những nhân viên có thể gây thiệt hại lớn về cả chi phí đào tạo lại và giảm năng suất làm việc. Vì vậy ta có thể tăng Recall, điều chỉnh ngưỡng phân loại (threshold) thấp hơn để tăng khả năng phát hiện nhân viên có nguy cơ rời bỏ, không có sai sót của mô hình.



Hình 4.14 Mô hình ROC của Logistic Regression

Đường cong ROC (ROC curve) mô tả mối quan hệ giữa tỷ lệ các trường hợp dương tính đúng (True Positive Rate - TPR) và tỷ lệ các trường hợp âm tính sai (False Positive Rate - FPR) ở các ngưỡng khác nhau của mô hình biểu diễn hiệu suất của mô hình. AUC là diện tích được tính dưới đường cong ROC. AUC có giá trị thường nằm trong khoảng từ 0 đến 1, càng gần 1 thì mô hình càng tốt. Nếu AUC bằng 1 thì đó là một mô hình hoàn hảo, và là một mô hình ngẫu nhiên khi AUC bằng 0.5.

Trong mô hình ROC ở trên có AUC bằng 0.65, cho thấy mô hình có khả năng phân loại ở mức trung bình, không phải rất tốt nhưng cũng không quá kém.

Đường cong ROC nằm phía trên đường chéo (đường đối xứng), chứng tỏ mô hình Logistic Regression có hiệu suất tốt hơn so với phân loại ngẫu nhiên (AUC-ROC = 0.5).

Với mức FPR càng cao, TPR cũng tăng lên, cho thấy mô hình có thể điều chỉnh độ nhạy (sensitivity) và độ đặc hiệu (specificity) bằng cách thay đổi ngưỡng phân loại.

Qua mô hình ROC, mô hình có hiệu suất trung bình trong việc phân biệt các trường hợp rời và không rời.

## Mô hình Naïve Bayes

### Chuẩn bị dữ liệu

Sử dụng bộ dữ liệu đã được xử lý ở trên để tiến hành xây dựng mô hình. Thực hiện import các thư viện cần thiết và đọc dữ liệu vào.

A screenshot of a computer

Description automatically generated

Hình 4.15 Import thư viện và đọc dữ liệu vào

Sử dụng bộ dữ liệu đã phân tích ở trên sẽ tiến hành chuyển đổi các biến phân loại.

A close-up of a computer screen

Description automatically generated

Hình 4.16 Code chuyển đổi biến phân loại

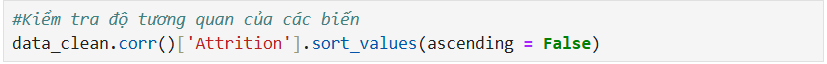
Naive Bayes là một thuật toán xác suất dựa trên định lý Bayes. Nó tính toán xác suất có điều kiện của từng lớp (category) dựa trên các đặc trưng đầu vào. Các tính toán xác suất này yêu cầu dữ liệu đầu vào phải là số để thực hiện các phép toán học. Do đó ở bước này cần sử dụng cần sử dụng **LabelEncoder** để chuyển đổi các cột phân loại gồm Attrition, BusinessTravel, Department, Education thành các giá trị số để giúp việc xây dựng mô hình chính xác hơn.

A screenshot of a white table

Description automatically generated

Hình 4.17 Dữ liệu sau khi chuyển đổi

Khi các cột được chuyển đổi xong, ta tiến hành kiểm tra và lọc độ tương quan của các biến. Sử dụng ma trận tương quan để chọn ra những biến độc lập có mối quan hệ tương quan cao với biến phụ thuộc Attrition. Loại bỏ những biến có mối quan hệ tương quan âm.



Hình 4.18 Kiểm tra độ tương quan

A screenshot of a computer

Description automatically generated

Hình 4.19 Kết quả độ tương quan

Đầu tiên, chúng ta tính toán ma trận hệ số tương quan giữa tất cả các cột trong DataFrame **data\_clean** bằng hàm **corr().** Từ ma trận này, trích xuất cột **Attrition** và sắp xếp các hệ số tương quan theo thứ tự giảm dần, kết quả là một Series với các hệ số tương quan giữa **Attrition** và các cột khác, sắp xếp từ lớn nhất đến nhỏ nhất.

A computer code with text

Description automatically generated with medium confidence

Hình 4.20 Tạo dữ liệu mới

Sau khi đã sắp xếp từ lớn nhất đến nhỏ nhất, chúng ta lọc các hệ số tương quan chỉ giữ lại những giá trị không âm (>= 0) và loại bỏ các cột có giá trị tương quan âm. Tiếp theo, lưu danh sách các cột giá trị không âm vào dataframe mới có tên là **Data\_new**. Dataframe mới này sẽ chứa các cột có giá trị không âm giúp đơn giản hóa mô hình và tập trung vào các biến có khả năng dự đoán tốt hơn nâng cao hiệu suất của mô hình.

Sau khi loại bỏ các cột có tương quan âm thì bộ dữ liệu lúc này còn 12 cột và 1470 bản ghi như sau:

A screenshot of a table

Description automatically generated

Hình 4.21 Dữ liệu mới

Trước khi xây dựng mô hình chia dữ liệu thành 2 tập biến độc lập x và biến phụ thuộc y để tiến hành cân bằng dữ liệu trước khi xây dựng mô hình.

A close-up of a computer screen

Description automatically generated

Hình 4.22 Chia dữ liệu

Ở tập dữ liệu này, chúng ta nhận thấy tỷ lệ nhân viên rời bỏ đang có sự mất cân bằng khi có 1233 nhân viên không rời bỏ và chỉ có 237 nhân viên rời bỏ. Sự mất cân bằng này sẽ ảnh hưởng đến việc xây dựng mô hình, dẫn tới đưa ra các dự đoán sai về sự rời bỏ của nhân viên.

A close-up of a screen

Description automatically generated

Hình 4.23 Dữ liệu trước khi cân bằng

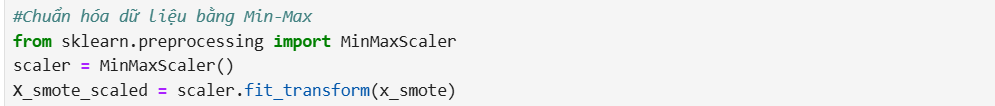
Để giải quyết vấn đề mất cân bằng dữ liệu sẽ sử dụng kĩ thuật SMOTE một phương pháp phổ biến dùng để cân bằng dữ liệu sẽ giúp cải thiện hiệu suất của mô hình dự đoán bằng cách giảm đi dữ liệu lớp đa số hoặc tạo ra các dữ liệu mới cho lớp thiểu số. Ở dữ liệu này, chúng ta sẽ dùng SMOTE để tạo thêm dữ liệu mới cho lớp 1 (nhân viên rời bỏ)

A white background with black text

Description automatically generated

Hình 4.24 Dữ liệu sau khi cân bằng

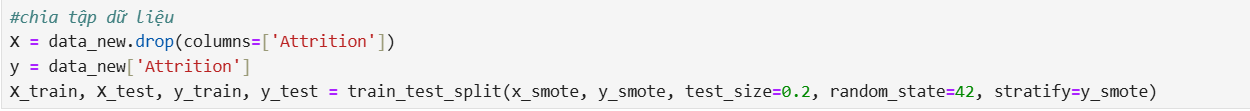
Để đảm bảo các biến có thang đo khác nhau sẽ có phạm vi giá trị tương đồng, tránh một vài biến có ảnh hưởng quá lớn so với các biến khác, chúng ta sẽ sử dụng phương pháp chuẩn hóa Min-max để chuẩn hóa lại các biến về khoảng 0-1.



Hình 4.25 Chuẩn hóa dữ liệu

### Huấn luyện mô hình

Sau khi hoàn thành các bước chuẩn bị dữ liệu sẽ tiến hành tách dữ liệu thành 2 tập, 80% dữ liệu dùng để huấn luyện và 20% còn lại dùng để kiểm tra.



Hình 4.26 Tách dữ liệu

Bắt đầu huấn luyện mô hình

A white background with a black and white image

Description automatically generated with medium confidence

Hình 4.27 Huấn luyện mô hình Naïve Bayes

### Đánh giá mô hình

A computer screen shot of a computer code

Description automatically generated

Hình 4.28 Code tạo mô hình đánh giá

Chúng ta sử dụng mô hình đã huấn luyện để dự đoán nhãn **(Attrition)** cho tập dữ liệu kiểm tra **(X\_test).** Kết quả của các dự đoán này được lưu trữ trong biến **y\_pred.**

Tiếp theo, chúng ta tính toán độ chính xác của mô hình bằng cách so sánh các nhãn thực tế **(y\_test)** với các nhãn dự đoán **(y\_pred).** Độ chính xác được tính bằng hàm **accuracy\_score(y\_test, y\_pred)** và kết quả được in ra cùng với báo cáo phân loại chi tiết.

A screenshot of a computer screen

Description automatically generated

Hình 4.29 Các chỉ số đánh giá mô hình Naïve Bayes

Báo cáo phân loại được tạo bằng hàm classification\_report(y\_test, y\_pred) cung cấp các chỉ số như Precision, Recall, và F1-score cho mỗi lớp. Việc in ra báo cáo này giúp chúng ta hiểu rõ hơn về hiệu suất của mô hình đối với từng lớp cụ thể như sau:

Ở lớp 0 ( lớp không rời bỏ) ý nghĩa của các chỉ số

Precision (0.73): Trong số các dự đoán mà mô hình dự đoán là lớp 0, có 73% là chính xác. Precision cao cho thấy mô hình ít bị dương tính giả (false positives) cho lớp 0.

Recall (0.44): Trong số các trường hợp thực tế là lớp 0, mô hình chỉ dự đoán chính xác 44%. Recall thấp cho thấy mô hình có nhiều âm tính giả (false negatives) cho lớp 0.

F1-score (0.55): F1-score là trung bình hài hòa giữa Precision và Recall. Với giá trị 0.55, F1-score cho thấy rằng mặc dù mô hình có độ chính xác tương đối cao khi dự đoán lớp 0, nhưng khả năng phát hiện chính xác các trường hợp thuộc lớp 0 còn hạn chế.

Còn ở lớp 1( lớp rời bỏ) các chỉ số thể hiện như sau:

Precision (0.60): Trong số các dự đoán mà mô hình dự đoán là lớp 1, có 60% là chính xác. Precision ở mức trung bình, chỉ ra rằng có một số lượng nhất định các dương tính giả (false positives) cho lớp 1.

Recall (0.84): Trong số các trường hợp thực tế là lớp 1, mô hình dự đoán chính xác 84%. Recall cao cho thấy mô hình ít bỏ sót các trường hợp thuộc lớp 1.

F1-score (0.70): F1-score là trung bình giữa 2 chỉ số là Precision và Recall. Với giá trị 0.70, F1-score cho thấy mô hình có khả năng phát hiện tốt các trường hợp thuộc lớp 1 và dự đoán tương đối chính xác.

Từ đó, Mô hình Naive Bayes có hiệu suất trung bình với độ chính xác 64%. Mô hình hoạt động tốt hơn trong việc phát hiện các trường hợp rời bỏ (lớp 1) với Recall cao (0.84) nhưng gặp khó khăn trong việc dự đoán chính xác các trường hợp không rời bỏ (lớp 0) với Recall thấp (0.44). Điều này có thể dẫn đến việc mô hình có xu hướng bỏ sót nhiều trường hợp không rời bỏ.

Sau đó, chúng tôi tạo ma trận nhầm lẫn từ các nhãn thực tế và nhãn dự đoán bằng hàm confusion\_matrix(y\_test, y\_pred). Ma trận nhầm lẫn này hiển thị số lượng các dự đoán đúng và sai cho mỗi lớp

A blue squares with white text

Description automatically generated

Hình 4.30 Mô hình Confusion Matrix của Naïve Bayes

Trong ma trận confusion matrix

True Negatives (TN): 108 - Số lượng các trường hợp mà mô hình dự đoán đúng là không rời bỏ và thực tế cũng là không rời bỏ.

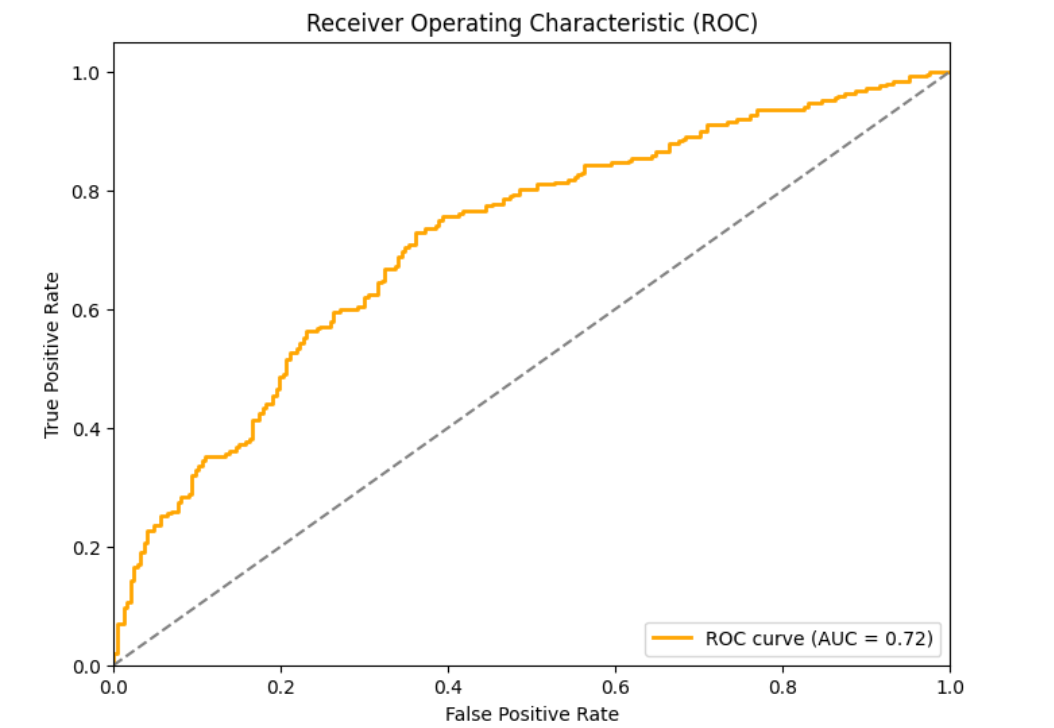
False Positives (FP): 139 - Số lượng các trường hợp mà mô hình dự đoán sai là rời bỏ trong khi thực tế là không rời bỏ.

False Negatives (FN): 40 - Số lượng các trường hợp mà mô hình dự đoán sai là không rời bỏ trong khi thực tế là rời bỏ.

True Positives (TP): 207 - Số lượng các trường hợp mà mô hình dự đoán đúng là rời bỏ và thực tế cũng là rời bỏ.

Hiệu suất của mô hình Naive Bayes cho thấy rằng trong dự đoán lớp 0 (Không rời bỏ), mô hình có 108 dự đoán đúng và 139 dự đoán sai, cho thấy nó thường xuyên dự đoán sai khi xử lý lớp này. Ngược lại, với lớp 1 (Rời bỏ), mô hình đạt 207 dự đoán đúng và chỉ 40 dự đoán sai, chứng tỏ khả năng dự đoán của nó tốt hơn nhiều với ít sai số hơn. Điều này cho thấy mô hình Naive Bayes hoạt động hiệu quả hơn trong việc dự đoán các trường hợp rời bỏ (lớp 1) so với các trường hợp không rời bỏ (lớp 0). Tuy nhiên, số lượng dự đoán sai ở lớp 0 vẫn còn khá cao.

Thực hiện đánh giá mô hình phân loại bằng cách vẽ đường cong ROC và tính toán chỉ số AUC. Đầu tiên, mã dự đoán xác suất thuộc lớp positive cho các mẫu trong tập kiểm tra bằng cách sử dụng model.predict\_proba(X\_test)[:, 1]. Sau đó, nó tính toán các giá trị False Positive Rate (FPR) và True Positive Rate (TPR) thông qua roc\_curve(y\_test, y\_prob), cùng với chỉ số AUC bằng roc\_auc\_score(y\_test, y\_prob). Đồ thị ROC được vẽ với màu cam, và đường chéo màu xám thể hiện mô hình phân loại ngẫu nhiên. Các trục được gán nhãn và giới hạn, với tiêu đề và chú thích mô tả AUC. Cuối cùng, giá trị AUC được in ra, cho thấy khả năng phân loại của mô hình.



Hình 4.31 Mô hình ROC của Naïve Bayes

Đường cong ROC nằm trên và bên trái của đường chéo 45 độ, cho thấy mô hình có khả năng phân loại tốt hơn so với phỏng đoán ngẫu nhiên. Điều này chỉ ra rằng mô hình có thể phân biệt khá tốt giữa nhân viên rời bỏ và nhân viên không rời bỏ.

Chỉ số AUC đạt 0.72, nằm trong khoảng từ 0.5 đến 1.0. Một AUC bằng 1 chỉ ra khả năng phân loại hoàn hảo, trong khi AUC bằng 0.5 cho thấy mô hình không tốt hơn so với phỏng đoán ngẫu nhiên. Mức AUC 0.72 cho thấy mô hình có khả năng phân loại khá tốt.

Tỷ lệ False Positive thấp (FPR ~ 0.3) mô hình ít nhầm lẫn các trường hợp không rời bỏ là rời bỏ, cho thấy độ chính xác tương đối cao.

Tỷ lệ True Positive khá tốt (TPR ~ 0.6) mô hình có khả năng phát hiện chính xác khoảng 60% các trường hợp nhân viên rời bỏ, đây là một mức TPR khá tốt.

Nhìn chung, qua các chỉ số này mô hình dự đoán rời bỏ của nhân viên này là khá hiệu quả và có thể được sử dụng để hỗ trợ các quyết định liên quan đến giữ chân nhân tài.

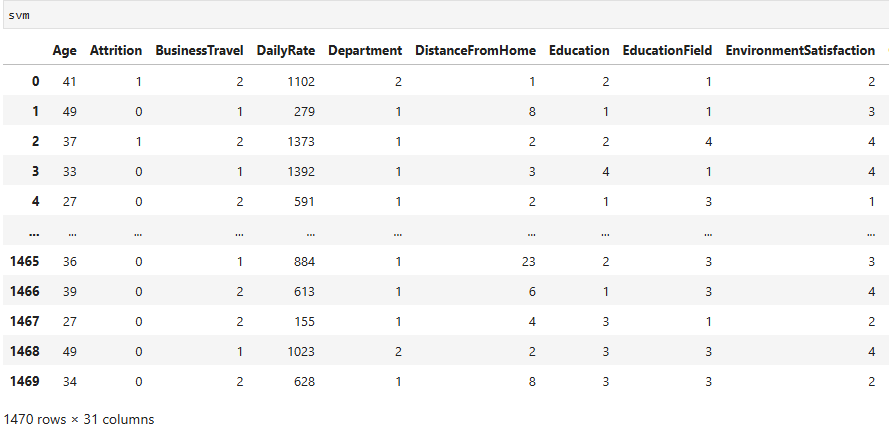
## Mô hình Support Vector Machine

### Chuẩn bị dữ liệu

Tiến hành import thư viện cần thiết, đọc dữ liệu đã xử lý vào, và chuẩn hóa dữ liệu tương tự min-max như các thuật toán ở trên



Hình 4.32 Chuyển đổi dữ liệu

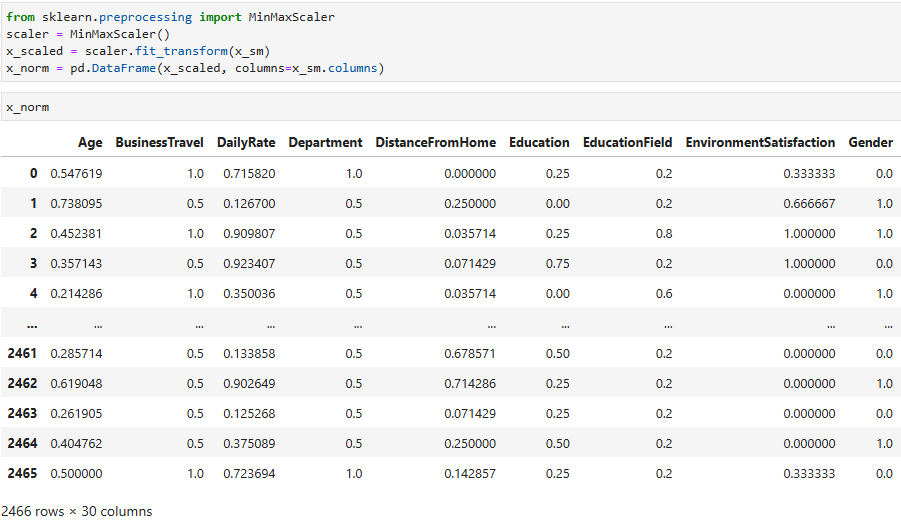


Hình 4.33 Dữ liệu sau khi chuyển đổi

Chia dữ liệu thành 2 tập biến độc lập x và biến phụ thuộc y trước khi tiến hành xây dựng mô hình.

Tương tự 2 thuật toán ở trên để giải quyết vấn đề mất cân bằng dữ liệu sẽ sử dụng kĩ thuật SMOTE để cân bằng lại dữ liệu bằng cách tạo thêm dữ liệu mới cho lớp 1 (nhân viên rời bỏ)

Sử dụng phương pháp chuẩn hóa Min-max để chuẩn hóa các biến về khoảng [0-1], đảm bảo các biến có thang đo khác nhau sẽ có phạm vi giá trị tương đồng, tránh một vài biến có ảnh hưởng quá lớn so với các biến khác và giúp giảm bớt sự ảnh hưởng của các outlier bằng cách nén các giá trị vào một khoảng nhỏ hơn.



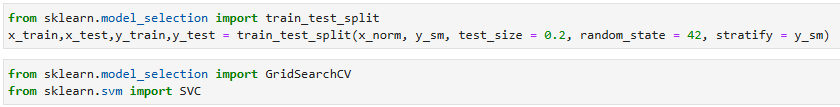
Hình 4.34 Dữ liệu sau khi chuẩn hóa

### Huấn luyện mô hình

Tách dữ liệu thành 2 tập, 80% dùng để huấn luyện và 20% còn lại được dùng để kiểm tra mô hình.

Nhập lớp GridSearchCV từ thư viện sklearn.model\_selection. GridSearchCV là một công cụ trong scikit-learn dùng để tìm kiếm toàn diện các tổ hợp siêu tham số (hyperparameters) cho một mô hình nhằm tối ưu hóa hiệu suất của mô hình đó.

Import lớp SVC từ thư viện sklearn.svm. SVC dùng cho việc xây dựng mô hình phân loại.



Hình 4.35 Import thư viện SVM

Tạo một từ điển param\_grid chứa các siêu tham số dùng thử nghiệm tìm giá trị tối ưu cho mô hình SVM, ở bài toán này sẽ dùng 3 tham số là C, gamma và kernel:

C: Tham số điều chỉnh độ mạnh của ràng buộc lỗi trong mô hình SVM.

Nếu C lớn, mô hình sẽ cố gắng phân loại tất cả các điểm dữ liệu một cách chính xác, điều này có thể dẫn đến overfitting. Nếu C nhỏ, mô hình sẽ cho phép một số điểm dữ liệu sai lệch để tránh overfitting và có thể dẫn đến underfitting.

Gamma: Tham số điều chỉnh độ rộng vùng ảnh hưởng xung quanh mỗi điểm dữ liệu.

Giá trị gamma cao, ảnh hưởng của mỗi điểm dữ liệu là ngắn và gần hơn, có thể dẫn đến overfitting. Giá trị gamma thấp, ảnh hưởng của mỗi điểm dữ liệu là xa và rộng hơn, có thể dẫn đến underfitting.

Kernel: Hàm dùng để biến đổi dữ liệu đầu vào thành không gian đặc trưng cao hơn, giúp mô hình phân tách các điểm dữ liệu phức tạp hơn. Các kernel phổ biến gồm:

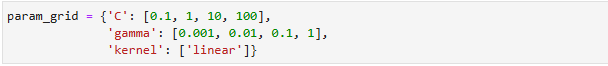
Linear Kernel: Dùng cho dữ liệu mà có thể phân tách tuyến tính.

Polynomial Kernel: Dùng cho dữ liệu có mối quan hệ không tuyến tính phức tạp.

Radial Basis Function (RBF) Kernel: Dùng cho dữ liệu mà không có đường phân tách tuyến tính rõ ràng.

Sigmoid Kernel: Dùng cho mạng nơ-ron nhân tạo.

Ở bài toán này sẽ sử dụng kernel là 'linear', mô hình sẽ sử dụng một hyperplane tuyến tính để phân loại dữ liệu.



Hình 4.36 Từ điển chứa siêu tham số

Trước khi vào huấn luyện mô hình sẽ khởi tạo một đối tượng GridSearchCV để thực hiện tìm kiếm lưới cho mô hình SVM.

Sử dụng GridSearchCV để huấn luyện mô hình với dữ liệu huấn luyện. values.ravel() được sử dụng để chuyển đổi nhãn thành mảng 1 chiều, đảm bảo tính tương thích với mô hình.

grid.best\_params\_: Trả về một từ điển chứa các siêu tham số tốt nhất đã được tìm thấy trong quá trình tìm kiếm. Đây là tổ hợp các siêu tham số mang lại hiệu suất tốt nhất cho mô hình dựa trên cross-validation.

grid.best\_estimator\_: Trả về mô hình SVM đã được huấn luyện với các siêu tham số tốt nhất. Đây là mô hình sẵn sàng để sử dụng cho việc dự đoán và đánh giá trên dữ liệu mới.

Kết quả trả về cho thuật toán mô hình với các siêu tham số bao gồm: C=0.1, gamma=0.001 và kernel=‘linear’

A screenshot of a computer program

Description automatically generated

Hình 4.37 Huấn luyện mô hình với siêu tham số tốt nhất

Nhờ các siêu tham số điều chỉnh, ta có thể thấy được tập huấn luyện và tập kiểm tra có độ chính xác cao. Tập huấn luyện cho độ chính xác khoảng 84% và tập kiểm tra là 82%.

A screenshot of a computer code

Description automatically generated

Hình 4.38 Độ chính xác của mô hình SVM

### Đánh giá mô hình

**A screenshot of a computer code

Description automatically generated**

Hình 4.39 Chỉ số đánh giá mô hình SVM

Với thuật toán SVM cho ra được báo cáo phân loại như sau:

Độ chính xác (Accuracy): Mô hình có độ chính xác là 0.82, cho thấy mô hình dự đoán đúng 82% các trường hợp. Đây là một tỷ lệ rất cao, cho thấy mô hình phân biệt tốt giữa hai lớp mục tiêu.

Độ chính xác (Precision): Đối với lớp 0 nhân viên không rời bỏ có độ chính xác khoảng 80% và độ chính xác lớp 1 lớp nhân viên rời bỏ là 84%. Cho thấy mô hình dự đoán lớp 1 chính xác hơn so với lớp 0.

Độ nhạy (Recall): Đối với lớp 0 có độ nhạy là 85% và lớp 1 có độ nhạy 79%. Cho thấy mô hình có khả năng phát hiện và nhận dạng tốt hơn nhãn 0 so với nhãn 1.

Điểm F1 (F1-score): Là mức trung bình giữa độ chính xác và độ nhạy. Với điểm F1 cho lớp 0 là 83% và cho lớp 1 là 81%. Điểm F1 nằm trong khoảng 0 đến 1, với 1 là hiệu suất hoàn hảo. Vì vậy, với điểm F1 trên 60% cho cả hai lớp, ta có thể đánh giá mô hình này có hiệu suất khá tốt.

**A diagram of a confusion matrix

Description automatically generated**

Hình 4.40 Mô hình Confusion Matrix của SVM

Nhận xét mô hình:

Precision của mô hình khoảng 84,4%, tức là trong số những nhân viên được dự đoán sẽ rời bỏ, có khoảng 84,4% là đúng. Thông số này cho thấy mô hình khá chính xác khi dự đoán nhân viên sẽ rời bỏ, nhưng vẫn còn khá nhiều dự đoán sai (False Positives).

Recall là 78,5% tức là mô hình chỉ phát hiện được khoảng 78,5% số nhân viên thực sự rời bỏ. Đồng nghĩa với việc mô hình bỏ sót khá nhiều nhân viên rời bỏ (False Negatives nhiều).

Specificity là khoảng 85,4%, cho thấy mô hình phát hiện đúng khoảng 85,4% số nhân viên không rời bỏ. Điều này khá tốt nhưng vẫn còn nhiều nhân viên không rời bỏ bị dự đoán sai là rời bỏ (False Positives).

F1-Score khoảng 81.3%, một chỉ số cao phản ánh mô hình không chỉ chính xác trong việc xác định các trường hợp nhân viên rời bỏ mà còn tương đối tốt trong việc phát hiện hầu hết các trường hợp nhân viên rời bỏ.

Qua các chỉ số đánh giá mô hình, ta có thể thấy được với thuật toán SVM đã xây dựng một mô hình có hiệu suất rất tốt trong bài toán dự đoán việc rời bỏ của nhân với các chỉ số đều có tỷ lệ trên 80%.

A graph of a curve

Description automatically generated

Hình 4.41 Mô hình ROC của SVM

Diện tích dưới đường cong (Area Under Curve - AUC) là 0,89, cho thấy mô hình có hiệu suất phân loại tốt. AUC càng gần 1, hiệu suất của mô hình càng cao.

Đường cong bắt đầu từ điểm (0,0), tức là khi không có dương tính nào được phát hiện, và kết thúc ở điểm (1,1), tức là tất cả các trường hợp dương tính và âm tính đều được phát hiện chính xác. Đường cong nằm trên đường chéo, thể hiện hiệu suất của mô hình tốt hơn so với phân loại ngẫu nhiên.

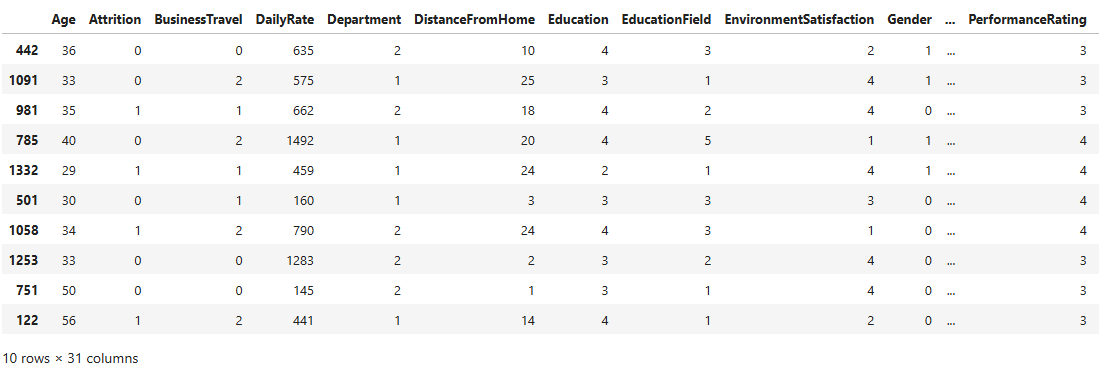
Đường cong tăng liên tục, nghĩa là khi tỷ lệ dương tính sai tăng, tỷ lệ dương tính đúng cũng tăng tương ứng. Điều này cho thấy mô hình có khả năng phân biệt giữa các lớp dữ liệu.

Đường cong càng gần góc trái trên (điểm [0,1]), thì hiệu suất của mô hình càng tốt, vì lúc này tỷ lệ dương tính đúng cao trong khi tỷ lệ dương tính sai thấp.

## Mô hình Neural Network

### Chuẩn bị dữ liệu

Tương tự các thuật toán trên, import các thư viện cần thiết và đưa bộ dữ liệu đầu vào đã qua xử lý. Tiến hành chuyển đổi các cột về dạng số để thực hiện xây dựng mô hình.



Hình 4.42 Dữ liệu sau khi chuyển đổi

Phân chia dữ liệu thành 2 tập x và y. Tập x chứa các biến độc lập không bao gồm cột Attrition. Tập y chỉ chứa 1 biến độc lập là Attrition.

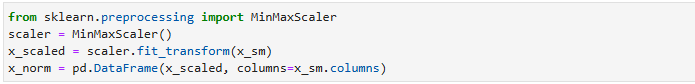
Vì bộ dữ liệu bị mất cân bằng sẽ ảnh hưởng tới việc đưa ra các dự đoán nên tương tự như các mô hình trên sẽ thực hiện việc cân bằng lại dữ liệu bằng phương pháp SMOTE. Đảm bảo cho mô hình sẽ đưa ra các dự đoán chính xác và ít sai lệch.

A screenshot of a computer

Description automatically generated

Hình 4.43 Chia dữ liệu và cân bằng dữ liệu

Sử dụng phương pháp chuẩn hóa Min-max để chuẩn hóa các biến về khoảng [0-1], để đảm bảo sự tương đồng giữa các.



Hình 4.44 Chuẩn hóa dữ liệu

### Huấn luyện mô hình

Trước khi huấn luyện mô hình sử dụng hàm train\_test\_split của thư viện scikit-learn để chia dữ liệu thành 2 tập riêng biệt 80% dữ liệu huấn luyện và 20% dữ liệu kiểm tra.



Hình 4.45 Chia dữ liệu

Import các thư viện cần thiết cho thuật toán Neural Network

Thư viện TensorFlow được sử dụng cho việc xây dựng và huấn luyện các mô hình học máy.

Keras là một API cấp cao chạy trên TensorFlow, giúp dễ dàng xây dựng và huấn luyện các mô hình học máy. Sequential là một lớp trong Keras, được sử dụng để khởi tạo một mô hình tuần tự, nơi các lớp được xếp chồng lên nhau theo thứ tự.

Dense: Đây là lớp kết nối đầy đủ (fully connected layer), nơi mỗi neuron trong lớp được kết nối với tất cả các neuron trong lớp trước và lớp sau. Lớp Dense thường được sử dụng trong các mạng nơ-ron nhân tạo. Dropout là một kỹ thuật regularization, được sử dụng để ngăn chặn overfitting bằng cách ngẫu nhiên bỏ đi một số đơn vị trong quá trình huấn luyện. Điều này giúp mô hình trở nên bền vững hơn và tránh việc quá khớp với dữ liệu huấn luyện.



Hình 4.46 Thư viện cho Neural Network

Khởi tạo mô hình Sequential. Sequential là một lớp trong Keras dùng để khởi tạo một mô hình tuần tự. Các lớp sẽ được thêm vào mô hình này theo thứ tự từ đầu đến cuối, và đầu ra của lớp trước sẽ là đầu vào của lớp kế tiếp.

Thêm một lớp Dense đầu tiên với 64 neuron cho phép mô hình học được các đặc trưng phức tạp từ dữ liệu đầu vào. Số lượng lớn neuron giúp mô hình có khả năng biểu diễn cao hơn và học được nhiều mối quan hệ phi tuyến tính. Bắt đầu với số lượng neuron lớn giúp mô hình có đủ công suất để bắt đầu học từ các đặc trưng của dữ liệu. Điều này quan trọng trong các bài toán phức tạp với nhiều feature.

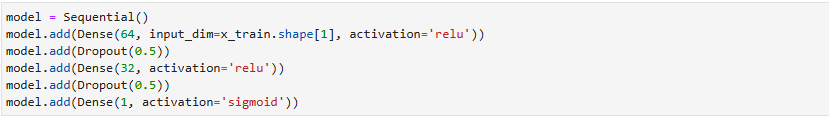
Dense(64) sẽ tạo một lớp kết nối đầy đủ với 64 neuron. input\_dim=x\_train.shape[1]: Xác định số lượng đầu vào của lớp này. x\_train.shape[1] cho biết số lượng feature trong dữ liệu huấn luyện (số cột của x\_train). Sử dụng hàm kích hoạt ReLU cho các neuron trong lớp này. Hàm kích hoạt ReLU giúp mô hình học các mối quan hệ phi tuyến tính giữa các feature và kết quả dự đoán.

Thêm một lớp Dropout với tỷ lệ 50%: Trong quá trình huấn luyện, ngẫu nhiên bỏ qua 50% các neuron của lớp này. Dropout giúp ngăn chặn mô hình bị overfitting bằng cách buộc mô hình phải học các biểu diễn mạnh mẽ và ít phụ thuộc vào một số lượng nhỏ các neuron.

Tiếp theo sẽ thêm một lớp Dense với 32 neuron. Giảm số lượng neuron từ 64 xuống 32 giúp giảm khả năng overfitting. Sau khi lớp đầu tiên đã học được các đặc trưng cơ bản, lớp thứ hai với số lượng neuron ít hơn sẽ học các đặc trưng tổng quát hơn và trừu tượng hơn. Giảm số lượng neuron cũng giúp giảm thời gian tính toán và tài nguyên yêu cầu trong quá trình huấn luyện.

Lớp Dense thứ hai này không cần chỉ định input\_dim vì nó tự động lấy kích thước đầu ra của lớp trước (64) làm kích thước đầu vào. Giống như lớp Dropout đầu tiên, lớp này không thay đổi kích thước của dữ liệu mà chỉ giảm overfitting bằng cách bỏ qua 50% số neuron trong quá trình huấn luyện.

Cuối cùng, thêm lớp đầu ra với 1 neuron. Tạo một lớp fully connected với 1 neuron. activation='sigmoid': Sử dụng hàm kích hoạt sigmoid cho neuron này. Hàm kích hoạt sigmoid chuyển đầu ra của neuron thành một giá trị trong khoảng [0, 1], rất phù hợp cho các bài toán phân loại nhị phân. Đầu ra này có thể được diễn giải như xác suất của một lớp nhất định.



Hình 4.47 Xây dựng mô hình ANN

Biên dịch mô hình với các tham số sau:

**Optimizer (Bộ tối ưu hóa)**: 'adam' là một thuật toán tối ưu hóa hiệu quả được sử dụng rộng rãi cho các mô hình học máy và học sâu. Adam kết hợp giữa hai phương pháp Gradient Descent và Root Mean Square Propagation. Nó giúp mô hình cập nhật các trọng số để giảm hàm mất mát.

**Loss function (Hàm mất mát)**: 'binary\_crossentropy' được sử dụng cho các bài toán phân loại nhị phân. Hàm mất mát này đánh giá sự khác biệt giữa nhãn thực tế và nhãn dự đoán của mô hình. Nó giúp hướng dẫn mô hình điều chỉnh trọng số để cải thiện dự đoán.

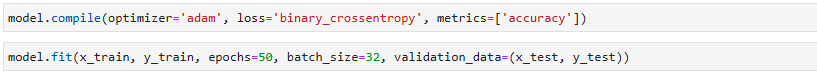
**Metrics (Thước đo đánh giá)**: 'accuracy' là thước đo để đánh giá hiệu suất của mô hình. Độ chính xác là tỷ lệ giữa số dự đoán đúng trên tổng số dự đoán. Trong quá trình huấn luyện và đánh giá mô hình, độ chính xác sẽ được tính toán và hiển thị.

Huấn luyện mô hình trên tập dữ liệu huấn luyện với các tham số:

**Epochs**: Số lần toàn bộ tập dữ liệu huấn luyện được đưa qua mô hình. Một epoch đại diện cho một vòng lặp qua toàn bộ dữ liệu huấn luyện. Ở đây, mô hình sẽ được huấn luyện trong 50 epoch.

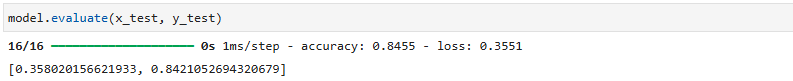
**Batch size**: Số lượng mẫu dữ liệu được đưa vào mô hình cùng một lúc trong mỗi lần cập nhật trọng số. Batch size nhỏ giúp cập nhật trọng số thường xuyên hơn, trong khi batch size lớn giúp tính toán hiệu quả hơn. Trong mô hình này mỗi batch chứa 32 mẫu dữ liệu.

**Validation data**: Tập dữ liệu kiểm tra được sử dụng để đánh giá hiệu suất của mô hình sau mỗi epoch huấn luyện. Giúp theo dõi độ chính xác của mô hình trên tập dữ liệu kiểm tra, không bị ảnh hưởng bởi việc học thuộc lòng dữ liệu huấn luyện.



Hình 4.48 Biên dịch và huấn luyện mô hình ANN

Đánh giá hiệu suất của mô hình huấn luyện trên tập kiểm tra, kết quả cho ra các chỉ số rất tốt. Quá trình đánh giá diễn ra nhanh chóng chỉ khoảng 1 mili giây. Độ chính xác của mô hình lên đến 84.55% một con số rất cao và hàm mất mát là 0.3551 giá trị loss càng nhỏ, mô hình càng tốt.



Hình 4.49 Đánh giá hiệu suất mô hình ANN

### Đánh giá mô hình

A screenshot of a computer screen

Description automatically generated

Hình 4.50 Các chỉ số đánh giá mô hình ANN

Thuật toán ANN cho ra được báo cáo phân loại như sau:

Độ chính xác (Accuracy): Mô hình có độ chính xác là 0.84, cho thấy mô hình dự đoán đúng 84% các trường hợp. Đây là một tỷ lệ rất cao, cho thấy mô hình phân biệt tốt giữa hai lớp mục tiêu.

Độ chính xác (Precision): Đối với lớp 0 nhân viên không rời bỏ có độ chính xác khoảng 81% và độ chính xác lớp 1 lớp nhân viên rời bỏ là 88%. Cho thấy mô hình dự đoán lớp 1 chính xác hơn so với lớp 0.

Độ nhạy (Recall): Đối với lớp 0 có độ nhạy là 89% và lớp 1 có độ nhạy 80%. Cho thấy mô hình có khả năng phát hiện và nhận dạng tốt hơn nhãn 0 so với nhãn 1.

Điểm F1 (F1-score): Là mức trung bình giữa độ chính xác và độ nhạy. Với điểm F1 cho lớp 0 là 85% và cho lớp 1 là 83%. Điểm F1 nằm trong khoảng 0 đến 1, với 1 là hiệu suất hoàn hảo. Cả hai lớp đều có điểm F1 trên 60% có thể đánh giá mô hình này có hiệu suất khá tốt.

A blue squares with white text

Description automatically generated

Hình 4.51 Mô hình Confusion Matrix của ANN

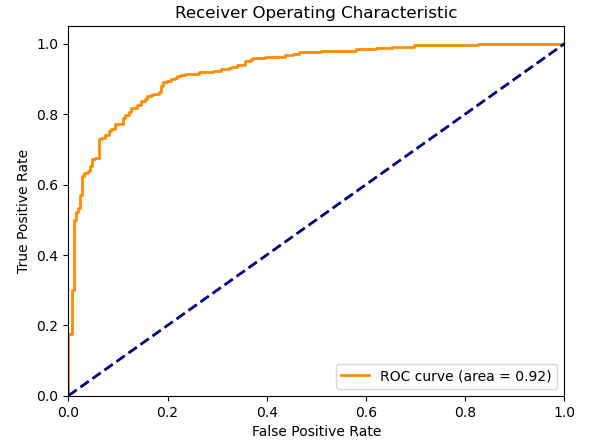
Nhận xét mô hình:

Precision của mô hình khoảng 87,6%, tức là trong số những nhân viên được dự đoán sẽ rời bỏ, có khoảng 87,6% là đúng. Thông số này cho thấy mô hình khá chính xác khi dự đoán nhân viên sẽ rời bỏ.

Recall là 79,7% tức là mô hình chỉ phát hiện được khoảng 79,7% số nhân viên thực sự rời bỏ. Tuy nhiên, có khoảng 20.3% các nhân viên có nguy cơ không được phát hiện, có thể dẫn đến việc không áp dụng kịp thời các biện pháp can thiệp.

Specificity là khoảng 88,7%, cho thấy mô hình phát hiện đúng khoảng 88,7% số nhân viên không rời bỏ. Điều này khá tốt nhưng vẫn còn nhiều nhân viên không rời bỏ bị dự đoán sai là rời bỏ (False Positives).

F1-Score khoảng 83.2%, tương tự như SVM đây là một chỉ số cao phản ánh mô hình không chỉ chính xác trong việc xác định các trường hợp nhân viên rời bỏ mà còn tương đối tốt trong việc phát hiện hầu hết các trường hợp nhân viên rời bỏ.



Hình 4.52 Mô hình ROC

Diện tích dưới đường cong (Area Under Curve - AUC) là 0,91, cho thấy mô hình có khả năng phân biệt rất tốt giữa các lớp. Một giá trị AUC càng gần 1 thì mô hình càng tốt.

Đường cong bắt đầu từ điểm (0,0), tức là khi không có dương tính nào được phát hiện, và kết thúc ở điểm (1,1), tức là tất cả các trường hợp dương tính và âm tính đều được phát hiện chính xác. Đường cong nằm trên đường chéo, thể hiện hiệu suất của mô hình tốt hơn so với phân loại ngẫu nhiên.

Đường cong tăng liên tục, nghĩa là khi tỷ lệ dương tính sai tăng, tỷ lệ dương tính đúng cũng tăng tương ứng. Điều này cho thấy mô hình có khả năng phân biệt giữa các lớp dữ liệu.

Đường cong càng gần góc trái trên (điểm [0,1]), thì hiệu suất của mô hình càng tốt, vì lúc này tỷ lệ dương tính đúng cao trong khi tỷ lệ dương tính sai thấp.

Qua các chỉ số ta có thể thấy được mô hình có hiệu suất tổng thể rất tốt trong việc dự đoán sự rời bỏ của nhân viên. Nó không chỉ phân loại chính xác các trường hợp rời bỏ và không rời bỏ mà còn đạt được sự cân bằng tốt giữa các chỉ số quan trọng như precision, recall, và F1 score. Để nâng cao hiệu quả của mô hình, mô hình xem xét việc cải thiện độ nhạy để giảm thiểu số lượng các trường hợp nhân viên có nguy cơ rời bỏ bị bỏ sót.

## Đánh giá và đưa ra kết luận

Trong quá trình xây dựng các mô hình dự đoán sự rời bỏ của nhân viên công ty IBM, bốn mô hình đã được đánh giá dựa trên các chỉ số khác nhau.

Mô hình Logistic Regression cho thấy Accuracy 60.7%, Precision 61.9%, Recall 55.4%, Specificity 65.9%, F1-score 58.5% và AUC 0,65. Mô hình này có độ chính xác và AUC thấp nhất trong số các mô hình. Điều này cho thấy khả năng phân loại của mô hình này không tốt, đặc biệt là trong việc nhận diện các trường hợp rời bỏ.

Mô hình Naïve Bayes có Accuracy cao hơn với 63.8%, Precision 59.8%, Recall 83.8%, Specificity 43.7%, F1-score 69.8% và AUC 0,72. Mô hình này có độ nhạy rất cao, có nghĩa là nó có khả năng phát hiện nhiều trường hợp rời bỏ hơn. Tuy nhiên, độ chính xác chỉ ở mức trung bình, và điểm F1 cũng không cao bằng các mô hình khác. Điều này cho thấy Naïve Bayes có thể tốt trong việc nhận diện các trường hợp rời bỏ, nhưng không cân bằng tốt giữa độ chính xác và độ nhạy.

Mô hình SVM đạt được Accuracy 82%, Precision 84.4%, Recall 78.5%, Specificity 85.4%, F1 - score 81.3% và AUC-ROC 0.89. Qua các chỉ số này mô hình có độ chính xác và AUC khá cao, cho thấy khả năng phân loại rất tốt. Độ chính xác của lớp dương và điểm F1 đều cao, chỉ ra rằng mô hình SVM có sự cân bằng tốt giữa độ nhạy và độ chính xác của lớp dương. Đây là một mô hình mạnh trong việc dự đoán sự rời bỏ của nhân viên.

Cuối cùng, mô hình ANN có hiệu suất tốt nhất với Accuracy 84.2%, Precision 87.6%, Recall 79.7%, Specificity 88.7%, F1-score 83.2% và AUC 0.92. Đây là mô hình có hiệu suất tốt nhất trong số các mô hình đã đánh giá. Độ chính xác, điểm F1, và AUC đều cao nhất, chỉ ra khả năng phân loại xuất sắc.

Trong số các mô hình đã được đánh giá, mô hình ANN và SVM có hiệu suất tốt nhất cho việc dự đoán sự rời bỏ của nhân viên. Mô hình ANN đặc biệt nổi bật với độ chính xác và AUC cao nhất, chỉ ra khả năng phân loại và phân biệt tốt của mô hình. Mô hình SVM cũng là một lựa tốt với sự cân bằng tốt giữa độ chính xác và độ nhạy. Mô hình Logistic Regression và Naïve Bayes có hiệu suất thấp hơn, với Logistic Regression có hiệu suất thấp nhất trong số bốn mô hình.

Bảng 4.1 Tổng hợp các mô hình và tham số

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chỉ số  Mô hình | Accuracy | Precision | Recall | Specificity | F1-score | AUC |
| Logistic Regression | 60.7% | 61.9% | 55.4% | 65.9% | 58.5% | 0.65 |
| Naïve Bayes | 63.8% | 59.8% | 83.8% | 43.7% | 69.8% | 0.72 |
| Support Vector Machine | 82% | 84.4% | 78.5% | 85.4% | 81.3% | 0.89 |
| Neural Network (ANN) | 84.2% | 87.6% | 79.7% | 88.7% | 83.2% | 0.92 |

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau khi hoàn thành quá trình nghiên cứu bài báo cáo chúng em đã tích lũy được rất nhiều kiến thức mới và hữu ích từ các thuật toán được áp dụng trong bài báo cáo, biết được những yếu tố có thể ảnh hưởng đến hành vi rời bỏ công ty của các nhân viên và những thách thức mà công ty phải đối mặt khi nhân viên rời bỏ công ty thông qua việc phân tích tập dữ liệu. Từ đó, nhóm chúng em đã thành công xây dựng được 4 mô hình dự đoán sự rời bỏ của nhân viên giúp cho công ty có thể hạn chế được những rủi ro có thể gặp phải khi nhân viên rời bỏ. Mô hình tốt nhất công ty có thể sử dụng là mô hình ANN với AUC là 0.92. Ngoài ra, việc thực hiện bài phân tích này đã giúp chúng em nâng cao kỹ năng sử dụng các công cụ phân tích như Power BI và Python, cải thiện kỹ năng tiền xử lý dữ liệu để giúp cho việc phân tích trở nên hiệu quả và mạng tính chính xác cao hơn, tích lũy kiến thức liên quan đến machine learning, các mô hình, phương pháp để xây dựng một mô hình dự đoán các vấn đề trong tương lai. Bên cạnh đó, chúng em còn xây dựng được những mối quan hệ mới với các anh chị hướng dẫn, góp phần vào thành công của báo cáo thực tập hè lần này. Ngoài kiến thức chuyên môn, chúng em còn phát triển và rèn luyện nhiều kỹ năng mềm như quản lý thời gian, kỷ luật bản thân trong quá trình nghiên cứu, xây dựng mối quan hệ mới và biết cách thích nghi hòa nhập với văn hóa công ty.

Trong quá trình thực hiện nghiên cứu lần này, chúng em đã gặp phải một số hạn chế đáng chú ý. Mặc dù đã học qua các lý thuyết về mô hình học máy và phân tích dữ liệu, chúng em vẫn gặp khó khăn khi áp dụng những kiến thức này vào dự án thực tế. Hơn nữa, khả năng phân tích dữ liệu của nhóm còn hạn chế.Và thời gian nghiên cứu ngắn đã khiến cho việc thu thập và xử lý dữ liệu, lựa chọn mô hình phù hợp trở nên khó khăn.

Từ những điểm mạnh và điểm hạn chế của mỗi cá nhân chúng em được nêu trong báo cáo này, chúng em mong rằng trong tương lai sẽ có thêm nhiều cơ hội để tham gia vào các dự án thực tế. Những trải nghiệm này sẽ giúp chúng em không chỉ áp dụng và phát huy những kiến thức đã học mà còn mở rộng thêm hiểu biết mới, từ đó nâng cao và hoàn thiện kỹ năng của bản thân. Chúng em tin rằng việc tiếp xúc với những thách thức thực tiễn sẽ giúp chúng em trưởng thành hơn, rút ra bài học quý giá và đạt được những thành tựu tốt hơn trong sự nghiệp.

TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | V. -. IT, "VNPT - IT," [Online]. Available: https://vnptit.vn/. |
| [2] | S. SOLUTIONS, "STI SOLUTIONS," [Online]. Available: https://stisolutions.net/. |
| [3] | 200lab, "Jupyter Notebook là gì?," [Online]. Available: https://200lab.io/blog/jupyter-notebook-la-gi/#2-l%E1%BB%A3i-%C3%ADch-m%C3%A0-jupyter-notebook-mang-l%E1%BA%A1i. |
| [4] | F. Shop, "Python là gì?," [Online]. Available: https://fptshop.com.vn/tin-tuc/danh-gia/python-la-gi-168825. |
| [5] | T. Dev, "Python Là Gì?," [Online]. Available: https://topdev.vn/blog/python-la-gi/. |
| [6] | BASE.VN, "Power BI là gì?," [Online]. Available: https://base.vn/blog/power-bi-la-gi/. |
| [7] | F. Shop, "Power BI là gì? Tại sao các doanh nghiệp nên sử dụng Power BI," [Online]. Available: https://fptshop.com.vn/tin-tuc/danh-gia/power-bi-la-gi-tai-sao-cac-doanh-nghiep-nen-su-dung-power-bi-146601. |
| [8] | aws, "Hồi quy logistic là gì?," [Online]. Available: https://aws.amazon.com/vi/what-is/logistic-regression/. |
| [9] | noron, "Giới thiệu về Naive Bayes trong Machine Learning," [Online]. Available: https://www.noron.vn/post/gioi-thieu-ve-naive-bayes-trong-machine-learning-40ww191xxs87. |
| [10] | D. Nguyen, "SVM, Soft Margin SVM, Kernel SVM, Multi-class SVM," [Online]. Available: https://viblo.asia/p/svm-soft-margin-svm-kernel-svm-multi-class-svm-W13VMer0VY7. |
| [11] | ITNavi, "Tổng quan về Neural Network(mạng Nơ Ron nhân tạo) là gì?," [Online]. Available: https://itnavi.com.vn/blog/neural-network-la-gi/. |
| [12] | VIBLO, "Tìm hiểu về Confusion matrix trong Machine Learning?," [Online]. Available: https://viblo.asia/p/tim-hieu-ve-confusion-matrix-trong-machine-learning-Az45bRpo5xY. |
| [13] | noron, "Tìm hiểu chi tiết về AUC - ROC trong Machine Learning?," [Online]. Available: https://www.noron.vn/post/tim-hieu-chi-tiet-ve-auc---roc-trong-machine-learning-1fz9nhqo5ut. |

CHECK LIST CỦA BÁO CÁO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nội dung công việc** | **Có** | **Không** | **Ghi chú** |
| 1 | Báo cáo được trình bày (định dạng) đúng với yêu cầu. | x |  |  |
| 2 | Báo cáo có số lượng trang đáp ứng đúng yêu cầu (30-50 trang) |  | x |  |
| 3 | Báo cáo trình bày được phần mở đầu bao gồm: Mục tiêu, Phạm vi và đối tượng, kết cấu … | x |  |  |
| 4 | Báo cáo trình bày về công ty, vị trí việc làm (công việc đó làm gì, kiến thức và kỹ năng cần thiết là gì, con đường phát triển sự nghiệp (career path)), cơ sở lý thuyết phù hợp với nội dung của đề tài (Tối đa 10-12 trang) | x |  |  |
| 5 | Báo cáo có sản phẩm cụ thể phù hợp với mục tiêu đặt ra của đề tài | x |  |  |
| 6 | Báo cáo có phần kết luận và hướng phát triển của đề tài | x |  |  |

PHỤ LỤC