ĐẠI HỌC UEH

TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ

KHOA CÔNG NGHỆ THÔNG TIN KINH DOANH

TIỂU LUẬN CUỐI KÌ

MÔN HỌC: KHOA HỌC DỮ LIỆU

ĐỀ TÀI:
DỰ BÁO TÌNH TRẠNG ĐẶT PHÒNG CỦA KHÁCH SẠN

Giảng viên hướng dẫn:	TS. Huỳnh Văn Đức
Mã LHP:	24D1INF50905917
Nhóm sinh viên thực hiện	Nhóm G02
Trần Thế Hào	31221026239
Ngô Mai Kim Huyền	31221022317
Phan Thúy Ngân	31221022358
Phạm Thái Nguyên	31221022252
Nguyễn Thị Xuân Nhi	31221023789
Lê Như Thi	31221023107

Thành phố Hồ Chí Minh, ngày 11 tháng 03 năm 2024

MỤC LỤC

LỜI CẢM ƠN **4**

DANH MỤC BẢNG BIỂU **5**

DANH MỤC HÌNH ẢNH **7**

BẢNG PHÂN CÔNG CÔNG VIỆC **10**

CHƯƠNG I: TỔNG QUAN ĐỀ TÀI NGHIÊN CỨU **11**

1.1. Lý do chọn đề tài 11

1.2. Mục tiêu nghiên cứu 11

1.3. Đối tượng và phạm vi nghiên cứu 11

1.4. Phương pháp nghiên cứu 12

CHƯƠNG II: CƠ SỞ LÝ THUYẾT **14**

2.1. Khai phá dữ liệu 14

2.1.1. Khái niệm 14

2.1.2. Ứng dụng 14

2.1.3. Phương pháp và công cụ khai phá 14

2.2. Kỹ thuật phân lớp 16

2.2.1. Khái niệm 16

2.2.2. Các phương pháp phân lớp 16

2.3. Phương pháp đánh giá mô hình phân lớp 16

2.3.1. Ma trận nhầm lẫn (Confusion Matrix) và độ chính xác (Accuracy); ROC, AUC, Precision/Recall 16

2.3.2. Cross Validation: Holdout và K-fold cross validation 17

2.3.3. Độ chính xác (Accuaracy) 18

2.3.4. Precision, Recall, F1-score 18

2.3.5. ROC và AUC 19

CHƯƠNG III: MÔ HÌNH VÀ THỰC HIỆN NGHIÊN CỨU **20**

3.1. Mô tả bộ dữ liệu 20

3.2. Tiền xử lý dữ liệu 23

3.3. Phân tách dữ liệu 25

3.4. Đề xuất mẫu 26

3.5. Trích xuất mẫu 28

3.5.1. Kỹ thuật Decision Tree – Phan Thúy Ngân 29
3.5.2. Kỹ thuật Support Vector Machine (SVM) – Ngô Mai Kim Huyền.....39
3.5.3. Kỹ thuật Random Forest – Phạm Thái Nguyên 47
3.5.4. Kỹ thuật Naive Bayes – Nguyễn Thị Xuân Nhi.....54
3.5.5. Kỹ thuật Logistic Regression – Lê Như Thi 61
3.5.6. Kỹ thuật k-Nearest Neighbors (kNN) – Trần Thế Hào.....67
3.6. Xây dựng mô hình 74
3.7. Kết quả và đánh giá 75
3.7.1. Theo kỹ thuật cá nhân 75
3.7.2. Kết quả và đánh giá chung 83
CHƯƠNG IV: KẾT LUẬN 85
4.1. Kết luận.....85
4.2. Hạn chế 85
4.3. Giải pháp.....85
TÀI LIỆU THAM KHẢO 87

LỜI CẢM ƠN

Đầu tiên, nhóm G02 chúng em xin bày tỏ lòng biết ơn sâu sắc tới thầy TS. Huỳnh Văn Đức, giảng viên của bộ môn Khoa học dữ liệu. Sự quan tâm không mệt mỏi, sự hỗ trợ tận tình và những bài giảng đầy cảm hứng của thầy đã là nguồn động viên lớn lao, giúp chúng em vượt qua mọi khó khăn trong hành trình khám phá bộ môn học này. Không chỉ là những bài học lý thuyết, thầy còn mở ra cho chúng em cánh cửa trải nghiệm thực tế, nơi chúng em được áp dụng kiến thức vào cuộc sống, nhận ra giá trị của việc học không ngừng nghỉ. Mặc dù ban đầu gặp không ít thử thách, nhưng với sự kiên trì và nỗ lực, hiện tại chúng em đã nắm bắt được nhiều kiến thức quý giá. Mỗi thử thách chúng em gặp phải không chỉ là một bài học mà còn là một cơ hội để chúng em mạnh mẽ và trưởng thành hơn.

Bài tiểu luận này là minh chứng cho quá trình nỗ lực không ngừng của nhóm chúng em, là bức tranh tổng hợp từng mảnh ghép kiến thức mà thầy đã kiên nhẫn truyền đạt. Chúng em hiểu rằng, trên con đường học vấn này, sẽ còn nhiều lỗi lầm và thiếu sót. Chính vì vậy, chúng em rất mong muốn nhận được những lời khuyên chân thành và sự chỉ bảo từ thầy, để từng bước hoàn thiện hơn nữa.

Lời cuối cùng, chúng em sẽ luôn ghi nhớ và trân trọng những điều thầy đã dạy. Chúng em xin chân thành cảm ơn thầy vì tất cả.

DANH MỤC BẢNG BIỂU

Bảng 3.1: Mô tả thuộc tính của bộ dữ liệu 23
Bảng 3.2: Bộ dữ liệu huấn luyện Tree 32
Bảng 3.3: Bộ dữ liệu kiểm định Tree 33
Bảng 3.4: Quang cảnh: Nắng 34
Bảng 3.5: Quang cảnh: Âm u 34
Bảng 3.6: Quang cảnh: Mưa 34
Bảng 3.7: Nhiệt độ: Nóng 35
Bảng 3.8: Nhiệt độ: Ôn hòa 35
Bảng 3.9: Nhiệt độ: Mát 36
Bảng 3.10: Child node quang cảnh = âm u 36
Bảng 3.11: Child node quang cảnh = nắng 37
Bảng 3.12: Child node quang cảnh = mưa 37
Bảng 3.13: Kết quả tính thủ công Tree 38
Bảng 3.14: Mô tả thuộc tính SVM 42
Bảng 3.15: Bộ dữ liệu huấn luyện SVM 42
Bảng 3.16: Chuyển đổi dữ liệu 43
Bảng 3.17: Bộ dữ liệu kiểm định SVM 45
Bảng 3.18: Kết quả phân lớp SVM 45
Bảng 3.19: Bộ dữ liệu huấn luyện Random Forest 50
Bảng 3.20: Bộ dữ liệu kiểm định Random Forest 51
Bảng 3.21: Kết quả phân lớp thủ công Random Forest 53
Bảng 3.22: Bộ dữ liệu huấn luyện Naive Bayes 57
Bảng 3.23: Bộ dữ liệu kiểm định Naive Bayes 57
Bảng 3.24: Xác suất theo thuộc tính “Mây” 58
Bảng 3.25: Xác suất theo thuộc tính “Áp suất” 58
Bảng 3.26: Xác suất theo thuộc tính “Gió” 58
Bảng 3.27: Xác suất theo thuộc tính “Kết quả” 58
Bảng 3.28: Kết quả xác suất phân lớp Naive Bayes 59
Bảng 3.29: Bộ dữ liệu huấn luyện Logistic Regression 64
Bảng 3.30: Bộ dữ liệu kiểm định Logistic Regression 65

Bảng 3.31: Kết quả xác suất phân lớp thủ công Logistic Regression.....66
Bảng 3.32: Bộ dữ liệu huấn luyện kNN.....70
Bảng 3.33: Bộ dữ liệu kiểm định kNN 70
Bảng 3.34: Xác định lượng Nearest Neighbor.....72
Bảng 3.35: Kết quả phân lớp thủ công kNN.....72

DANH MỤC HÌNH ẢNH

Hình 2.1: Ma trận nhầm lẫn (Confusion Matrix) 17
Hình 2.2: Bảng so sánh hai phương pháp Holdout và K-fold Cross Validation 18
Hình 2.3: Minh họa đường cong ROC 19
Hình 2.4: Minh họa AUC 19
Hình 3.1: Dùng Feature Statistics quan sát tổng quan bộ dữ liệu 24
Hình 3.2: Thao tác phân loại dữ liệu ngoại lai 24
Hình 3.3: Sau khi phân loại dữ liệu ngoại lai 25
Hình 3.4: Mô tả phân tách bộ dữ liệu 25
Hình 3.5: Các bước tiền xử lý và phân tách dữ liệu 26
Hình 3.6: Quan sát thuộc tính Lead_time qua Boxplot 26
Hình 3.7: Quan sát thuộc tính No_of_previous_bookings _not_ canceled qua Distrubution 27
Hình 3.8: Quan sát Lead_time và No_of_previous_bookings _not_ canceled qua Scatter plot 27
Hình 3.9: Chạy dữ liệu lần đầu 28
Hình 3.10: Kết quả các chỉ số của widget Test and Score 28
Hình 3.11: Minh họa Decision Tree 29
Hình 3.12: Vẽ cây quyết định 37
Hình 3.13: Kết quả so sánh với Orange Tree 38
Hình 3.14: Xây dựng mô hình theo Decision Tree 38
Hình 3.15: Tree Widget 39
Hình 3.16: Kỹ thuật Support Vector Machine 40
Hình 3.17: Kết quả so sánh với Orange SVM 45
Hình 3.18: Xây dựng mô hình theo SVM 46
Hình 3.19: SVM Widget 46
Hình 3.20: Sơ đồ biểu diễn các cây quyết định trong phương pháp Random Forest 47
Hình 3.21: 10 cây quyết định ngẫu nhiên 52
Hình 3.22: Kết quả so sánh Orange Random Forest 53
Hình 3.23: Xây dựng mô hình theo Random Forest 53
Hình 3.24: Random Forest Widget 53

Hình 3.25: Mô tả bước xây dựng mô hình phân lớp 55
Hình 3.26: Kết quả so sánh với Orange Naive Bayes 60
Hình 3.27: Xây dựng mô hình theo Naive Bayes 60
Hình 3.28: Naive Bayes Widget 60
Hình 3.29: Hệ số ước lượng thông qua SPSS 65
Hình 3.30: Kết quả so sánh Orange Logistic Regression 66
Hình 3.31: Xây dựng mô hình theo Logistic Regression 66
Hình 3.32: Logistic Regression Widget 67
Hình 3.33: Kết quả so sánh Orange kNN 73
Hình 3.34: Xây dựng mô hình theo kNN 73
Hình 3.35: kNN Widget 73
Hình 3.36: Mô hình phân lớp dữ liệu để dự báo tình trạng đặt phòng của khách sạn 74
Hình 3.37: Kết quả Test and Score kỹ thuật Tree 75
Hình 3.38: Kết quả Confusion Matrix của Tree 76
Hình 3.39: Kết quả dự báo bằng kỹ thuật Tree 76
Hình 3.40: Kết quả Test and Score kỹ thuật SVM 77
Hình 3.41: Kết quả Confusion Matrix của SVM 77
Hình 3.42: Kết quả dự báo bằng kỹ thuật SVM 77
Hình 3.43: Kết quả Test and Score của Random Forest 78
Hình 3.44: Kết quả Confusion Matrix của Random Forest 78
Hình 3.45: Kết quả dự báo bằng kỹ thuật Random Forest 79
Hình 3.46: Kết quả Test and Score của Naive Bayes 79
Hình 3.47: Kết quả Confusion Matrix của Naive Bayes 79
Hình 3.48: Kết quả dự báo dựa trên phương pháp Naive Bayes 80
Hình 3.49: Kết quả Test and Score của Logistic Regression 80
Hình 3.50: Kết quả Confusion Matrix của Logistic Regression 81
Hình 3.51: Kết quả dự báo dựa trên phương pháp Logistic Regression 81
Hình 3.52: Kết quả Test and Score của kNN 81
Hình 3.53: Kết quả Confusion Matrix của kNN 82
Hình 3.54: Kết quả dự báo dựa trên phương pháp kNN 82
Hình 3.55: Kết quả Test and Score của mô hình 83

Hình 3.56: Kết quả Confusion Matrix của mô hình.....83
Hình 3.57: Kết quả dự báo 84

BẢNG PHÂN CÔNG CÔNG VIỆC

STT	Họ và tên	MSSV	Công việc	Đánh giá
1	Trần Thế Hào	31221026239	- Tổng quan đề tài nghiên cứu - Cơ sở lý thuyết - Kỹ thuật kNN - Xây dựng mô hình và đánh giá	100%
2	Ngô Mai Kim Huyền	31221022317	- Tổng quan đề tài nghiên cứu - Cơ sở lý thuyết - Kỹ thuật SVM - Xây dựng mô hình và đánh giá	100%
3	Phan Thúy Ngân	31221022358	- Cơ sở lý thuyết - Kỹ thuật Decision Tree - Xây dựng mô hình và đánh giá	100%
4	Phạm Thái Nguyên	31221022252	- Lời cảm ơn - Tổng quan đề tài nghiên cứu - Cơ sở lý thuyết - Kỹ thuật Random Forest - Xây dựng mô hình và đánh giá	100%
5	Nguyễn Thị Xuân Nhi	31221023789	- Cơ sở lý thuyết - Kỹ thuật Naive Bayes - Xây dựng mô hình và đánh giá	100%
6	Lê Như Thi	31221023107	- Cơ sở lý thuyết - Kỹ thuật Logistic Regression - Xây dựng mô hình và đánh giá	100%

CHƯƠNG I: TỔNG QUAN ĐỀ TÀI NGHIÊN CỨU

1.1. Lý do chọn đề tài

Trong bối cảnh công nghệ ngày nay đang chứng kiến sự bùng nổ của khoa học dữ liệu, không có gì ngạc nhiên khi lĩnh vực khách sạn, như một phần quan trọng của nền kinh tế, đặt mình vào vị trí không thể thiếu điều này. Quản lý một khách sạn không chỉ là việc đơn giản "đặt phòng và chờ khách đến", mà đòi hỏi sự nhạy bén đặc biệt về hành vi của khách hàng và khả năng linh hoạt trong việc tối ưu hóa mọi khía cạnh của hoạt động.

Chính vì lý do này, chúng tôi đã quyết định chọn đề tài "Phân tích dữ liệu đặt phòng khách sạn". Điều mà chúng tôi muốn khám phá là tình trạng hủy đặt phòng - một thách thức thực tế và quan trọng đối với quản lý khách sạn. Sự hiểu biết sâu sắc về lý do và xu hướng hủy đặt phòng không chỉ giúp khách sạn áp dụng các chiến lược và biện pháp một cách hiệu quả mà còn nâng cao hiệu suất kinh doanh và trải nghiệm của khách hàng.

Đối với chúng tôi, không chỉ là việc dự đoán hủy đặt phòng mà còn là khám phá những yếu tố ẩn sau quyết định của khách hàng. Qua đó, chúng tôi muốn mang lại giá trị thực tế cho khách sạn bằng cách hỗ trợ họ đưa ra những chiến lược động và đáp ứng nhanh chóng với sự biến động của thị trường.

Bên cạnh đó, đề tài của chúng tôi không chỉ là một cuộc nghiên cứu mà còn là một góc nhìn mới về việc sử dụng khoa học dữ liệu trong ngành khách sạn. Chúng tôi hy vọng rằng những kết quả thu được có thể góp phần vào sự phát triển của lĩnh vực này trong bối cảnh thời đại số hóa ngày nay.

Nhìn chung, nghiên cứu về tình trạng hủy đặt phòng không chỉ đưa ra thông tin quan trọng cho quản lý khách sạn mà còn hỗ trợ phát triển ngành du lịch và khách sạn ở Việt Nam, nơi mà sự cạnh tranh ngày càng trở nên khốc liệt và đòi hỏi sự linh hoạt và sáng tạo.

1.2. Mục tiêu nghiên cứu

Mục tiêu tổng quát: Dự đoán được một cách tốt nhất về khả năng phòng có bị hủy hay không để đưa ra các chiến lược hiệu quả và phù hợp với điều kiện thực tế của khách sạn. Tìm ra các yếu tố ảnh hưởng đến tình trạng hủy đặt phòng, từ đó giúp giảm tình trạng khách hàng hủy đơn đặt phòng.

Mục tiêu cụ thể: Mục tiêu nghiên cứu chính của đề tài ứng với bài toán cần giải quyết.

Bài toán: Dự đoán quyết định đặt phòng của khách sạn có bị hủy hay không. Nghiên cứu này sẽ giúp cho người quản lý biết được các yếu tố ảnh hưởng nhiều đến hành vi khách hàng, giúp tìm hiểu được nguyên nhân của tình trạng hủy đặt phòng trên. Từ đó giúp khách sạn có thể đưa ra biện pháp phù hợp để giảm thiểu tình trạng này, tránh gây thất thoát về doanh thu.

1.3. Đối tượng và phạm vi nghiên cứu

- Đối tượng nghiên cứu: Nghiên cứu liên quan đến các phòng của khách sạn có bị hủy hay không bị hủy.

- Phạm vi nghiên cứu: Tập dữ liệu bao gồm thông tin dữ liệu thô chứa 36275 hàng dữ liệu (chi tiết đặt phòng của khách hàng) và 18 cột (thuộc tính) được lấy từ web kaggle.

1.4. Phương pháp nghiên cứu

- *Phương pháp nghiên cứu lý luận:*

Xây dựng mô hình lý thuyết: Trước khi tiến hành thu thập và phân tích dữ liệu, việc xây dựng một khung lý thuyết về các yếu tố ảnh hưởng đến quá trình đặt phòng khách sạn có thể được thực hiện. Các yếu tố này có thể bao gồm giá cả, loại phòng do khách hàng đặt, số ngày từ ngày đặt phòng đến ngày nhận phòng, số đêm cuối tuần, số đêm trong tuần, loại gói bữa ăn do khách hàng đặt và các yếu tố khác. Việc xây dựng khung lý thuyết này sẽ cung cấp một cấu trúc cho việc phân tích dữ liệu và giải thích các mối quan hệ giữa các yếu tố.

Thu thập dữ liệu: Sau khi hoàn thiện khung lý thuyết, nghiên cứu viên có thể tiến hành thu thập dữ liệu về đặt phòng khách sạn, bao gồm các thông tin về khách hàng, số lượng người lớn, số lượng trẻ em, giá cả và các yếu tố khác.

Phân tích dữ liệu: Dữ liệu thu thập có thể được phân tích bằng các phương pháp thống kê để kiểm chứng các giả thuyết trong mô hình và xác định mối quan hệ giữa các yếu tố. Các phương pháp phân tích dữ liệu có thể bao gồm phân tích tương quan, phân tích hồi quy và phân tích đa biến.

Kiểm chứng mô hình lý thuyết: Cuối cùng, sau khi phân tích dữ liệu, việc kiểm chứng mô hình lý thuyết có thể được thực hiện bằng cách so sánh các dự báo từ mô hình với kết quả thực tế. Nếu kết quả dự báo tương đồng với kết quả thực tế, mô hình lý thuyết có thể được coi là hợp lý và có thể được sử dụng để dự báo các kết quả trong tương lai.

- *Phương pháp nghiên cứu thực tiễn:*

Để phân tích dữ liệu và giải quyết vấn đề nghiên cứu, các thuật toán khai phá dữ liệu được áp dụng, cùng với việc sử dụng công cụ Orange - một nền tảng hình ảnh dễ sử dụng để thực hành các thuật toán học máy và khám phá dữ liệu phổ biến hiện nay.

Thu thập dữ liệu: Nghiên cứu viên có thể thực hiện thu thập từ nhiều nguồn khác nhau như các trang web đặt phòng, hệ thống quản lý khách sạn hoặc cuộc khảo sát khách hàng. Các yếu tố này có thể bao gồm giá cả, loại phòng do khách hàng đặt, số ngày từ ngày đặt phòng đến ngày nhận phòng, số đêm cuối tuần, số đêm trong tuần, loại gói bữa ăn do khách hàng đặt và các yếu tố khác.

Xử lý dữ liệu: Sau khi thu thập dữ liệu, quá trình xử lý dữ liệu được tiến hành bằng các công cụ phân tích dữ liệu để trích xuất thông tin quan trọng và chuẩn bị dữ liệu cho phân tích.

Phân tích dữ liệu: Dữ liệu được phân tích bằng các kỹ thuật thống kê như phân tích tương quan, phân tích hồi quy và phân tích đa biến, với mục tiêu tìm ra các mối quan hệ giữa các yếu tố ảnh hưởng đến quá trình đặt phòng khách sạn. Những kết quả này có thể áp dụng cho việc dự báo đặt phòng, hủy phòng hoặc các mục tiêu khác quản lý khách sạn.

Đưa ra kết luận và khuyến nghị: Cuối cùng, dựa trên kết quả phân tích, nghiên cứu viên có thể đưa ra kết luận và khuyến nghị cho các nhà quản lý khách sạn để cải thiện hoạt động kinh doanh của họ. Các khuyến nghị này có thể liên quan đến giá cả, quảng cáo, chính sách hủy phòng, cải thiện trải nghiệm khách hàng hoặc các yếu tố khác.

CHƯƠNG II: CƠ SỞ LÝ THUYẾT

2.1. Khai phá dữ liệu

2.1.1. Khái niệm

Data mining là quá trình phân tích dữ liệu từ các nguồn khác nhau để khám phá các mẫu, thông tin ẩn và mối quan hệ có thể giúp hiểu rõ hơn về hành vi và xu hướng của dữ liệu. Cụ thể, data mining sử dụng các phương pháp và công cụ máy tính để khám phá và phân tích lượng lớn dữ liệu từ các cơ sở dữ liệu, kho dữ liệu, và nguồn thông tin khác nhau. Mục tiêu chính của data mining là tìm ra thông tin có giá trị và tri thức từ dữ liệu, giúp hỗ trợ quyết định và dự đoán trong các lĩnh vực như kinh doanh, y tế, khoa học, và nhiều lĩnh vực khác.

2.1.2. Ứng dụng

Data mining có rất nhiều ứng dụng trong các lĩnh vực khác nhau. Dưới đây là một số ứng dụng phổ biến của data mining:

- Kinh doanh và tiếp thị: Data mining được sử dụng để phân tích dữ liệu khách hàng và dự đoán hành vi mua hàng, giúp các doanh nghiệp hiểu rõ hơn về khách hàng của mình và tối ưu hóa chiến lược tiếp thị.
- Y tế: Trong lĩnh vực y tế, data mining được sử dụng để phân tích dữ liệu bệnh nhân và dự đoán nguy cơ bệnh tật, phân loại bệnh, tối ưu hóa quy trình chẩn đoán, và nghiên cứu dược lý.
- Tài chính: Data mining được áp dụng trong lĩnh vực tài chính để phân tích dữ liệu thị trường, dự đoán xu hướng tài chính, phát hiện gian lận tài chính và quản lý rủi ro.
- Khoa học và nghiên cứu: Trong lĩnh vực khoa học và nghiên cứu, data mining được sử dụng để khám phá mẫu trong dữ liệu thí nghiệm và phân tích dữ liệu khoa học phức tạp.
- Hệ thống hỗ trợ quyết định: Data mining cung cấp thông tin và tri thức từ dữ liệu để hỗ trợ quyết định trong nhiều lĩnh vực, bao gồm quản lý chuỗi cung ứng, dự đoán thời tiết, và quản lý tài nguyên.
- Chăm sóc khách hàng: Data mining được sử dụng để phân tích dữ liệu phản hồi từ khách hàng, đánh giá sự hài lòng của khách hàng, và cải thiện dịch vụ khách hàng.
- Giáo dục: Trong lĩnh vực giáo dục, data mining được sử dụng để phân tích dữ liệu học sinh và hiểu rõ hơn về cách mà họ học, từ đó cải thiện quy trình giảng dạy và đào tạo.

Những ứng dụng này chỉ là một phần nhỏ của những cách mà data mining có thể được sử dụng trong thực tế, và có thể được áp dụng ở nhiều lĩnh vực khác nhau để đem lại lợi ích và giá trị.

2.1.3. Phương pháp và công cụ khai phá

2.1.3.1. Phương pháp khai phá

- ***Kỹ thuật phân tích phân loại (Classification Analysis)***

Được sử dụng để phân loại các mẫu dữ liệu vào các nhóm đã được xác định trước. Các kỹ thuật phân tích phân loại tập trung vào việc xây dựng một mô hình dự đoán hoặc phân loại dựa trên các thuộc tính đối tượng và phân lớp những đặc tính đó.

- ***Kỹ thuật phát hiện bất thường (Anomaly or Outlier Detection)***

Là quá trình nhận diện và xác định các điểm dữ liệu không bình thường, các tập dữ liệu không khớp với mẫu dự kiến thông qua việc quan sát trong một tập dữ liệu. Các điểm dữ liệu này thường là những điểm có đặc điểm khác biệt so với hầu hết các điểm trong tập dữ liệu, có thể biểu thị cho các sự kiện, hành vi hoặc điều kiện đặc biệt. Bất thường ở đây có thể đề cập đến độ lệch, sự khác thường, các nhiễu và ngoại lệ.

- ***Kỹ thuật phân tích theo cụm (Clustering Analysis)***

Là một tác vụ gom nhóm một tập các đối tượng theo cách các đối tượng cùng nhóm (gọi là cụm, cluster) sẽ có tính tương quan, tương tự, giống nhau (theo các đặc tính nào đó) hơn so với các đối tượng ngoài nhóm hoặc thuộc các nhóm khác. Điều này giúp trực quan hóa dữ liệu. Điển hình như việc chia phân khúc khách hàng.

- ***Kỹ thuật phân tích hồi quy (Regression Analysis)***

Được sử dụng để nghiên cứu mối quan hệ giữa một biến phụ thuộc (biến mục tiêu) và một hoặc nhiều biến độc lập (biến dự đoán). Mục tiêu chính của phân tích hồi quy là dự đoán hoặc mô tả biến phụ thuộc dựa trên các biến độc lập. Trong phân tích hồi quy, biến phụ thuộc là biến chúng ta muốn dự đoán hoặc hiểu rõ hơn. Các biến độc lập là những yếu tố mà chúng ta cho là có ảnh hưởng đến biến phụ thuộc. Phân tích hồi quy thường được thực hiện bằng cách xác định một mô hình hồi quy để mô tả mối quan hệ giữa các biến.

2.1.3.2. Công cụ khai phá dữ liệu

Với sự tiến bộ trong lĩnh vực khoa học và kỹ thuật, các tập dữ liệu ngày càng trở nên đa dạng và phức tạp. Điều này đòi hỏi sự xuất hiện của những công cụ mạnh mẽ, có đầy đủ tính năng hơn để đáp ứng nhu cầu của các chuyên gia trong việc tổng hợp, phân tích và báo cáo dữ liệu một cách chính xác nhất. Để thực hiện khai phá dữ liệu thì sẽ gồm các bước như: làm sạch dữ liệu, tích hợp dữ liệu, chọn dữ liệu, chuyển đổi dữ liệu, khai thác dữ liệu, đánh giá mẫu và trình bày thông tin.

Một số công cụ khai phá dữ liệu phổ biến hiện nay: Data mining SAS, Data Melt, Rapid Miner, Weka, Rattle, KNime, Apache Mahout. Trong bài tiểu luận này nhóm chúng em sử dụng công cụ Orange để thực hiện.

Tại sao chọn công cụ orange?

Công cụ này cung cấp cho người dùng một giao diện đồ họa dễ sử dụng và một loạt các tính năng xử lý, phân tích, khai thác, dự đoán, trực quan hóa,... từ dữ liệu đơn giản cho đến phức tạp. Orange được sử dụng trong các lĩnh vực như khoa học dữ liệu, khai thác dữ liệu và phân tích dữ liệu cho các nghiên cứu khoa học, công nghiệp và giáo dục.

2.2. Kỹ thuật phân lớp

2.2.1. Khái niệm

Phân lớp dữ liệu là quá trình phân một đối tượng dữ liệu vào một hay nhiều lớp (loại) đã cho trước nhờ một mô hình phân lớp. Mô hình này được xây dựng dựa trên một tập dữ liệu đã được gán nhãn trước đó (thuộc về lớp nào). Mục tiêu của phân lớp dữ liệu là tạo ra một mô hình có khả năng dự đoán lớp của các dữ liệu mới dựa trên các đặc trưng của chúng.

Nhiệm vụ của bài toán phân lớp là phân các đối tượng dữ liệu vào n lớp cho trước. Có các loại phân lớp như sau:

- Phân lớp nhị phân: Khi chỉ có hai lớp được xác định trước. (n=2)
- Phân lớp đa lớp: Khi có hơn hai lớp được xác định trước. (n>2)
- Phân lớp đơn nhãn: Mỗi đối tượng dữ liệu chỉ thuộc vào một lớp duy nhất.
- Phân lớp đa nhãn: Một đối tượng dữ liệu có thể thuộc vào nhiều lớp khác nhau cùng một lúc.

2.2.2. Các phương pháp phân lớp

Mỗi kỹ thuật có những ưu điểm và hạn chế riêng, và việc lựa chọn phương pháp phân lớp phù hợp phụ thuộc vào bản chất của dữ liệu và bài toán cụ thể đang giải quyết. Dưới đây là ba kỹ thuật thường được dùng nhiều nhất:

- *Logistic Regression:* Đây là một mô hình thống kê được sử dụng cho các nhiệm vụ phân loại nhị phân, nơi biến mục tiêu có hai kết quả có thể xảy ra. Phương pháp này ước lượng xác suất của một sự kiện dựa trên các biến đầu vào và được sử dụng rộng rãi trong các lĩnh vực như y tế, tài chính và khoa học xã hội.
- *Naive Bayes Classifier:* Đây là một thuật toán phân loại dựa trên định lý Bayes với giả định về sự độc lập giữa các đặc trưng, thường được sử dụng trong các bài toán có dữ liệu đa chiều.
- *Support Vector Machines - SVM:* Đây là một kỹ thuật phân loại mạnh mẽ, tìm ra siêu phẳng tối ưu để phân chia các lớp dữ liệu, đặc biệt hiệu quả trong không gian đặc trưng có số chiều cao.

2.3. Phương pháp đánh giá mô hình phân lớp

2.3.1. Ma trận nhầm lẫn (Confusion Matrix) và độ chính xác (Accuracy); ROC, AUC, Precision/Recall

Ma trận nhầm lẫn (Confusion Matrix) là một công cụ đánh giá hiệu suất của một mô hình dự đoán, thuật toán phân loại trong học máy và thị giác máy tính bằng cách tính toán số lượng các điểm dữ liệu được phân loại đúng và sai, so sánh dự đoán của mô hình với nhãn thực tế của dữ liệu.

Confusion Matrix thường có dạng một bảng hai chiều, trong đó hàng thể hiện các dự đoán của mô hình, còn cột thể hiện các nhãn thực tế. Các ô trong ma trận biểu thị số lượng mẫu

được phân loại đúng và nhầm lẫn theo từng loại. Nó được xây dựng dựa trên hai loại kết quả: dự đoán đúng (True) và dự đoán sai (False). Ma trận nhầm lẫn bao gồm bốn chỉ số chính: True Positive (TP), False Positive (FP), True Negative (TN) và False Negative (FN).

			Actual Values	
			1 (Positive)	0 (Negative)
			TRUE	FALSE
Predicted Values	1 (+ve)	TRUE	True Positive (TP)	False Positive(FP)
	0(-ve)	FALSE	False Negative (FN)	True Negative (TN)

Hình 2.1: Ma trận nhầm lẫn (Confusion Matrix)

Để đơn giản hóa, ta sẽ sử dụng lại bài toán về chẩn đoán ung thư để giải thích 4 chỉ số này. Trong bài toán chẩn đoán ung thư ta có 2 lớp: lớp bị ung thư được chẩn đoán Positive và lớp không bị ung thư được chẩn đoán là Negative.

- TP (True Positive): Số lượng dự đoán chính xác. Là khi mô hình dự đoán đúng một người bị ung thư.
- TN (True Negative): Số lương dự đoán chính xác một cách gián tiếp. Là khi mô hình dự đoán đúng một người không bị ung thư, tức là việc không chọn trường hợp bị ung thư là chính xác.
- FP (False Positive - Type 1 Error): Số lượng các dự đoán sai lệch. Là khi mô hình dự đoán một người bị ung thư và người đó hoàn toàn khỏe mạnh.
- FN (False Negative - Type 2 Error): Số lượng các dự đoán sai lệch một cách gián tiếp. Là khi mô hình dự đoán một người không bị ung thư nhưng người đó bị ung thư, tức là việc không chọn trường hợp bị ung thư là sai.

Dựa trên ma trận nhầm lẫn có thể tính các chỉ số đánh giá hiệu suất như: độ chính xác (Accuracy), độ nhạy (Recall), độ chính xác dương tính (Precision) và F1-score để đánh giá hiệu suất của mô hình phân loại.

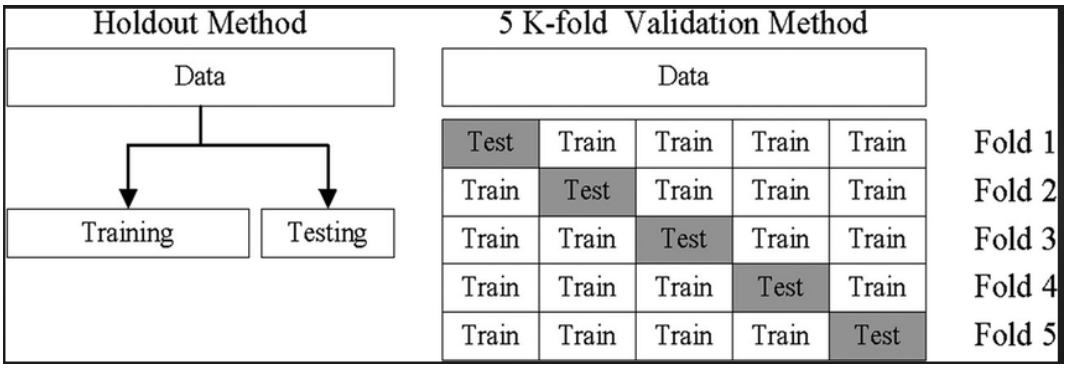
2.3.2. Cross Validation: Holdout và K-fold cross validation

Cross validation - kiểm chứng chéo là việc phân nhóm một mẫu dữ liệu thành các mẫu con để cho việc phân tích ban đầu chỉ thực hiện trên một mẫu con đơn, còn các mẫu con còn lại được giữ "kín" để dùng cho việc xác nhận và kiểm chứng lại lần phân tích đầu tiên đó.

Trong Machine Learning và các phương pháp thống kê, Cross Validation là một kỹ thuật để đánh giá hiệu suất của một mô hình dự đoán trên dữ liệu huấn luyện. Nó giúp đánh giá khả năng tổng quát hóa của mô hình trên dữ liệu mới mà nó chưa từng thấy trước đó.

Có nhiều phương pháp Cross Validation khác nhau, trong đó Holdout và K-fold Cross Validation là hai kỹ thuật phổ biến.

- Trong Holdout, dữ liệu được chia thành hai phần: một phần để huấn luyện mô hình và một phần để kiểm tra mô hình. Thông thường, tỷ lệ phân chia là khoảng 70-80% dữ liệu được sử dụng để huấn luyện và 30-20% dữ liệu được sử dụng để kiểm tra. Một số biến thể của Holdout có thể chia dữ liệu thành ba phần: huấn luyện, kiểm tra và xác thực (validation).
- Trong K-fold Cross Validation, dữ liệu được chia thành K phần bằng nhau (gọi là folds). Quá trình Cross Validation được thực hiện K lần, trong mỗi lần, một fold được chọn làm tập kiểm tra và các folds còn lại được sử dụng để huấn luyện mô hình. Kết quả từ K lần thử nghiệm này được kết hợp lại để tính toán các thước đo đánh giá hiệu suất của mô hình.



Hình 2.2: Bảng so sánh hai phương pháp Holdout và K-fold Cross Validation

Ưu điểm của Holdout là nhanh chóng và dễ triển khai, nhưng nhược điểm là hiệu quả thấp khi dữ liệu huấn luyện ít. K-fold Cross Validation giúp tận dụng tốt hơn dữ liệu huấn luyện và đánh giá mô hình một cách chính xác hơn, nhưng nó đòi hỏi chi phí tính toán cao hơn do cần huấn luyện mô hình K lần.

2.3.3. Độ chính xác (Accuaracy)

Độ chính xác (Accuracy) là một phương pháp đánh giá hiệu suất mô hình phân loại bằng cách tính toán tỷ lệ phân loại chính xác trên tất cả các điểm dữ liệu trong tập kiểm tra. Nó phản ánh mức độ hoạt động của mô hình trên dữ liệu mới và chưa được nhìn thấy. Độ chính xác được tính bằng tổng số điểm dữ liệu được phân loại chính xác chia cho tổng số điểm dữ liệu.Tuy nhiên, một mô hình có độ chính xác cao chưa hẳn đã tốt. Accuracy lộ rõ hạn chế khi được sử dụng trên bộ dữ liệu không cân bằng (imbalanced dataset)

$$\text{Accuracy} = \frac{correct\ predictions}{all\ predictions}$$

2.3.4. Precision, Recall, F1-score

Precision, Recall và F1-score là các thước đo hiệu quả để đánh giá mô hình của các bài toán phân lớp.

- **Precision:** bằng cách tính toán tỷ lệ các điểm dữ liệu được phân loại chính xác trong lớp Positive. Precision là tỷ lệ giữa số điểm dữ liệu Positive được phân loại chính xác trên tổng số điểm dữ liệu được phân loại thành lớp Positive. Precision càng cao đồng nghĩa với việc độ chính xác của các điểm tìm được là cao.

$$\text{Precision} = \frac{TP}{TP+FP}$$

- **Recall:** là tỷ lệ giữa số điểm dữ liệu Positive được phân loại chính xác trên tổng số điểm dữ liệu thực sự là Positive. Recall càng cao đồng nghĩa với việc True Positive Rate cao, tức là tỉ lệ bỏ sót các điểm thực sự là positive là thấp.

$$\text{Recall} = \frac{TP}{TP+FN}$$

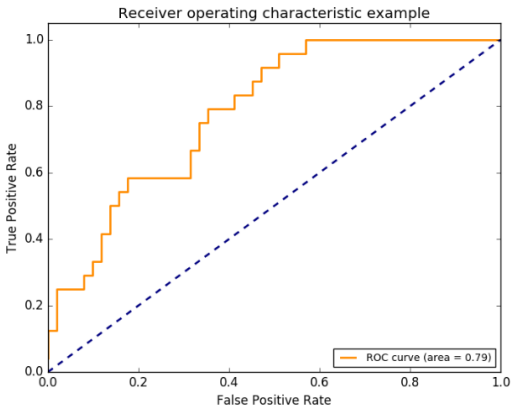
- **F1-score:** là một độ đo để đánh giá hiệu suất của một mô hình phân loại dựa trên sự kết hợp giữa độ chính xác (precision) và độ bao phủ (recall). F1-score là trung bình điều hòa giữa precision và recall, giá trị của nó nằm trong khoảng từ 0 đến 1.

$$\text{F1-score} = \frac{2.precision.recall}{precision+\ recall}$$

2.3.5. ROC và AUC

ROC (Receiver Operating Characteristic): là một đồ thị được sử dụng khá phổ biến trong đánh giá các mô hình phân loại nhị phân. Đường cong này được tạo ra bằng cách biểu diễn tỷ lệ dự báo True Positive Rate (TPR) dựa trên tỷ lệ dự báo False Positive Rate (FPR) tại các ngưỡng khác nhau.

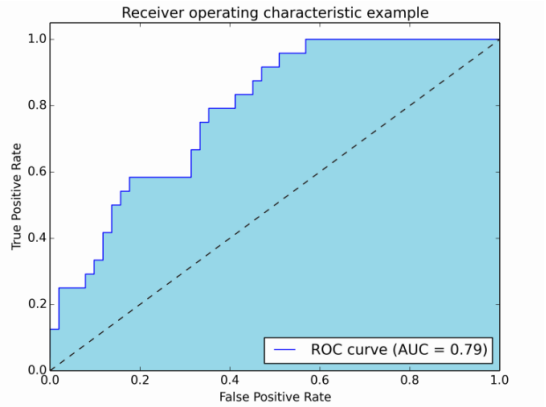
- Một mô hình hiệu quả khi có FPR thấp và TPR cao, hay ROC càng tiệm cận với điểm (0;1) - góc trên bên trái trong đồ thị thì mô hình càng hiệu quả.



Hình 2.3: Minh họa đường cong ROC

AUC (Area Under the Curve): là phần diện tích nằm dưới đường cong ROC.

- Giá trị này là một số dương nhỏ hơn hoặc bằng 1. Và khi giá trị này càng lớn thì mô hình sẽ càng tốt.



Hình 2.4: Minh họa AUC

CHƯƠNG III: MÔ HÌNH VÀ THỰC HIỆN NGHIÊN CỨU

3.1. Mô tả bộ dữ liệu

Với mong muốn đạt được sự chính xác và tin cậy cao trong việc dựa đoán quyết định hủy đặt phòng hoặc không của khách hàng, nhóm đã thu thập dữ liệu từ Kaggle – một trang web cung cấp số liệu đáng tin cậy. Sau khi sàng lọc và chọn lựa, nhóm đã chọn bộ dữ liệu có tên là " Hotel Reservations Dataset" (Bộ dữ liệu đặt phòng khách sạn).

Nguồn dữ liệu được lấy từ Kaggle: **Hotel Reservations Dataset**

Bộ dữ liệu bao gồm một số thông tin cơ bản như sau:

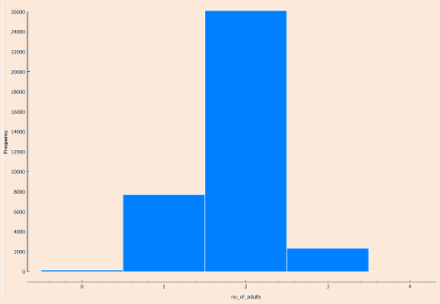
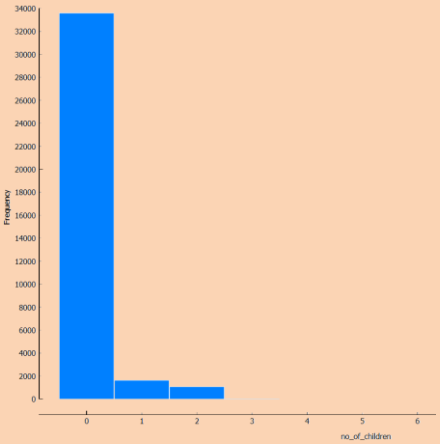
+ Đặc điểm của bộ dữ liệu: Đa biến

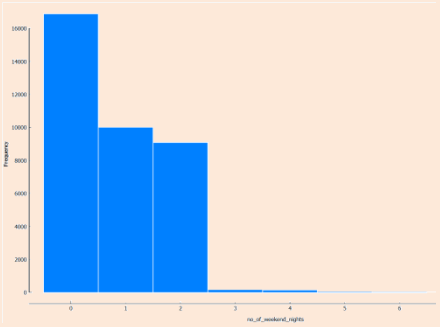
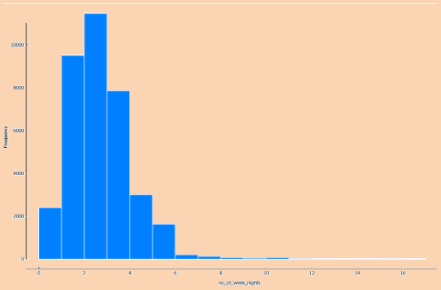
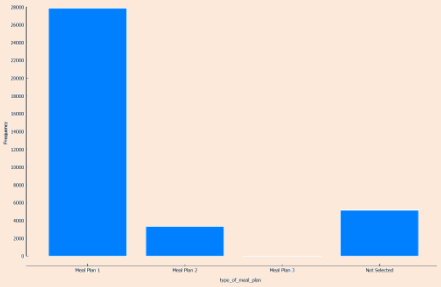
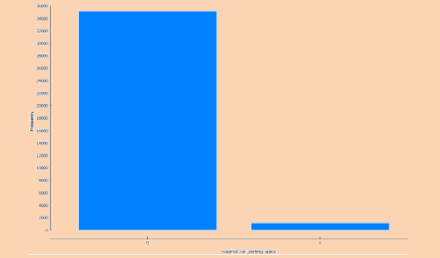
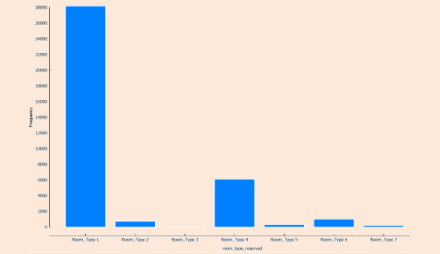
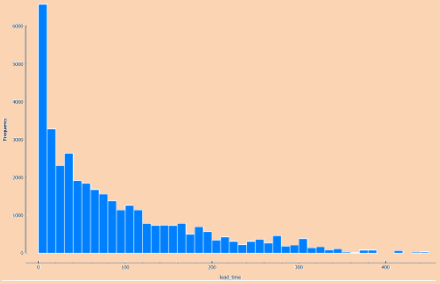
+ Số lượng thuộc tính: 18

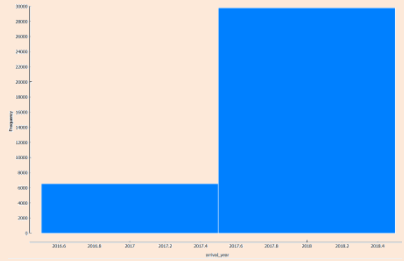
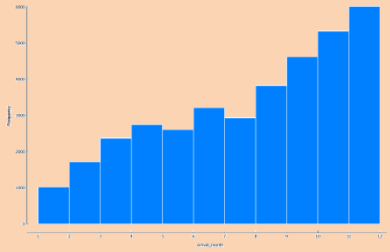
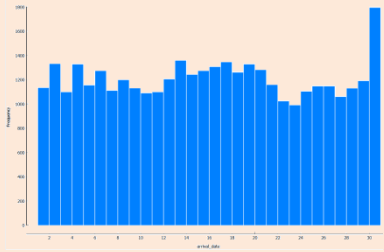
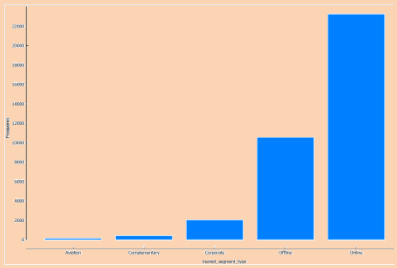
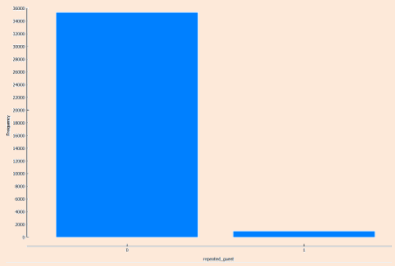
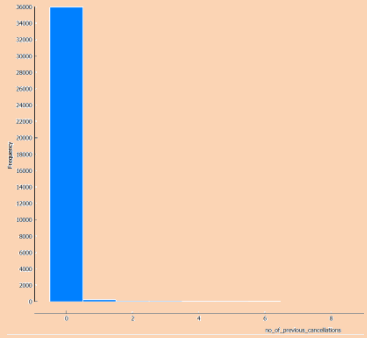
+ Số lượng mục tiêu (target): 1 target với 2 giá trị

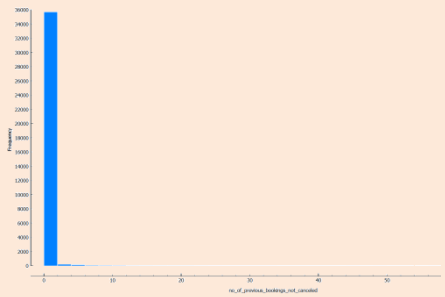
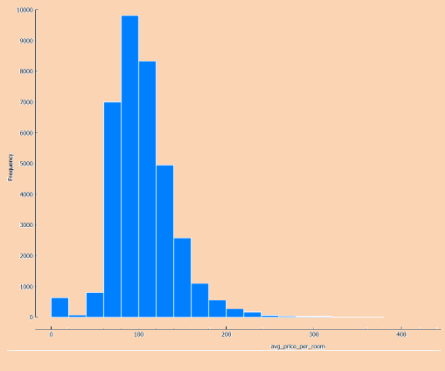
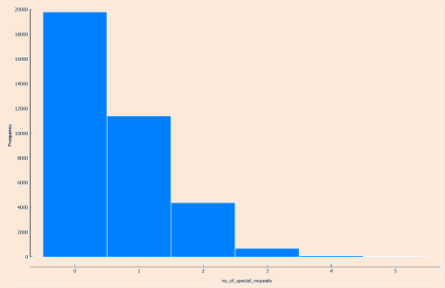
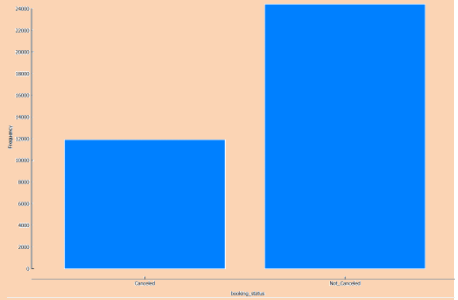
+ Số lượng quan sát: 36275

Thông tin mô tả các thuộc tính:

STT	Thuộc tính	Ý nghĩa	Ghi chú	Phân phối
1	Bookling_ID	Mã đặt phòng	Meta	
2	No_of_adults	Số lượng người lớn	Feature	
3	No_of_children	Số lượng trẻ em	Feature	

4	No_of_ weekend_nights	Số đêm cuối tuần (thứ 7 và chủ nhật)	Feature	
5	No_of_week_ nights	Số đêm trong tuần (từ thứ 2 đến thứ 6)	Feature	
6	Type_of_meal_ plan	Loại gói bửa ăn do khách hàng đặt	Feature	
7	Required_car_ parking_space	Yêu cầu chổ đỗ xe	Feature	
8	Room_type _reserved	Loại phòng do khách hàng đặt	Feature	
9	Lead_time	Số ngày từ ngày đặt phòng đến ngày nhận phòng	Feature	

10	Arrival_year	Năm nhận phòng	Feature	
11	Arrival_month	Tháng nhận phòng	Feature	
12	Arrival_date	Ngày nhận phòng	Feature	
13	Market_segment _type	Chỉ định phân khúc thị trường	Feature	
14	Repeated_guest	Xác định khác hàng thân quen	Feature	
15	No_of_previous _cancellations	Lịch sử hủy phòng của khách hàng	Feature	

16	No_of_previous_ bookings_ not_canceled	Lịch sử khách hàng không hủy phòng	Feature	
17	Avg_prive_ per_room	Giá trung bình mỗi ngày đặt phòng	Feature	
18	No_of_special _requests	Tổng số yêu cầu đặt biệt của khách hàng	Feature	
19	Booking_status	Phòng có bị hủy hay không	Target	

Bảng 3.1: Mô tả thuộc tính của bộ dữ liệu

3.2. Tiền xử lý dữ liệu

Bộ dữ liệu gốc bao gồm có 18 thuộc tính quan sát (Feature), nhóm chọn thuộc tính "Booking_status" làm target để dự báo khả năng hủy đặt phòng của khách hàng.

Sử dụng công cụ Feature Statistics để mô tả tất cả các giá trị của dữ liệu:

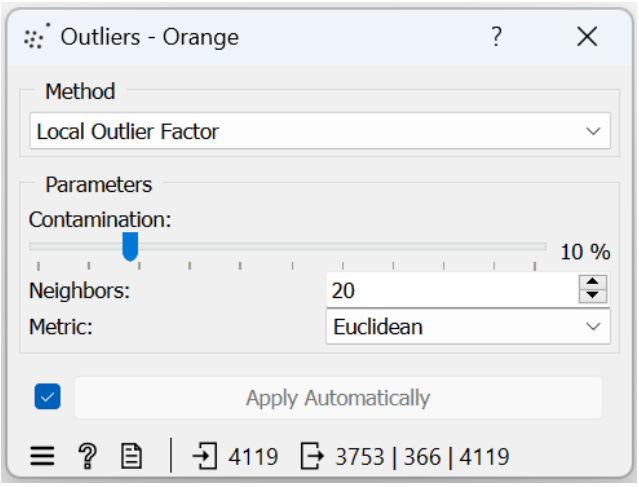
Quan sát thấy được các thuộc tính (Feature) đều có cột Missing value (dữ liệu thiếu) là 0%. Quan sát sự phân phối cho thấy có nhiều điểm dữ liệu cách xa so với trung bình mẫu nên ta dùng Outliers để phân loại các điểm ngoại lai để làm tăng độ chính xác cho mô hình.

Name	Mean	Mode	Median	Dispersion	Min.	Max.	Missing
no_of_adults	1.84	2	2	0.28	0	4	0 (0 %)
no_of_children	0.11	0	0	3.82	0	10	0 (0 %)
no_of_weekend...	0.81	0	1	1.07	0	7	0 (0 %)
no_of_week_nig...	2.20	2	2	0.64	0	17	0 (0 %)
lead_time	85.23	0	57	1.01	0	443	0 (0 %)
arrival_year	2017.82	2018	2018	0	2017	2018	0 (0 %)
arrival_month	7.42	10	8	0.41	1	12	0 (0 %)
arrival_date	15.60	13	16	0.56	1	31	0 (0 %)
no_of_previous_...	0.02	0	0	15.77	0	13	0 (0 %)
no_of_previous_...	0.15	0	0	11.43	0	58	0 (0 %)
avg_price_per_r...	103.4235	65.00	99.45	0.3393	0.00	540.00	0 (0 %)
no_of_special_r...	0.62	0	0	1.27	0	5	0 (0 %)
type_of_meal_pl...		Meal Plan 1		0.699			0 (0 %)
required_car_pa...		0		0.138			0 (0 %)
room_type_rese...		Room_Type 1		0.729			0 (0 %)
market_segmen...		Online		0.874			0 (0 %)
repeated_guest		0		0.119			0 (0 %)
booking_status		Not_Canceled		0.632			0 (0 %)

Color: booking_status Send Automatically

36.3k | - | 18

Hình 3.1: Dùng Feature Statistics quan sát tổng quan bộ dữ liệu



Hình 3.2: Thao tác phân loại dữ liệu ngoại lai

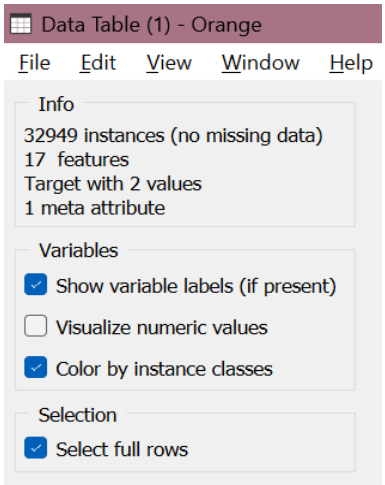
Tại Method chọn Local Outlier Factor – đây là một kỹ thuật khai thác ý tưởng về việc sử dụng các mẫu lân cận để phát hiện ngoại lệ. Mỗi mẫu sẽ được gán cho một giá trị Score thể hiện mức độ cô lập hoặc khả năng nó có thể là Outlier dựa trên quy mô của vùng lân cận của nó.

Tại Parameters lựa chọn các tham số bao gồm:

+ Contamination: Tỷ lệ phần trăm các ngoại lai trong bộ dữ liệu

+ Neighbors: Số lượng biến lân cận

+ Metric: Đơn vị đo khoảng cách

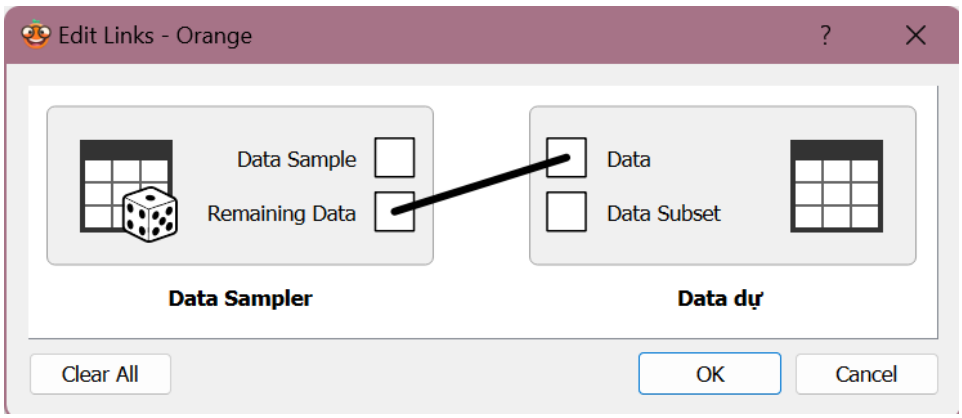


Hình 3.3: Sau khi phân loại dữ liệu ngoại lai

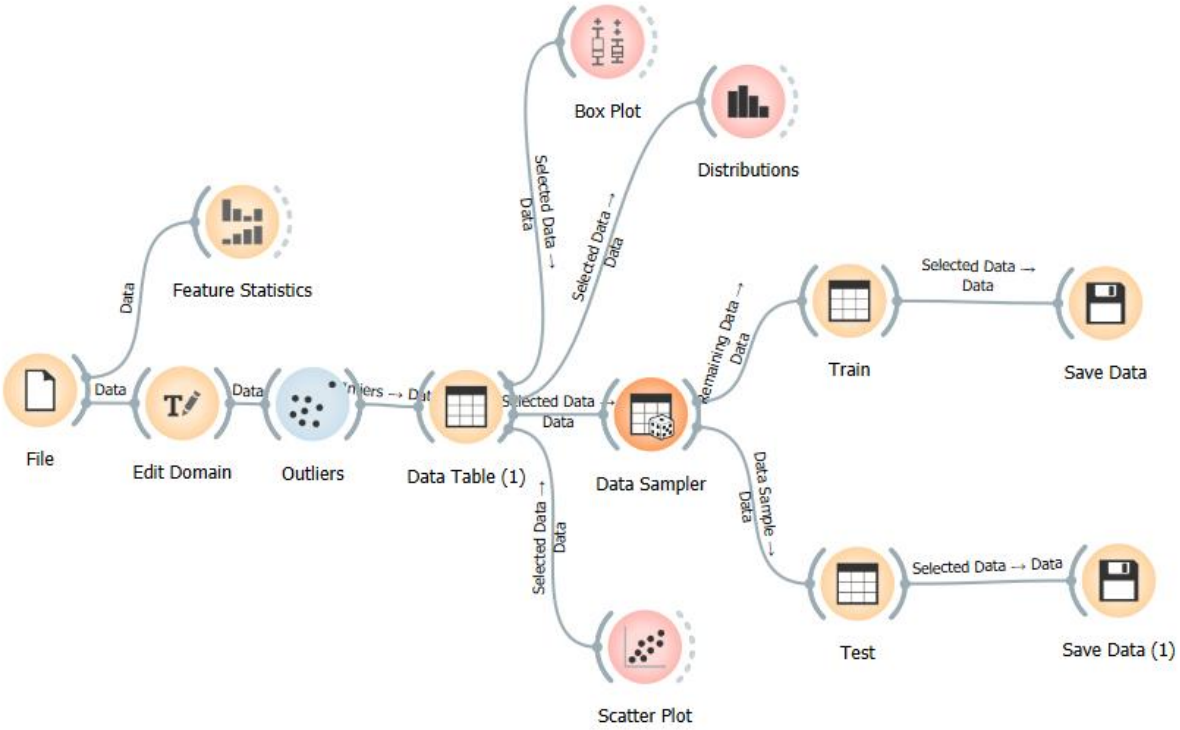
Sau khi lọc các giá trị ngoại lai bộ dữ liệu còn 32949 quan sát (Instances), 17 biến đặc trưng (features) với 0% dữ liệu bị thiếu (missing value), 1 biến mục tiêu (target) có 2 giá trị và 1 biến biến đổi (meta).

3.3. Phân tách dữ liệu

Trong bài nghiên cứu này, hai bộ dữ liệu mà nhóm sử dụng để phân tích đều được tách ra thành 2 file dữ liệu riêng biệt: 70% của mỗi bộ dữ liệu được sử dụng để làm dữ liệu mẫu cho mô hình phân lớp dữ liệu, 30% dữ liệu còn lại của mỗi bộ được sử dụng để dự báo.



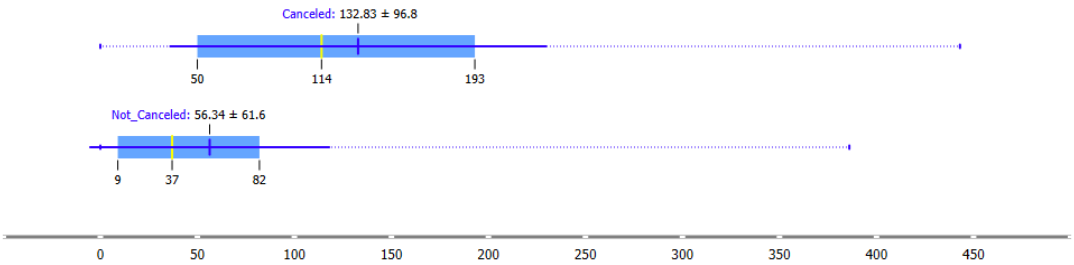
Hình 3.4: Mô tả phân tách bộ dữ liệu



Hình 3.5: Các bước tiền xử lý và phân tách dữ liệu

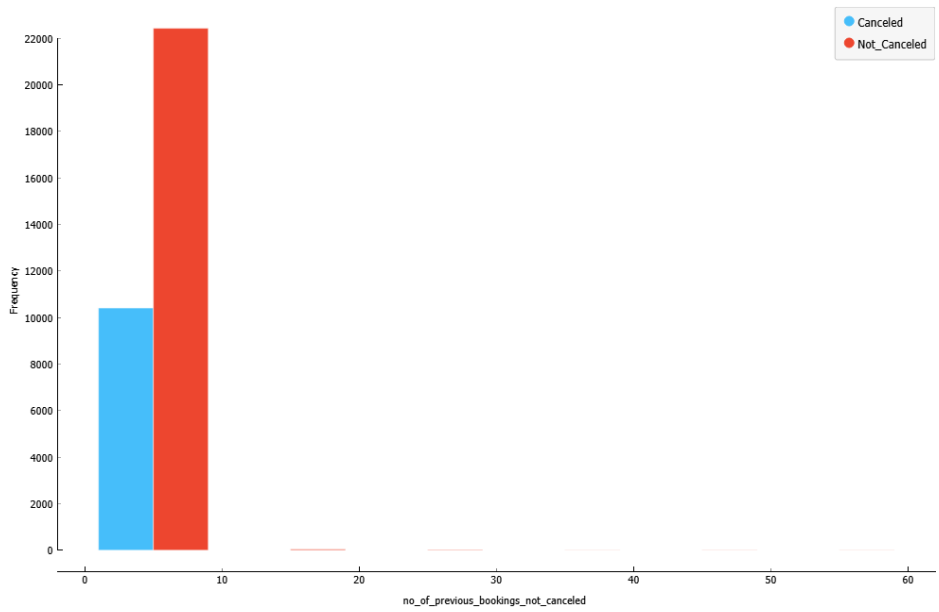
3.4. Đề xuất mẫu

Sau khi quan sát phân phối các biến, nhóm nhận thấy biến **"Lead_time"** và **"No_of_previous_bookings _not_ canceled"** có ảnh hưởng nhiều tới tình trạng đặt phòng khách sạn.



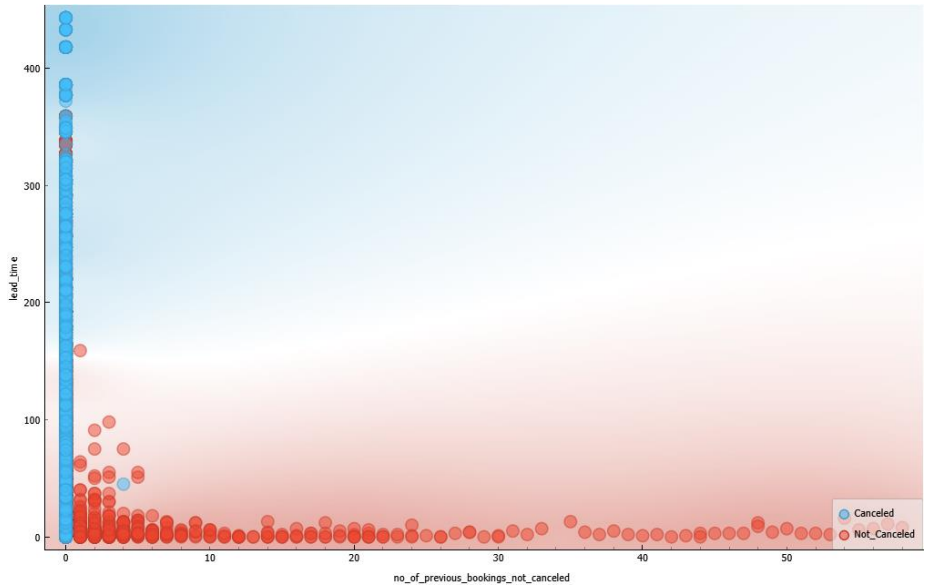
Hình 3.6: Quan sát thuộc tính Lead_time qua Boxplot

Dựa vào Boxplot widget, ta thấy mối quan hệ giữa biến **"Booking_Status"** và **"Lead_time"**: Đối với các khách hàng hủy phòng (*Canceled*) *có khoảng biến thiên lớn hơn* từ 50 đến 193 với trung bình khoảng 132, trung vị là 114; trong khi các khách hàng không hủy phòng (*Not Canceled*) *có khoảng biến thiên nhỏ hơn* từ 9 đến 82 với trung bình khoảng 56 và trung vị là 37.



Hình 3.7: Quan sát thuộc tính No_of_previous_bookings _not_ canceled qua Distrubution

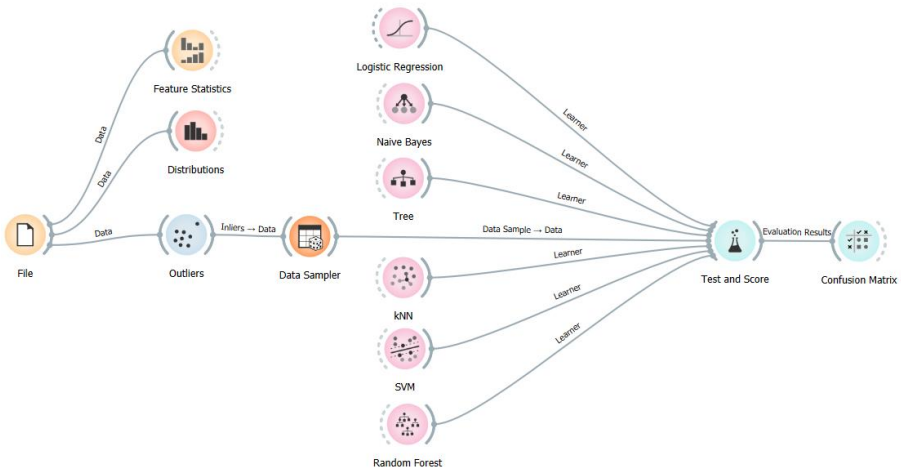
Mối quan hệ giữa biến **"Booking_Status"** và **"No_of_previous_bookings _not_ canceled"**: Các khách hàng không hủy phòng (*Not Canceled*) đều có lịch sử *số lần không hủy phòng (No_of_previous_bookings_not_ canceled) cao (>10 lần)* trong khi các khách hàng hủy phòng (*Canceled*) lại có lịch sử *số lần không hủy phòng thấp hơn (<10 lần).*



Hình 3.8: Quan sát Lead_time và No_of_previous_bookings _not_ canceled qua Scatter plot

Dựa vào **Scatter plot** xem xét mối quan hệ giữa **"Lead_time"** và **"No_of_ previous_bookings_not_ canceled"** ảnh hưởng đến target **"Booking_Status"**: Những khách hàng hủy phòng thì thường *có số ngày từ ngày đặt phòng đến ngày nhận phòng (Lead_time) dài hơn* so với các khách hàng không hủy phòng và *có số lần không hủy phòng trong quá khứ thấp hơn* ("No_of_ previous_bookings_not_ canceled").

3.5. Trích xuất mẫu

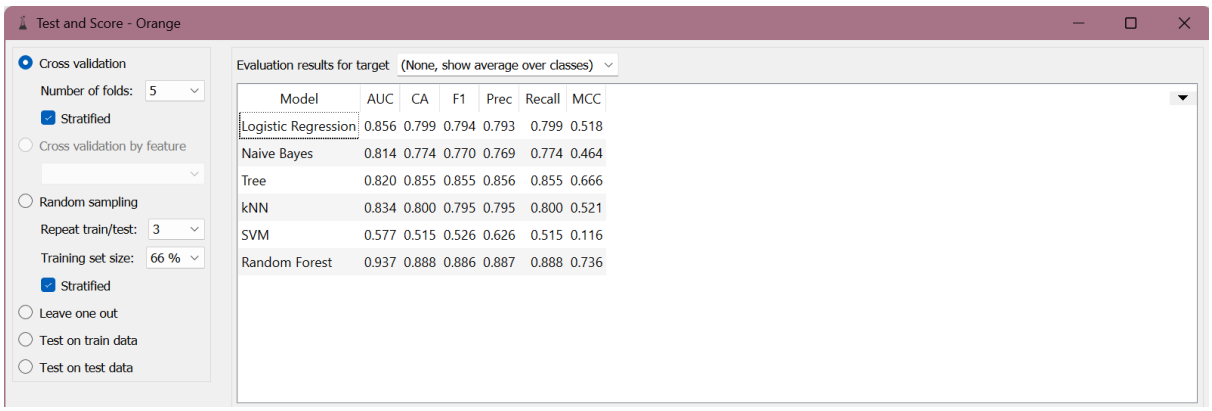


Hình 3.9: Chạy dữ liệu lần đầu

Tiến hành chạy dữ liệu, xem xét các kỹ thuật được chọn như trên có phù hợp với bài toán phân loại biến mục tiêu rời rạc của nhóm hay không.

Chọn chạy dữ liệu với các thông số mặc định của 6 kỹ thuật như trên bao gồm:

- Logistic Regression
- Naive Bayes
- Decision Tree
- kNN
- SVM
- Random Forest



Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression	0.856	0.799	0.794	0.793	0.799	0.518
Naive Bayes	0.814	0.774	0.770	0.769	0.774	0.464
Tree	0.820	0.855	0.855	0.856	0.855	0.666
kNN	0.834	0.800	0.795	0.795	0.800	0.521
SVM	0.577	0.515	0.526	0.626	0.515	0.116
Random Forest	0.937	0.888	0.886	0.887	0.888	0.736

Hình 3.10: Kết quả các chỉ số của widget Test and Score

➔ Các kỹ trên đều chạy được trên bộ dữ liệu và phù hợp để sử dụng cho bài toán phân lớp nhị phân của nhóm, sau đây nhóm sẽ phân công tìm hiểu các kỹ thuật trên cho từng thành viên.

3.5.1. Kỹ thuật Decision Tree – Phan Thúy Ngân

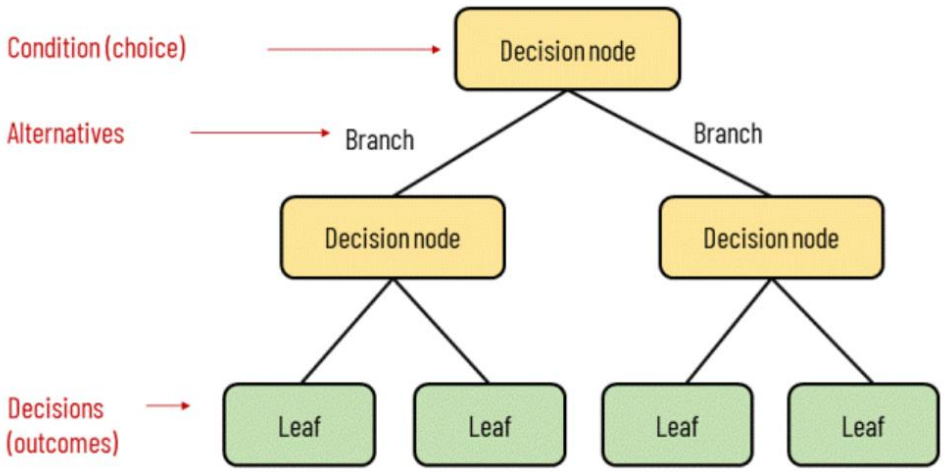
3.5.1.1. Lý thuyết

a) Định nghĩa Decision Tree

Decision Tree hay cây quyết định là một mô hình phân cấp được sử dụng trong hỗ trợ quyết định mô tả các quyết định và kết quả tiềm năng của chúng, kết hợp các sự kiện ngẫu nhiên, chi phí nguồn lực và tiện ích. Mô hình thuật toán này sử dụng các câu lệnh điều khiển có điều kiện và là mô hình học tập có giám sát, không tham số, hữu ích cho cả nhiệm vụ phân loại và hồi quy. Cấu trúc cây bao gồm nút gốc, các nhánh, nút bên trong và nút lá, tạo thành một cấu trúc phân cấp giống như cây. Cây quyết định được sử dụng cho các nhiệm vụ phân loại và hồi quy, cung cấp các mô hình dễ hiểu.

Một số khái niệm liên quan tới cây quyết định:

- Node gốc (root node): Là node ở vị trí đầu tiên của cây quyết định. Mọi phương án đều bắt nguồn từ node này.
- Node cha (parent node): Là node mà có thể rẽ nhánh xuống những node khác bên dưới. Node bên dưới được gọi là node con.
- Node con (child node): Là những node tồn tại node cha.
- Node lá (leaf node): Là node cuối cùng của một quyết định. Tại đây chúng ta thu được kết quả dự báo. Node lá ở vị trí cuối cùng nên sẽ không có node con.
- Node quyết định (non-leaf node): Những node khác node lá.



Hình 3.11: Minh họa Decision Tree

Ý nghĩa: Phương pháp cây quyết định là một phương pháp quan trọng trong khoa học dữ liệu để phân loại hoặc dự đoán kết quả dựa trên các đặc trưng của dữ liệu đầu vào. Phương pháp này hoạt động bằng cách xây dựng một cây quyết định từ tập dữ liệu huấn luyện.

b) Thuật toán ID3

Trong ID3, tổng có trọng số của *entropy* tại các *leaf-node* sau khi xây dựng decision tree được coi là hàm mất mát của decision tree đó. Các trọng số ở đây tỉ lệ với số điểm dữ liệu

được phân vào mỗi node. Công việc của ID3 là tìm các cách phân chia hợp lý (thứ tự chọn thuộc tính hợp lý) sao cho hàm mất mát cuối cùng đạt giá trị càng nhỏ càng tốt. Như đã đề cập, việc này đạt được bằng cách chọn ra thuộc tính sao cho nếu dùng thuộc tính đó để phân chia, entropy tại mỗi bước giảm đi một lượng lớn nhất. Bài toán xây dựng một decision tree bằng ID3 có thể chia thành các bài toán nhỏ, trong mỗi bài toán, ta chỉ cần chọn ra thuộc tính giúp cho việc phân chia đạt kết quả tốt nhất. Mỗi bài toán nhỏ này tương ứng với việc phân chia dữ liệu trong một n*on-leaf node*. Chúng ta sẽ xây dựng phương pháp tính toán dựa trên mỗi node này.

Xét một bài toán với C class khác nhau. Giả sử ta đang làm việc với một non-leaf node với các điểm dữ liệu tạo thành một tập S với số phần tử là $|S| = N$. Giả sử thêm rằng trong số N điểm dữ liệu này, N_c, $c = 1,2,\ldots,C$ điểm thuộc vào class c. Xác suất để mỗi điểm dữ liệu rơi vào một class c được xấp xỉ bằng $\frac{N_c}{N}$ (maximum likelihood estimation). Như vậy, entropy tại node này được tính bởi:

$$H(S) = -\sum_{c=1}^{C} \frac{N_c}{N} log(\frac{N_c}{N}) \qquad (2)$$

Tiếp theo, giả sử thuộc tính được chọn là x. Dựa trên x, các điểm dữ liệu trong S được phân ra thành K child note $S_1, S_2, \ldots, S_K$ với số điểm trong mỗi child note lần lượt là $m_1, m_2, \ldots, m_K$. Ta định nghĩa:

$$H(x,S) = \sum_{k=1}^{K} \frac{m_K}{N} H(S_K) \qquad (3)$$

Là tổng có trọng số entropy của mỗi child node được tính tương tự như (2). Việc lấy trọng số này là quan trọng vì các node thường có số lượng điểm khác nhau.

Tiếp theo, ta định nghĩa information gain dựa trên thuộc tính x:

$$G(x,S) = H(S) - H(x,S)$$

Trong ID3, tại mỗi node, thuộc tính được chọn được xác định dựa trên:

$$x^* = arg\ max_x\ G(x,S) = arg\ min_x\ H(x,S)$$

Tức thuộc tính khiến cho *information gain* đạt giá trị lớn nhất.

c) Xây dựng Decision Tree

Bắt đầu từ gốc: Thuật toán bắt đầu ở trên cùng, được gọi là "nút gốc", đại diện cho toàn bộ tập dữ liệu.

Đặt câu hỏi tốt nhất: Nó tìm kiếm tính năng hoặc câu hỏi quan trọng nhất để chia dữ liệu thành các nhóm riêng biệt nhất. Điều này giống như đặt một câu hỏi tại một ngã ba trên cây.

Phân nhánh: Dựa trên câu trả lời cho câu hỏi đó, nó chia dữ liệu thành các tập hợp con nhỏ hơn, tạo các nhánh mới. Mỗi nhánh đại diện cho một tuyến đường có thể đi qua cây.

Lặp lại quy trình: Thuật toán tiếp tục đặt câu hỏi và phân chia dữ liệu ở mỗi nhánh cho đến khi đến "nút lá" cuối cùng, đại diện cho các kết quả hoặc phân loại được dự đoán.

Điều kiện dừng: Tất cả các mẫu rơi vào một nút thuộc về cùng một lớp (nút lá), không còn thuộc tính nào có thể dùng để phân chia mẫu nữa, không còn lại mẫu nào tại nút.

d) Ưu và nhược điểm của Decision Tree

- Ưu điểm

- Mô hình dễ hiểu và dễ giải thích, tạo ra bộ luật với mỗi nhánh lá là một luật của cây.
- Dữ liệu đầu vào có thể là là dữ liệu missing, không cần chuẩn hóa hoặc tạo biến giả.
- Cần ít dữ liệu để huấn luyện.
- Là một mô hình hộp trắng
- Có thể làm việc với cả dữ liệu số (rời rạc và liên tục) và dữ liệu phân loại.
- Có thể xác thực mô hình bằng cách sử dụng các kiểm tra thống kê.
- Có thể xử lý tốt một lượng dữ liệu lớn trong thời gian ngắn.

- Nhược điểm

- Mô hình cây quyết định phụ thuộc rất lớn vào dữ liệu. Thậm chí, với một sự thay đổi nhỏ trong bộ dữ liệu, cấu trúc mô hình cây quyết định có thể thay đổi hoàn toàn.
- Không đảm bảo xây dựng được cây tối ưu.
- Cây quyết định hay gặp vấn đề overfitting (tạo ra những cây quá khớp với dữ liệu huấn luyện hay quá phức tạp).
- Thường ưu tiên thuộc tính có nhiều giá trị (khắc phục bằng cách sử dụng Gain Ratio).

e) Ứng dụng

Decision Tree được ứng dụng vào các lĩnh vực khác nhau như:

- Y tế và dược phẩm: Dự đoán khả năng mắc các bệnh lý dựa trên các yếu tố như tuổi, giới tính, lối sống, và dữ liệu y tế. Hỗ trợ quyết định trong chẩn đoán bệnh và lựa chọn phương pháp điều trị dựa trên triệu chứng và kết quả xét nghiệm. Dự đoán phản ứng của bệnh nhân với các loại thuốc và liệu pháp.
- Tài chính và ngân hàng: Đánh giá rủi ro tín dụng của khách hàng và quyết định việc cấp vay dựa trên lịch sử tín dụng và thông tin tài chính. Dự đoán khả năng nợ xấu và xác định các yếu tố quan trọng ảnh hưởng đến khả năng thanh toán. Phân loại và phát hiện gian lận trong giao dịch tài chính và thẻ tín dụng.
- Bán lẻ và tiếp thị: Phân loại khách hàng theo nhóm đối tượng tiềm năng, khách hàng tiềm năng và khách hàng trung thành dựa trên thông tin hành vi mua hàng và lịch sử mua hàng. Dự đoán xu hướng mua sắm và phản ứng với các chiến lược quảng cáo và giảm giá.
- Quản lý chuỗi cung ứng và Logistics: Dự đoán nhu cầu sản phẩm và dự trữ dựa trên dữ liệu về xu hướng tiêu dùng và thông tin thị trường. Tối ưu hóa quy trình đặt hàng và phân phối hàng hóa dựa trên dữ liệu về yêu cầu và tài nguyên.

- Giáo dục: Dự đoán hiệu suất học tập của sinh viên dựa trên dữ liệu về điểm số, dữ liệu học tập và các yếu tố liên quan.

3.5.1.2. Thực hành

Cho bộ dữ liệu huấn luyện sau:

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	Nắng	Nóng	Cao	Yếu	Không
2	Nắng	Nóng	Cao	Mạnh	Không
3	Âm u	Nóng	Cao	Yếu	Có
4	Mưa	Ôn hòa	Cao	Yếu	Có
5	Mưa	Mát	Bình thường	Yếu	Có
6	Mưa	Mát	Bình thường	Mạnh	Không
7	Âm u	Mát	Bình thường	Mạnh	Có
8	Nắng	Ôn hòa	Cao	Yếu	Không
9	Nắng	Mát	Bình thường	Yếu	Có
10	Mưa	Ôn hòa	Bình thường	Yếu	Có
11	Nắng	Ôn hòa	Bình thường	Mạnh	Có
12	Âm u	Ôn hòa	Cao	Mạnh	Có
13	Âm u	Nóng	Bình thường	Yếu	Có
14	Mưa	Ôn hòa	Cao	Mạnh	Không

Bảng 3.2: Bộ dữ liệu huấn luyện Tree

Và bộ dữ liệu kiểm định sau:

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	Nắng	Mát	Cao	Yếu	?
2	Nắng	Nóng	Bình thường	Yếu	?
3	Mưa	Mát	Cao	Yếu	?

4	Âm u	Ôn hòa	Bình thường	Mạnh	?
5	Âm u	Mát	Cao	Yếu	?

Bảng 3.3: Bộ dữ liệu kiểm định Tree

Đây là một bài toán dự đoán liệu đội bóng có nên chơi bóng không dựa trên các quan sát thời tiết. Ở đây, các quan sát đều ở dạng categorical. Bộ dữ liệu huấn luyện mô tả mối quan hệ giữa thời tiết trong 14 ngày (bốn cột đầu, không tính cột ID) và việc một đội bóng có chơi bóng hay không (cột cuối cùng). Ta phải dự đoán giá trị ở cột cuối cùng ở bộ dữ liệu kiểm định nếu biết giá trị của bốn cột còn lại.

Có bốn thuộc tính thời tiết:

Quang cảnh nhận một trong ba giá trị: ***nắng, âm u, mưa***.

Nhiệt độ nhận một trong ba giá trị: ***nóng, lạnh, ôn hòa***.

Độ ẩm nhận một trong hai giá trị: ***cao, bình thường***.

Gió nhận một trong hai giá trị: ***yếu, mạnh***.

(Tổng cộng có 3×3×2×2 = 36 loại thời tiết khác nhau, trong đó 14 loại được thể hiện trong bảng.)

Tìm thứ tự các thuộc tính bằng thuật toán ID3.

$$H(S) = -\sum_{c=1}^{C} \frac{N_c}{N} log(\frac{N_c}{N})$$

Trong 14 giá trị của bộ dữ liệu huấn luyện, có 5 giá trị bằng *không* và 9 giá trị bằng *có*. Entropy tại root node của bài toán là:

$$H(S) = -\frac{5}{14} log(\frac{5}{14}) - \frac{9}{14} log(\frac{9}{14}) \approx 0.65$$

Tiếp theo, chúng ta tính tổng có trọng số entropy của các *child node* nếu chọn một trong các thuộc tính *quang cảnh, nhiệt độ, độ ẩm, gió* để phân chia dữ liệu *chơi bóng đá*.

Xét thuộc tính *quang cảnh*. Thuộc tính này có thể nhận một trong ba giá trị *nắng, âm u, mưa*. Mỗi một giá trị sẽ tương ứng với một *child node*. Gọi tập hợp các điểm trong mỗi child node này lần lượt là $S_{nắng}$, $S_{âm\ u}$, $S_{mưa}$ với tương ứng $m_{nắng}$, $m_{âm\ u}$,$m_{mưa}$ phần tử. Sắp xếp bộ dữ liệu huấn luyện theo thuộc tính quang cảnh ta được ba bảng dữ liệu nhỏ sau:

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	Nắng	Nóng	Cao	Yếu	Không
2	Nắng	Nóng	Cao	Mạnh	Không
8	Nắng	Ôn hòa	Cao	Yếu	Không
9	Nắng	Mát	Bình thường	Yếu	Có
11	Nắng	Ôn hòa	Bình thường	Mạnh	Có

Bảng 3.4: Quang cảnh: Nắng

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
3	Âm u	Nóng	Cao	Yếu	Có
7	Âm u	Mát	Bình thường	Mạnh	Có
12	Âm u	Ôn hòa	Cao	Mạnh	Có
13	Âm u	Nóng	Bình thường	Yếu	Có

Bảng 3.5: Quang cảnh: Âm u

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
4	Mưa	Ôn hòa	Cao	Yếu	Có
5	Mưa	Mát	Bình thường	Yếu	Có
6	Mưa	Mát	Bình thường	Mạnh	Không
10	Mưa	Ôn hòa	Bình thường	Yếu	Có
14	Mưa	Ôn hòa	Cao	Mạnh	Không

Bảng 3.6: Quang cảnh: Mưa

Quan sát nhanh ta thấy rằng child node ứng với *Quang cảnh = "Âm u"* sẽ có entropy bằng 0 vì tất cả $m_0 = 4$ output đều là *có*. Hai *child node* còn lại với $m_{nắng} = m_{mưa} = 5$ có entropy khá cao vì tần suất output bằng *có* hoặc *không* là xấp xỉ nhau. Tuy nhiên, hai *child node* này có thể được phân chia tiếp dựa trên hai thuộc tính nhận hai giá trị là *độ ẩm* và *gió*.

Ta tính được rằng:

$$H(S_{nắng}) = -\frac{2}{5}log(\frac{2}{5}) - \frac{3}{5}log(\frac{3}{5}) \approx 0.673$$

$H(S_{âm\ u}) = 0$

$H(S_{mưa}) = -\frac{3}{5}log(\frac{3}{5}) - \frac{2}{5}log(\frac{2}{5}) \approx 0.673$

$H(quang\ cảnh,S) = \frac{5}{14}H(S_{nắng}) + \frac{4}{14}H(S_{âm\ u}) + \frac{5}{14}H(S_{mưa}) \approx 0.48$

Xét thuộc tính *nhiệt độ*. Thuộc tính này có thể nhận một trong ba giá trị *nóng, ôn hòa, mát*. Mỗi một giá trị sẽ tương ứng với một *child node*. Gọi tập hợp các điểm trong mỗi child node này lần lượt là $S_{nóng}$, $S_{ôn\ hòa}$, $S_{mát}$ với tương ứng $m_{nóng}$, $m_{ôn\ hòa}$, $m_{mát}$ phần tử. Sắp xếp bộ dữ liệu huấn luyện theo thuộc tính nhiệt độ ta được ba bảng dữ liệu nhỏ sau:

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	Nắng	Nóng	Cao	Yếu	Không
2	Nắng	Nóng	Cao	Mạnh	Không
3	Âm u	Nóng	Cao	Yếu	Có
13	Âm u	Nóng	Bình thường	Yếu	Có

Bảng 3.7: Nhiệt độ: Nóng

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
4	Mưa	Ôn hòa	Cao	Yếu	Có
8	Nắng	Ôn hòa	Cao	Yếu	Không
10	Mưa	Ôn hòa	Bình thường	Yếu	Có
11	Nắng	Ôn hòa	Bình thường	Mạnh	Có
12	Âm u	Ôn hòa	Cao	Mạnh	Có
14	Mưa	Ôn hòa	Cao	Mạnh	Không

Bảng 3.8: Nhiệt độ: Ôn hòa

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
5	Mưa	Mát	Bình thường	Yếu	Có
6	Mưa	Mát	Bình thường	Mạnh	Không
7	Âm u	Mát	Bình thường	Mạnh	Có

9	Nắng	Mát	Bình thường	Yếu	Có

Bảng 3.9: Nhiệt độ: Mát

$H(S_{nóng}) = -\frac{2}{4}log(\frac{2}{4}) - \frac{2}{4}log(\frac{2}{4}) \approx 0.693$

$H(S_{ôn\ hòa}) = -\frac{4}{6}log(\frac{4}{6}) - \frac{2}{6}log(\frac{2}{6}) \approx 0.637$

$H(S_{mát}) = -\frac{3}{4}log(\frac{3}{4}) - \frac{1}{4}log(\frac{1}{4}) \approx 0.562$

$H(nhiệt\ độ,S) = \frac{4}{14}H(S_{nóng}) + \frac{6}{14}H(S_{ôn\ hòa}) + \frac{4}{14}H(S_{mát}) \approx 0.631$

Tính toán tương tự cho 2 thuộc tính độ ẩm và gió, có kết quả như sau;

$H(độ\ ẩm,S) \approx 0.547$

$H(gió,S) \approx 0.618$

Ta có: *H(quang cảnh,S) < H(độ ẩm,S) < H(gió,S) < H(nhiệt độ,S)*

➔ Như vậy, thuộc tính cần chọn ở bước đầu tiên là quang cảnh vì *H(quang cảnh,S)* đạt giá trị nhỏ nhất (information gain là lớn nhất).

Sau bước chia đầu tiên này, ta nhận được ba child node với các phần tử như trong ba bảng phân chia theo *quang cảnh*.

Child node *quang cảnh = âm u* không cần phân chia tiếp vì nó đã tinh khiết.

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
3	Âm u	Nóng	Cao	Yếu	**Có**
7	Âm u	Mát	Bình thường	Mạnh	**Có**
12	Âm u	Ôn hòa	Cao	Mạnh	**Có**
13	Âm u	Nóng	Bình thường	Yếu	**Có**

Bảng 3.10: Child node quang cảnh = âm u

Với child node ứng với *quang cảnh = nắng*, kết quả tính được bằng ID3 sẽ cho chúng ta thuộc tính *độ ẩm* vì tổng trọng số của entropy sau bước này sẽ bằng 0 với output bằng *có* khi và chỉ khi *độ ẩm = bình thường*.

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	Nắng	Nóng	Cao	Yếu	Không
2	Nắng	Nóng	Cao	Mạnh	Không

8	Nắng	Ôn hòa	Cao	Yếu	Không
9	Nắng	Mát	**Bình thường**	Yếu	**Có**
11	Nắng	Ôn hòa	**Bình thường**	Mạnh	**Có**

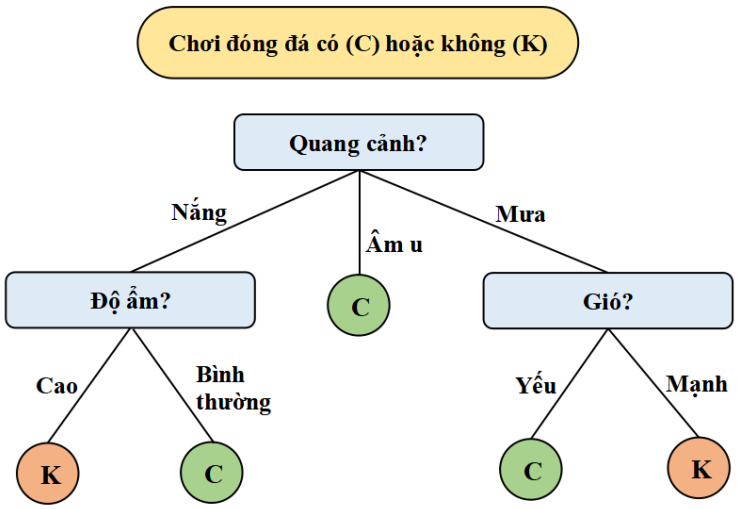
Bảng 3.11: Child node quang cảnh = nắng

Tương tự, child node ứng với *quang cảnh = mưa* sẽ được tiếp tục phân chia bởi thuộc tính gió với output bằng có khi và chỉ khi *gió = yếu*.

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
4	Mưa	Ôn hòa	Cao	**Yếu**	**Có**
5	Mưa	Mát	Bình thường	**Yếu**	**Có**
6	Mưa	Mát	Bình thường	Mạnh	Không
10	Mưa	Ôn hòa	Bình thường	**Yếu**	**Có**
14	Mưa	Ôn hòa	Cao	Mạnh	Không

Bảng 3.12: Child node quang cảnh = mưa

Như vậy, cây quyết định cho bài toán này dựa trên ID3 sẽ có dạng như hình sau:



Hình 3.12: Vẽ cây quyết định

Cách dự đoán dưới đây tương đối đơn giản và khá chính xác từ decision tree, có thể không phải là cách ra quyết định tốt nhất:

Nếu quang cảnh = nắng và độ ẩm = cao thì chơi bóng đá = không.

Nếu quang cảnh = mưa và gió = mạnh thì chơi bóng đá = không.

Nếu quang cảnh = âm u thì chơi bóng đá = có.

Ngoài ra, nếu độ ẩm = bình thường thì chơi bóng đá = có.

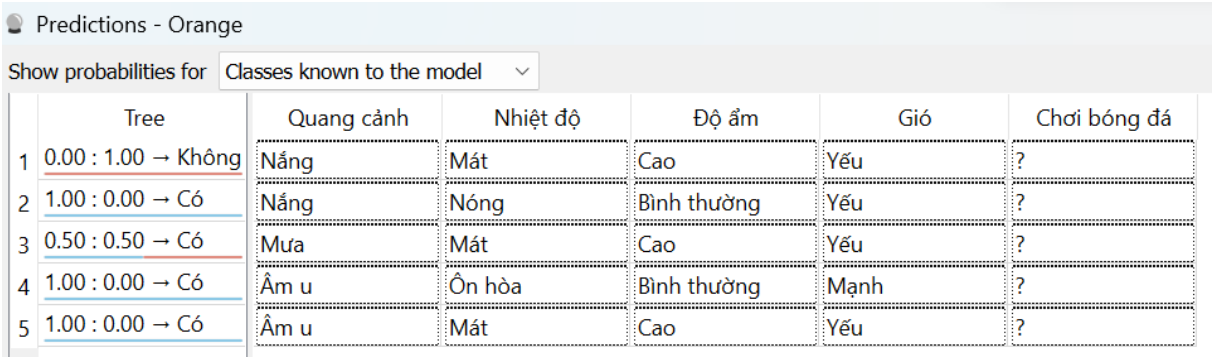
Ngoài ra, chơi bóng đá = có.

Áp dụng cách dự đoán trên cho bộ dữ liệu kiểm định ta được kết quả như sau:

ID	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	Nắng	Mát	Cao	Yếu	**Không**
2	Nắng	Nóng	Bình thường	Yếu	**Có**
3	Mưa	Mát	Cao	Yếu	**Có**
4	Âm u	Ôn hòa	Bình thường	Mạnh	**Có**
5	Âm u	Mát	Cao	Yếu	**Có**

Bảng 3.13: Kết quả tính thủ công Tree

Để tăng độ tin cậy, thực hiện chạy bộ dữ liệu trên Orange để kiểm tra độ chính xác của kết quả trên:

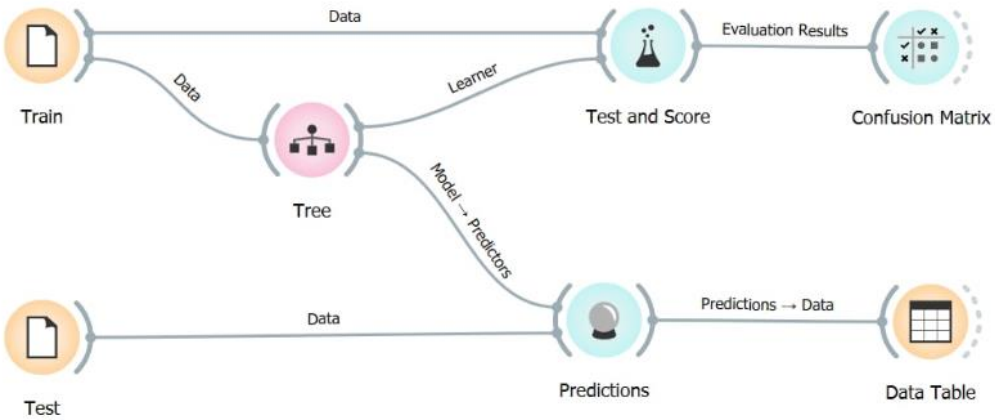


	Tree	Quang cảnh	Nhiệt độ	Độ ẩm	Gió	Chơi bóng đá
1	0.00 : 1.00 → Không	Nắng	Mát	Cao	Yếu	?
2	1.00 : 0.00 → Có	Nắng	Nóng	Bình thường	Yếu	?
3	0.50 : 0.50 → Có	Mưa	Mát	Cao	Yếu	?
4	1.00 : 0.00 → Có	Âm u	Ôn hòa	Bình thường	Mạnh	?
5	1.00 : 0.00 → Có	Âm u	Mát	Cao	Yếu	?

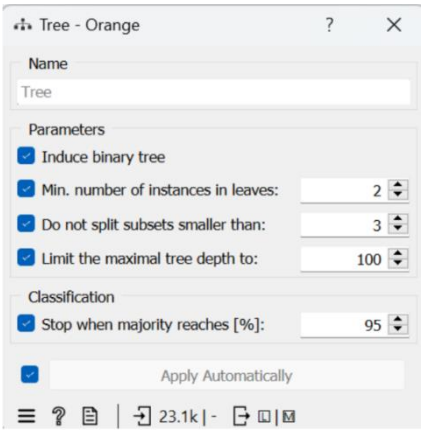
Hình 3.13: Kết quả so sánh với Orange Tree

⇒ Kết quả phân lớp dữ liệu trùng khớp với kết quả đã được thực hiện tính toán.

3.5.1.3. Xây dựng mô hình Decision Tree – Dữ liệu nhóm



Hình 3.14: Xây dựng mô hình theo Decision Tree



Hình 3.15: Tree Widget

Decision Tree là một thuật toán đơn giản phân chia dữ liệu thành các nút bởi các lớp dữ liệu. Nó là tiền thân của Random Forest. Tree trong phần mềm Orange được thiết kế bên trong và có thể xử lý cả bộ dữ liệu rời rạc và liên tục. Nó cũng có thể được sử dụng cho cả nhiệm vụ phân loại và hồi quy

Dưới đây là các tùy chỉnh của Tree Widget trong phần mềm Orange:

- Parameters - Thông số cây
  - Induce binary tree: xây dựng cây nhị phân (chia thành hai nút con)
  - Min. number of instances in leaves (số tối thiểu các ví dụ lá): nếu được chọn, thuật toán sẽ không bao giờ đặt số nút ít hơn số dữ liệu tham khảo
  - Do not split subsets smaller than (Không phân chia các tập hợp nhỏ hơn): cấm thuật toán phân chia các nút có ít hơn số lượng ví dụ đã cho.
  - Limit the maximal tree depth (Giới hạn độ sâu cây tối đa): giới hạn độ sâu của cây phân loại ở số cấp nút được chỉ định.
- Classfication - Phân loại
  - Stop when majority reaches [%]: dừng chia tách các nút sau khi đạt đến ngưỡng đa số được chỉ định

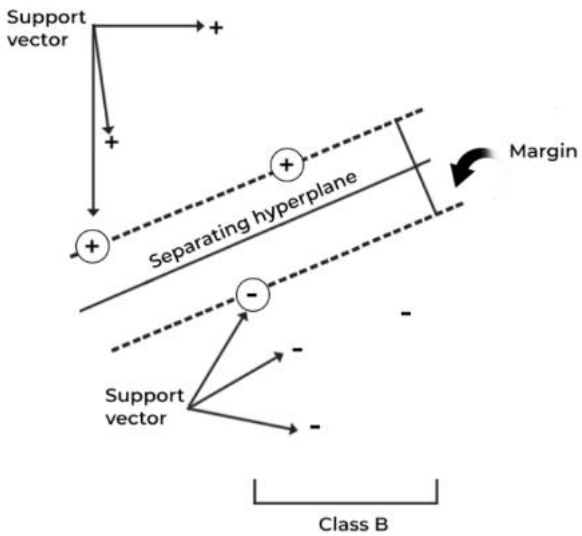
3.5.2. Kỹ thuật Support Vector Machine (SVM) – Ngô Mai Kim Huyền

3.5.2.1. Lý thuyết

a) Định nghĩa

Một máy vector hỗ trợ (SVM) là một thuật toán học máy sử dụng mô hình học có giám sát để giải quyết các vấn đề phân loại, hồi quy, và phát hiện ngoại lệ phức tạp bằng cách thực hiện các biến đổi dữ liệu tối ưu để xác định ranh giới giữa các điểm dữ liệu dựa trên các lớp, nhãn, hoặc đầu ra được xác định trước.

Mục tiêu chính của thuật toán SVM là xác định một siêu phẳng phân biệt rõ ràng các điểm dữ liệu của các lớp khác nhau. Siêu phẳng được định vị sao cho ranh giới lớn nhất tách biệt các lớp đang xem xét.



Hình 3.16: Kỹ thuật Support Vector Machine

Khoảng cách biên đề cập đến độ rộng tối đa của lát cắt chạy song song với siêu phẳng mà không có bất kỳ vector hỗ trợ nào ở bên trong. Các siêu phẳng như vậy dễ xác định hơn cho các vấn đề phân loại tuyến tính có thể phân biệt được; tuy nhiên, đối với các vấn đề hoặc tình huống trong thực tế, thuật toán SVM cố gắng tối đa hóa khoảng cách giữa các vector hỗ trợ, do đó dẫn đến việc phân loại sai cho các phần nhỏ của điểm dữ liệu.

SVM ban đầu được thiết kế cho các vấn đề phân loại nhị phân. Tuy nhiên, với sự gia tăng của các vấn đề phân loại đa lớp yêu cầu tính toán mạnh mẽ, nhiều bộ phân loại nhị phân được xây dựng và kết hợp để tạo ra SVM có thể thực hiện các phân loại đa lớp thông qua các phương pháp nhị phân.

Trong ngữ cảnh toán học, một SVM tham chiếu đến một tập hợp các thuật toán học máy sử dụng phương pháp kernel để biến đổi đặc trưng dữ liệu bằng cách sử dụng các hàm kernel. Các hàm kernel phụ thuộc vào quá trình ánh xạ tập dữ liệu phức tạp lên các chiều cao hơn một cách làm cho việc phân tách điểm dữ liệu trở nên dễ dàng hơn. Hàm này đơn giản hóa ranh giới dữ liệu cho các vấn đề phi tuyến tính bằng cách thêm các chiều cao hơn để ánh xạ các điểm dữ liệu phức tạp.

Khi giới thiệu thêm các chiều, dữ liệu không được biến đổi hoàn toàn vì điều này có thể là một quá trình tốn nhiều tài nguyên tính toán. Kỹ thuật này thường được gọi là kernel trick, trong đó biến đổi dữ liệu vào các chiều cao hơn được thực hiện một cách hiệu quả và giá rẻ.

b) Ưu và nhược điểm của SVM

- **Ưu điểm**

Là một kĩ thuật phân lớp khá phổ biến SVM thể hiện được khá nhiều ưu điểm, trong đó có việc tính toán hiệu quả trên các tập dữ liệu lớn. Có thể kể đến thêm một số ưu điểm của phương pháp SVM như:

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

- Tiết kiệm bộ nhớ: vì chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới, nên chỉ có các điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết định.
- Tính linh hoạt: phân lớp thường là phi tuyến tính, khả năng áp dụng Kernel mới cho phép sự linh động giữa các phương pháp tuyến tính và phi tuyến tính, từ đó khiến cho hiệu suất phân loại lớn hơn.

- **Nhược điểm**
  - Khi số lượng đặc trưng (p) vượt trội so với số lượng mẫu (n), SVM có thể không hoạt động hiệu quả, dẫn đến kết quả không chính xác.
  - Ngoài ra, SVM không cung cấp thông tin xác suất trực tiếp cho việc phân loại, mà chỉ tập trung vào việc tìm ra siêu phẳng tối ưu để phân chia dữ liệu. Tuy nhiên, thông qua việc đánh giá khoảng cách từ các điểm dữ liệu mới đến siêu phẳng (margin), ta có thể có được một ước lượng về độ tin cậy của việc phân loại.

c) Các bước thực hiện

Bước 1: kỹ thuật SVM sẽ lựa chọn không gian đặc trưng. SVM có thể làm việc trong không gian nhiều chiều. Nếu dữ liệu không tách biệt tuyến tính, SVM sẽ sử dụng kỹ thuật như Kernel Trick để chuyển dữ liệu sang không gian đặc trưng cao chiều hơn.

Bước 2: SVM tìm siêu phẳng phân chia tối ưu giữa các lớp dữ liệu. Siêu phẳng này là ranh giới quyết định mà ở đó, khoảng cách (margin) giữa các điểm dữ liệu gần nhất của mỗi lớp là lớn nhất.

Bước 3: SVM sử dụng các vector hỗ trợ (support vectors) - những điểm dữ liệu gần siêu phẳng nhất - để tối ưu hóa margin và đảm bảo rằng siêu phẳng có khoảng cách lớn nhất có thể từ các lớp dữ liệu.

Bước 4: Dùng tập dữ liệu huấn luyện để tìm ra các tham số của siêu phẳng.

d) Ứng dụng

Kỹ thuật phân lớp SVM (Support Vector Machine) được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau do tính linh hoạt và hiệu suất của nó. Dưới đây là một số ứng dụng chính của SVM:

Phân loại ảnh và video: SVM có thể được sử dụng để phân loại ảnh và video vào các nhóm khác nhau, chẳng hạn như nhận diện khuôn mặt, nhận diện cảm xúc, phân loại đối tượng, và phát hiện hoặc phân loại hành động trong video.

Nhận dạng văn bản: SVM có thể được sử dụng để nhận dạng văn bản, bao gồm phân loại văn bản thành các danh mục như spam và không phải spam trong email, phân loại văn bản theo ngôn ngữ, hoặc phân loại văn bản theo chủ đề.

Dự đoán hành vi của người dùng: SVM có thể áp dụng trong các ứng dụng dự đoán hành vi của người dùng trên internet, chẳng hạn như dự đoán hành vi mua sắm trực tuyến, phân loại người dùng thành các nhóm để tùy chỉnh trải nghiệm người dùng.

Tín hiệu và xử lý âm thanh: SVM có thể được áp dụng để phân loại và phát hiện tín hiệu trong xử lý tín hiệu và âm thanh, như phân loại âm thanh thành các loại (ví dụ: nói, âm nhạc, tiếng ồn), hoặc phát hiện sự kiện cụ thể trong tín hiệu.

3.5.2.2. Thực hành

Mô tả thuộc tính:

Buying	Giá mua
Safety	Ước tính độ an toàn của xe
Car	Mức độ đánh giá (không chấp nhận được, chấp nhận được)

Bảng 3.14: Mô tả thuộc tính SVM

Cho bộ dữ liệu huấn luyện sau:

Car	Buying	Safety
Unacc	High	High
Acc	Med	Med
Unacc	High	Med
Acc	Low	Med
Unacc	Med	Low
Acc	Med	Med
Unacc	Vhigh	High
Unacc	High	Med
Unacc	Med	Med
Unacc	High	High
Acc	Low	High
Acc	Med	High
Unacc	Vhigh	High
Unacc	Vhigh	High
Unacc	High	High
Unacc	Vhigh	Med
Unacc	High	Med
Unacc	Low	High
Unacc	Med	High
Acc	Low	Med

Bảng 3.15: Bộ dữ liệu huấn luyện SVM

Bước đầu tiên: Chuyển đổi các thuộc tính không phải dạng số thành dạng số:

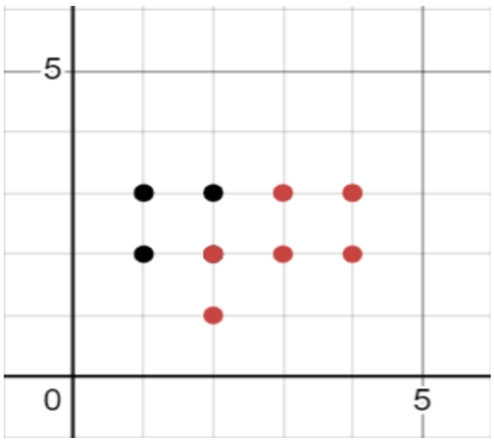
- Buying: “high” → 3, “med” → 2, “low” → 1, “vhigh” → 4
- Safety: “low” → 1, “med” → 2, “high” → 3

Bảng dữ liệu chuyển đổi tương ứng:

Car	Buying	Safety
Unacc	3	3
Acc	2	2
Unacc	3	2
Acc	1	2
Unacc	2	1
Acc	2	2
Unacc	4	3
Unacc	3	2
Unacc	2	2
Unacc	3	3
Acc	1	3
Acc	2	3
Unacc	4	3
Unacc	4	3
Unacc	3	3
Unacc	4	2
Unacc	3	2
Unacc	1	3
Unacc	2	3
Acc	1	2

Bảng 3.16: Chuyển đổi dữ liệu

Bước 2: Chọn kernel: sử dụng một hàm kernel là radial basis function để biến đổi dữ liệu vào không gian cao chiều.

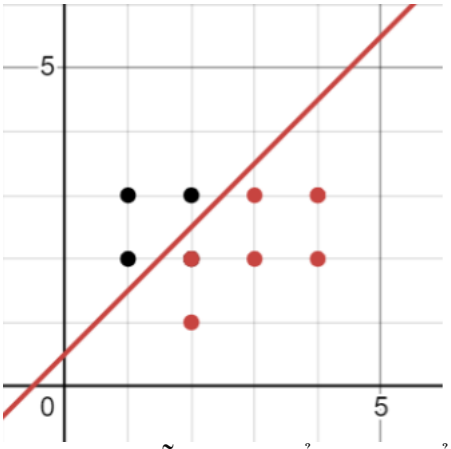


Bước 3: Huấn luyện SVM: SVM tìm ra siêu phẳng tốt nhất để phân chia các điểm dữ liệu thuộc các lớp khác nhau. Siêu phẳng này cách xa nhất các điểm dữ liệu gần nhất của các lớp:

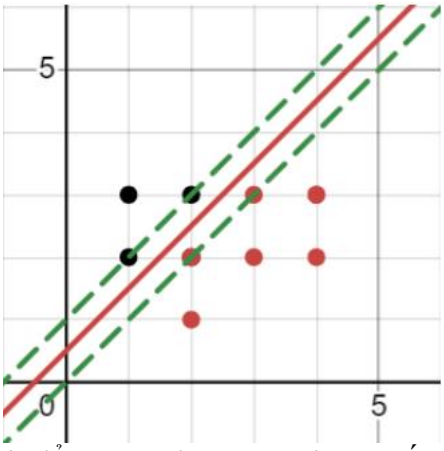
- Đầu tiên, chúng ta cần xác định vector hỗ trợ (support vectors). Đây là các điểm dữ liệu gần nhất với siêu phẳng phân chia.
- Đây là không gian 2 chiều thế nên để tìm siêu phẳng: $w_1x + w_2y + b = 0$ với w_1 và w_2 là các hệ số của siêu phẳng, và b là hệ số chặn. Thỏa điều kiện sau:

$$\min_{w,b} \quad \frac{1}{2}||w||^2$$
$$\text{s.t.} \quad y_i(w^T x_i + b) \geq 1, \forall i$$

- Với x_i là các vectơ hỗ trợ:



- Trong trường hợp này, vectơ hỗ trợ có thể là các điểm (2,2) thuộc lớp A và (1,2) thuộc lớp B.



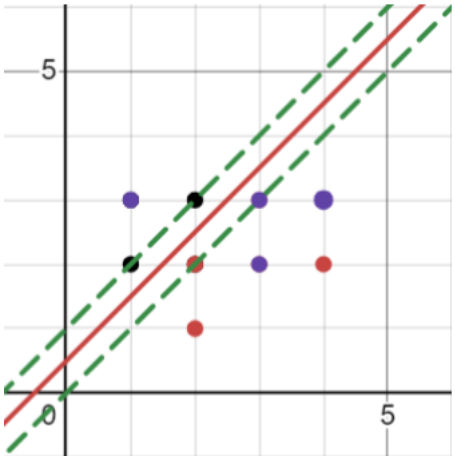
- Việc giải phương trình để tìm ra được các thông số của siêu phẳng khá phức tạp, tuy nhiên khi minh họa bằng hình ảnh có thể dễ dàng xác định được hình dạng của siêu phẳng phân lớp dữ liệu với 2 lề minh họa bằng đường màu xanh lá. (Kết quả siêu phẳng được kiểm định đúng với kết quả thực hiện trên nền tảng [Interactive demo of Support Vector Machines (SVM)])

Bước 4: Sau khi huấn luyện, SVM có thể dự đoán lớp của các điểm dữ liệu mới dựa trên vị trí của chúng so với siêu phẳng đã tìm được.

Thực hiện dự đoán với bộ dữ liệu mất nhãn sau:

Car	Buying	Safety
?	4	3
?	3	2
?	1	3
?	3	3
?	4	3

Bảng 3.17: Bộ dữ liệu kiểm định SVM



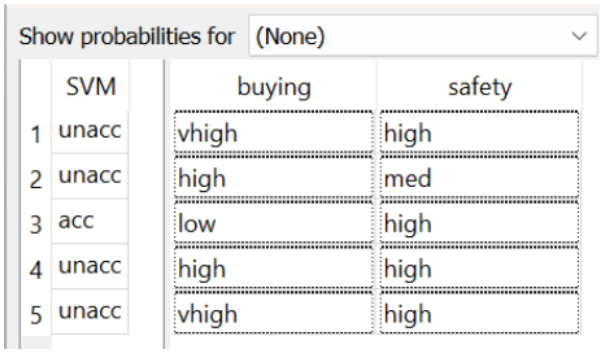
Các điểm dữ liệu cần dự báo nhãn có màu tím.

Kết quả dự báo nhãn mới:

Car	Buying	Safety
Unacc	4	3
Unacc	3	2
Acc	1	3
Unacc	3	3
Unacc	4	3

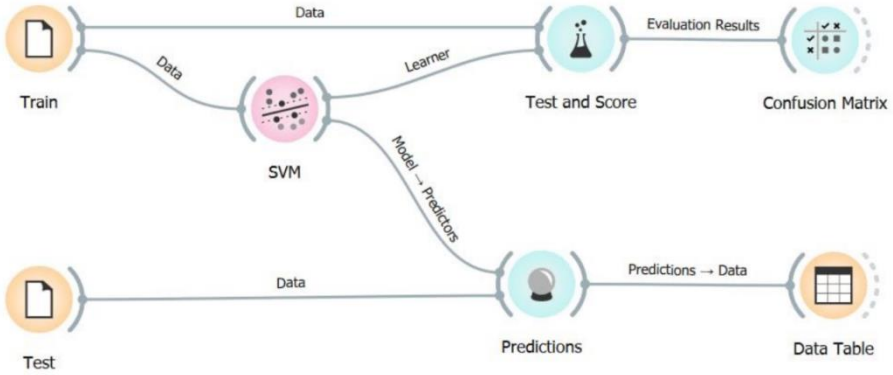
Bảng 3.18: Kết quả phân lớp SVM

Để chắc chắn hơn, kiểm định lại bằng Orange và có được kết quả giống với kết quả trên:

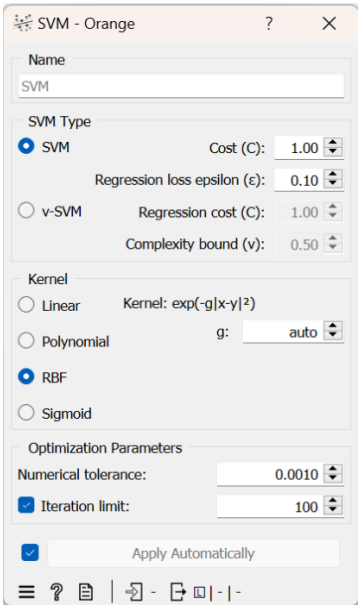


Hình 3.17: Kết quả so sánh với Orange SVM

3.5.2.3. Xây dựng mô hình SVM – Dữ liệu nhóm



Hình 3.18: Xây dựng mô hình theo SVM



Hình 3.19: SVM Widget

Các thông số Widget SVM:

- Loại SVM có cài đặt lỗi kiểm tra. SVM và ν-SVM dựa trên việc giảm thiểu hàm lỗi khác nhau. Ở bên phải là giới hạn lỗi kiểm tra:
  + **Đối với SVM:**
    - Chi phí: thời hạn phạt đối với tổn thất và áp dụng cho các nhiệm vụ phân loại và hồi quy.
    - ε: tham số của mô hình epsilon-SVR, áp dụng cho các tác vụ hồi quy. Xác định khoảng cách từ các giá trị thực mà trong đó không có hình phạt nào liên quan đến các giá trị dự đoán.
  + **Đối với ν-SVM:**
    - Chi phí: thời hạn phạt thua lỗ và chỉ áp dụng cho các nhiệm vụ hồi quy
    - ν: tham số của mô hình ν-SVR, áp dụng cho các nhiệm vụ phân loại và hồi quy. Giới hạn trên của tỷ lệ lỗi huấn luyện và giới hạn dưới của tỷ lệ vectơ hỗ trợ.
- Kernel là một hàm biến đổi không gian thuộc tính thành không gian đặc trưng mới để phù hợp với siêu phẳng có lề tối đa, do đó cho phép thuật toán tạo mô hình với các hạt nhân

Tuyến tính, Đa thức, RBF và Sigmoid. Các hàm chỉ định kernel được trình bày khi chọn chúng và các hằng số liên quan là:

+ g cho hằng số gamma trong hàm kernel (giá trị được đề xuất là 1/k, trong đó k là số thuộc tính, nhưng vì có thể không có tập huấn luyện nào được cung cấp cho tiện ích nên mặc định là 0 và người dùng phải đặt tùy chọn này bằng tay),
+ c cho hằng số c0 trong hàm kernel (mặc định là 0)
+ d cho mức độ của kernel (mặc định 3).

- Đặt độ lệch cho phép so với giá trị mong đợi trong Dung sai số. Đánh dấu vào ô bên cạnh Iteration Limit để đặt số lần lặp tối đa được phép.

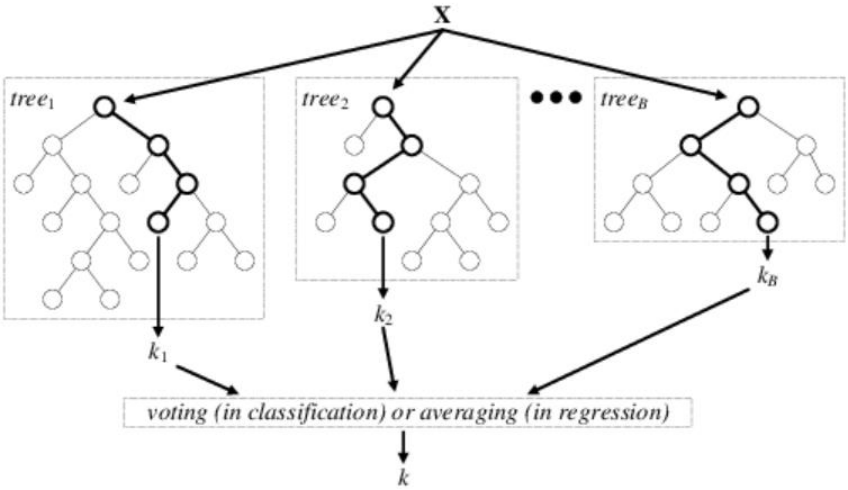
3.5.3. Kỹ thuật Random Forest – Phạm Thái Nguyên

3.5.3.1. Lý thuyết

a) Khái niệm

Random Forest là một phương pháp thống kê mô hình hóa bằng máy (machine learning statistic) dùng để phục vụ các mục đích phân loại, tính hồi quy và các nhiệm vụ khác bằng cách xây dựng nhiều cây quyết định (Decision Tree). Mỗi Node của cây là các thuộc tính, và các nhánh là giá trị lựa chọn của thuộc tính đó. Bằng cách đi theo các giá trị thuộc tính trên cây, cây quyết định sẽ cho ta biết giá trị dự đoán.

Random Forest có khả năng tìm ra thuộc tính nào quan trọng hơn so với những thuộc tính khác. Trên thực tế, nó còn có thể chỉ ra rằng một số thuộc tính không có tác dụng trong cây quyết định.



Hình 3.20: Sơ đồ biểu diễn các cây quyết định trong phương pháp Random Forest

Random là ngẫu nhiên, Forest là rừng, nên ở thuật toán Random Forest mình sẽ xây dựng nhiều cây quyết định bằng thuật toán Decision Tree, tuy nhiên mỗi cây quyết định sẽ khác nhau (có yếu tố random). Sau đó kết quả dự đoán được tổng hợp từ các cây quyết định.

b) Thuật toán mô hình bằng máy

Thuật toán lấy mẫu cho phương pháp random forest ứng dụng cho các phương pháp sử dụng thuật toán mô tả thống kê được ước lượng số lượng từ một mẫu dữ liệu (bagging). Ví

dụ như mốt tập mẫu X = x_1, …, x_n với các câu trả lời Y = y_1, …, y_n, lấy giá trị trung bình (B lần), chọn một mẫu ngẫu nhiên từ bộ mẫu phù hợp với cây quyết định:

Lặp b = 1, ..., B

n là mẫu giá trị tọa độ (X, Y); gọi là (X_b, Y_b); lớp dữ liệu hay kết quả hồi quy f_b của biến X_b, Y_b;

Sau khi lấy mẫu, các phép tính toán cho các mẫu là ẩn số x' có thể được thực hiện bằng cách lấy trung bình các giá trị nội suy từ tất cả các cây hồi quy riêng lẻ của biến x' hoặc lấy giá trị từ đa số các mẫu trong cây quyết định:

$$f = \frac{1}{B}\sum_{b=1}^{B} f_b(x')$$

Phương pháp thống kê máy này ước lượng một giá trị trung bình của số lượng dữ liệu. Chúng ta cần rất nhiều mẫu từ tập dữ liệu, tính giá trị trung bình. Sau đó, tính trung bình tất cả các giá trị trung bình của tập dữ liệu trong các cây quyết định thành phần để tính toán được kết quả tốt hơn giá trị trung bình thật. Kết quả dẫn đến hiệu suất mô hình được tính toán sẽ tốt hơn vì nó làm tang độ lệch. điều này có nghĩa là khi thiết kế nhiều quyết định trong một tập các mẫu được lấy sẽ đưa ra sự tương quan tốt hơn của các quyết định với nhau.

c) Các bước thực hiện của Random Forest

- **Bước 1: Chọn mẫu ngẫu nhiên.** Sử dụng kỹ thuật lấy mẫu có hoàn lại (bootstrapping) để chọn một tập hợp con ngẫu nhiên từ tập dữ liệu ban đầu cho mỗi cây quyết định.
- **Bước 2: Xây dựng cây quyết định.** Với mỗi tập hợp con, xây dựng một cây quyết định. Trong quá trình này, chọn ngẫu nhiên một số lượng nhất định các thuộc tính khi tách mỗi nút.
- **Bước 3: Lặp lại bước 1 và 2** cho đến khi đạt được số lượng cây quyết định mong muốn.
- **Bước 4: Kết hợp các cây.** Sử dụng phương pháp bỏ phiếu đa số (trong trường hợp phân loại) hoặc trung bình cộng (trong trường hợp hồi quy) từ tất cả các cây để đưa ra dự đoán cuối cùng.
- **Bước 5: Tinh chỉnh mô hình.** Áp dụng kiểm định chéo và điều chỉnh các tham số như số lượng cây quyết định để cải thiện hiệu suất mô hình.
- **Bước 6: Đánh giá mô hình.** Sử dụng tập dữ liệu kiểm tra để đánh giá hiệu suất của mô hình và xác định độ chính xác của nó.

d) Ưu và nhược điểm của Random Forest

- ***Ưu điểm:***
  - Độ chính xác cao: Sử dụng nhiều cây quyết định, mỗi cây được huấn luyện trên một tập con dữ liệu khác nhau, giúp giảm biến động và tăng độ chính xác.

- Kháng nhiễu và dữ liệu ngoại lai: Random Forest có khả năng chống lại nhiễu và dữ liệu ngoại lai, làm cho mô hình ổn định hơn.
- Quản lý dữ liệu đa chiều: Có khả năng xử lý hiệu quả các bộ dữ liệu có số chiều lớn và cung cấp ước lượng về mức độ quan trọng của từng đặc trưng.
- Hiệu quả cao: Random Forest có thể giải quyết cả bài toán regression và classification.
- Tính linh hoạt: Thuật toán này linh hoạt và dễ sử dụng, có thể áp dụng cho nhiều loại dữ liệu và vấn đề khác nhau.
- Xác định tầm quan trọng của tính năng: Cung cấp chỉ số về tầm quan trọng của các tính năng, giúp lựa chọn những tính năng quan trọng.

- ***Nhược điểm:***

- Tính toán phức tạp: Cần nhiều tài nguyên tính toán do xây dựng nhiều cây quyết định và kết hợp kết quả từ chúng.
- Thời gian huấn luyện: Mất nhiều thời gian để huấn luyện do phải kết hợp nhiều cây quyết định để xác định lớp.
- Khó giải thích: Do sự tổng hợp của nhiều cây quyết định, mô hình có thể khó giải thích và không xác định được tầm quan trọng của từng biến.

e) Ứng dụng

Trong lĩnh vực tài chính và kinh tế, nó được sử dụng để phân tích rủi ro và dự báo xu hướng thị trường.

- Dự báo giá nhà đất: sử dụng các thông tin về địa điểm, diện tích, số phòng…, random forest regression có thể dự báo giá nhà đất trong khu vực đó.
- Dự báo doanh số bán hàng: sử dụng thông tin về khách hàng, sản phẩm, điều kiện thời tiết…, random forest regression có thể dự báo doanh số bán hàng của một sản phẩm trong tương lai.
- Dự báo lượng mưa: sử dụng các thông tin về độ ẩm, nhiệt độ, tốc độ gió…, random forest regression có thể dự báo lượng mưa trong khu vực đó.
- Đánh giá rủi ro tín dụng: sử dụng thông tin về lịch sử thanh toán, thu nhập, dư nợ…, random forest regression có thể đánh giá rủi ro tín dụng của một cá nhân hoặc một doanh nghiệp.
- Xác định đối tượng mục tiêu: sử dụng các thông tin về đặc điểm khách hàng, lịch sử mua hàng…, random forest regression có thể xác định đối tượng mục tiêu để tập trung tiếp cận và quảng cáo cho hiệu quả hơn.

Khả năng xử lý dữ liệu phức tạp và cung cấp dự đoán chính xác, làm cho nó trở thành một công cụ hữu ích trong nhiều lĩnh vực khác nhau.

- Y tế: Thuật toán giúp dự đoán bệnh và phân loại bệnh nhân dựa trên các triệu chứng và kết quả xét nghiệm.
- Nông nghiệp: Dự đoán sản lượng nông nghiệp và phát hiện sâu bệnh trên cây trồng.
- Phát hiện gian lận: Được áp dụng để phát hiện gian lận trong các giao dịch tài chính.
- Tối ưu hóa công nghệ thông tin: Cải thiện hệ thống khuyến nghị và tìm kiếm, phân loại và sắp xếp dữ liệu lớn.
- Quản lý rủi ro: Đánh giá và quản lý rủi ro trong các dự án và quyết định kinh doanh.

3.5.3.2. Thực hành

Cho bộ dữ liệu huấn luyện sau:

ID	Mua nhà	Giá	Số phòng ngủ	Chỗ gửi xe	Cơ sở vật chất	Vị trí
1	Có	Cao	3	Có	Có	Gần
2	Không	Thấp	2	Không	Có	Xa
3	Có	Cao	4	Có	Không	Gần
4	Không	Cao	1	Không	Có	Gần
5	Có	Cao	3	Có	Có	Xa
6	Không	Cao	2	Có	Không	Gần
7	Không	Thấp	1	Có	Không	Xa
8	Có	Thấp	4	Không	Có	Gần
9	Có	Thấp	3	Không	Có	Xa
10	Không	Cao	2	Không	Có	Xa

Bảng 3.19: Bộ dữ liệu huấn luyện Random Forest

Các cột trong bộ dữ liệu này là:

- ID

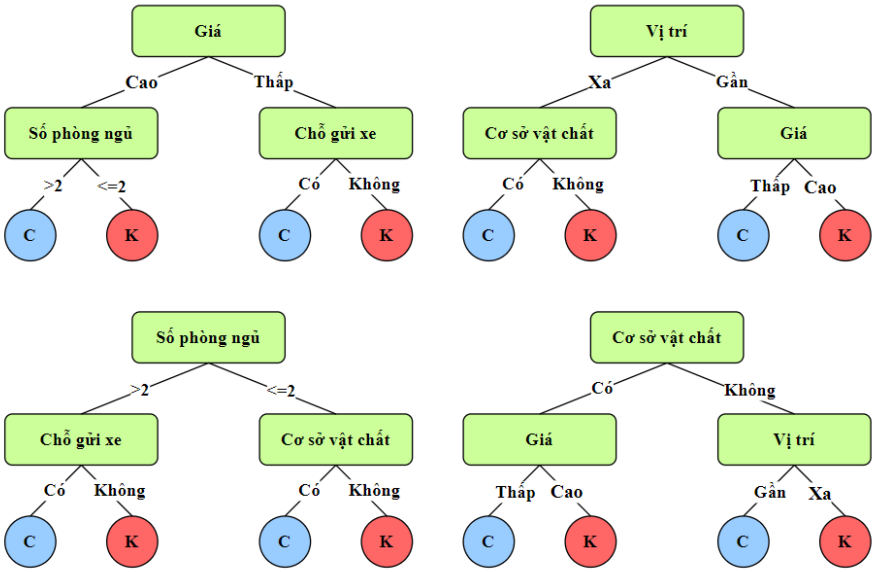
- Mua Nhà
- Giá
- Số phòng ngủ
- Chỗ gửi xe
- Cơ sở vật chất
- Vị trí (gần hay xa khu trung tâm)

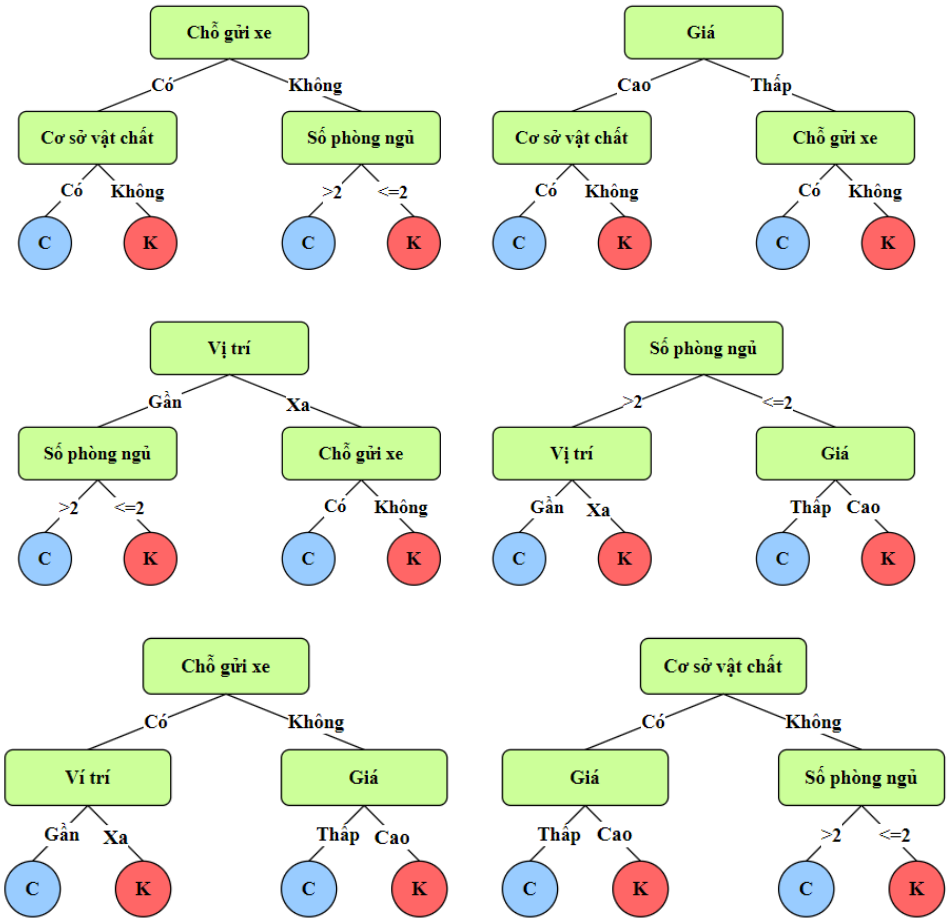
Và bộ dữ liệu kiểm định sau:

ID	Mua nhà	Giá	Số phòng ngủ	Chỗ gửi xe	Cơ sở vật chất	Vị trí
1	?	Cao	3	Không	Có	Xa
2	?	Thấp	2	Không	Không	Gần
3	?	Thấp	3	Có	Có	Gần
4	?	Cao	4	Có	Có	Xa
5	?	Cao	1	Không	Có	Gần

Bảng 3.20: Bộ dữ liệu kiểm định Random Forest

Dưới đây là 10 cây quyết định được xây dựng dựa trên 3 thuộc tính ngẫu nhiên từ bộ dữ liệu huấn luyện:





Hình 3.21: 10 cây quyết định ngẫu nhiên

Thực hiện thuật toán thống kê để ước lượng giá trị trung bình:

ID	Mua nhà	Giá	Số phòng ngủ	Chỗ gửi xe	Cơ sở vật chất	Vị trí
1	?	Cao	3	Không	Có	Xa

Với đầu tượng đầu tiên trong dữ liệu kiểm định, trong 10 cây quyết định bên trên có 4 cây cho kết quả "Có" và 6 cây cho kết quả "Không"; tỉ lệ 0.40: 0.60 → Không

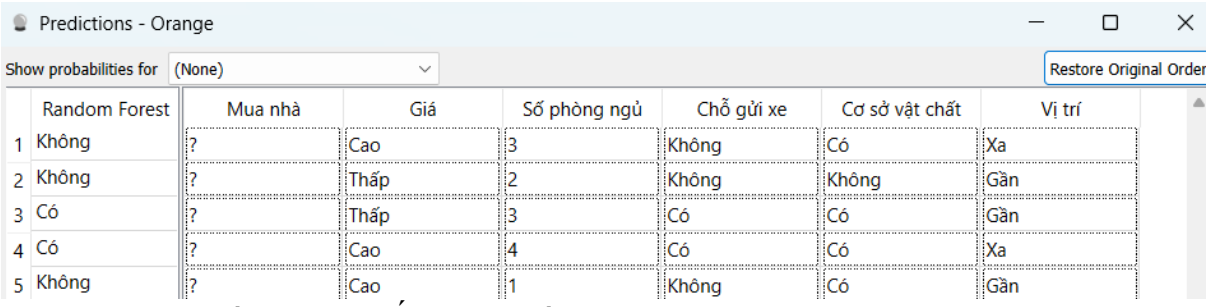
Thực hiện tương tự với các đối tượng còn lại trong bộ dữ liệu kiểm định ta được kết quả như sau:

ID	Mua nhà	Tỉ lệ	Giá	Số phòng ngủ	Chỗ gửi xe	Cơ sở vật chất	Vị trí
1	Không	0.40: 0.60	Cao	3	Không	Có	Xa
2	Không	0.40: 0.60	Thấp	2	Không	Không	Gần

3	Có	1.00: 0.00	Thấp	3	Có	Có	Gần
4	Có	0.60: 0.40	Cao	4	Có	Có	Xa
5	Không	0.20: 0.80	Cao	1	Không	Có	Gần

Bảng 3.21: Kết quả phân lớp thủ công Random Forest

Để tăng độ tin cậy, thực hiện chạy bộ dữ liệu trên Orange để kiểm tra độ chính xác của kết quả trên:

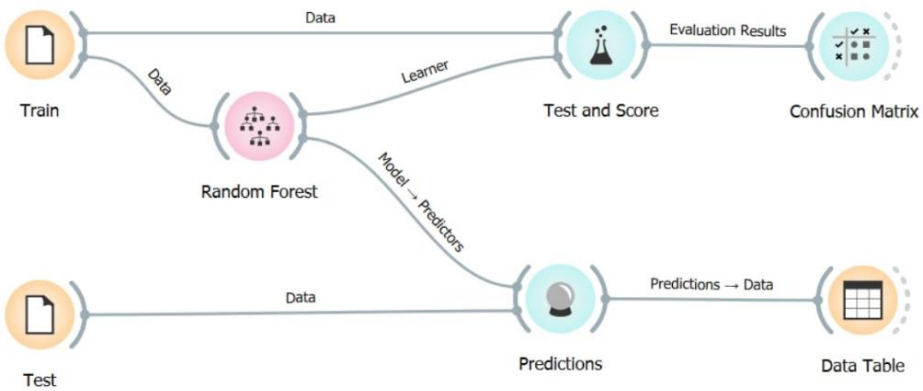


	Random Forest	Mua nhà	Giá	Số phòng ngủ	Chỗ gửi xe	Cơ sở vật chất	Vị trí
1	Không	?	Cao	3	Không	Có	Xa
2	Không	?	Thấp	2	Không	Không	Gần
3	Có	?	Thấp	3	Có	Có	Gần
4	Có	?	Cao	4	Có	Có	Xa
5	Không	?	Cao	1	Không	Có	Gần

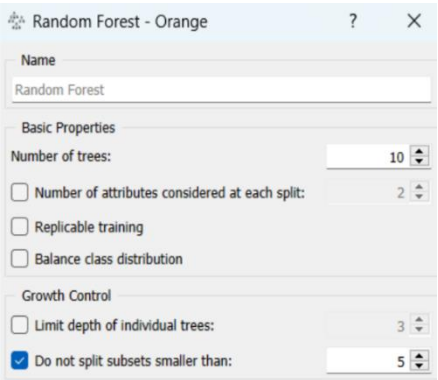
Hình 3.22: Kết quả so sánh Orange Random Forest

➔ Kết quả phân lớp dữ liệu không có sự khác biệt.

3.5.3.3. Xây dựng mô hình Random Forest – Dữ liệu nhóm



Hình 3.23: Xây dựng mô hình theo Random Forest



Hình 3.24: Random Forest Widget

Random Forest xây dựng một tập hợp các cây quyết định. Mỗi cây được phát triển từ một mẫu bootstrap từ dữ liệu huấn luyện. Khi phát triển từng cây, một tập hợp ngẫu nhiên các thuộc tính được rút ra, từ đó thuộc tính tốt nhất cho việc chia được chọn. Mô hình cuối cùng dựa trên số phiếu đa số từ các cây được phát triển riêng lẻ trong rừng.

Random Forest hoạt động cho cả nhiệm vụ phân loại và hồi quy. Dưới đây là các tùy chỉnh của Random Forest Widget trong phần mềm Orange:

- Number of trees: Số lượng cây quyết định trong rừng.
- Number of attributes considered at each split: Số lượng thuộc tính xem xét tại mỗi điểm chia.
- Replicable training: Đào tạo có thể tái tạo hay không.
- Balance class distribution: Cân bằng phân phối lớp.
- Limit depth of individual trees: Giới hạn độ sâu của từng cây.
- Do not split subsets smaller than: Không chia nhỏ hơn số lượng mẫu chỉ định.

3.5.4. Kỹ thuật Naive Bayes – Nguyễn Thị Xuân Nhi

3.5.4.1. Lý thuyết

a) Định lý Bayes

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 này được kí hiệu là P(A|B) được gọi là xác suất của A nếu có B. Đại lượng này được gọi là xác suất có điều kiện hay xác suất hậu nghiệm vì nó được rút ra từ giá trị được cho của B hoặc phụ thuộc vào giá trị đó.

Giả sử A và B là hai sự kiện đã xảy ra. Xác suất có điều kiện A khi biết trước điều kiện B được cho bởi:

$$P(A|B) = \frac{P(B|A) \text{ x } P(A)}{P(B)}$$

Với:

- P(A): Xác suất của sự kiện A xảy ra.
- P(B): Xác suất của sự kiện B xảy ra.
- P(B|A): Xác suất có điều kiện của sự kiện B xảy ra, nếu biết rằng sự kiện A xảy ra.
- P(A|B): Xác suất có điều kiện của sự kiện A xảy ra, nếu biết rằng sự kiện B xảy ra.

Ví dụ: Giả sử P(Lửa) là xác suất cháy và P(Khói) là xác suất ta nhìn thấy khói. Với các trường hợp sau:

P(Lửa|Khói): Xác suất cháy khi chúng ta nhìn thấy khói.

P(Khói|Lửa): Xác suất chúng ta nhìn thấy khói khi có cháy .

Nếu một đám cháy nguy hiểm có xác suất là 1% nhưng khói lại khá được nhìn thấy phổ biến do từ các nhà máy, xí nghiệp,...là 10% và 90% đám cháy nguy hiểm tạo ra khói . Ta có:

P(Khói) = 10%, P(Lửa) = 1%

P(Lửa|Khói) = 90%

Theo công thức ta có: P(Khói|Lửa) = $\frac{0.9 \text{ x } 0.01}{0.1}$= 9%

➔ Vậy khi thấy khói thì 9% đó là đám cháy nguy hiểm.

b) Naive Bayes Classifiers

Trong lĩnh vực thống kê, phân loại Naive Bayes là một nhóm các phân loại theo xác suất tuyến tính. Đặc điểm của phương pháp này là giả định các thuộc tính của dữ liệu độc lập với nhau và biết trước nhãn phân loại (biến mục tiêu). Naive Bayes được xem là 1 trong các mô hình phân loại đơn giản và hiệu quả.

Trong học máy, phân loại Naive Bayes là một mô hình học máy xác suất dựa trên định lý Bayes với giả định về tính độc lập giữa các yếu tố dự đoán. Nói cách khác, phân loại Naive Bayes giả định rằng sự tồn tại của một thuộc tính cụ thể trong một lớp không liên quan đến sự hiện diện của bất kỳ thuộc tính nào khác. Thuật toán này thuộc nhóm Supervised Learning (học có giám sát).

Một phân loại Naive Bayes dựa trên ý tưởng là một lớp được dự đoán bằng các giá trị của đặc trưng cho các thành viên của lớp đó. Các đối tượng là một nhóm trong các lớp nếu chúng có cùng chung các đặc trưng. Có thể có nhiều lớp rời rạc hoặc lớp nhị phân.

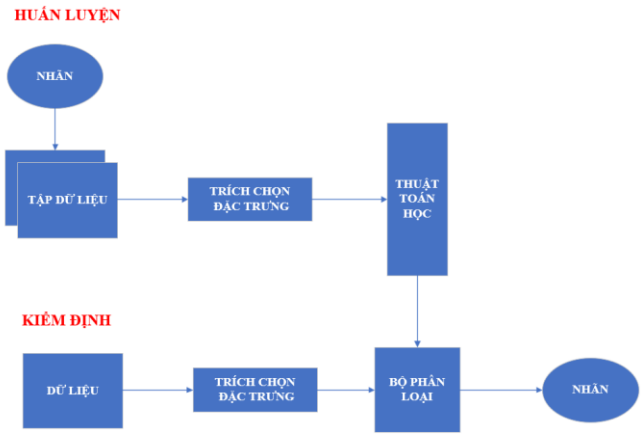
Các luật Bayes dựa trên xác suất để dự đoán chúng về các lớp có sẵn dựa trên các đặc trưng được trích rút. Trong phân loại Bayes, việc học được coi như xây dựng một mô hình xác suất của các đặc trưng và sử dụng mô hình này để dự đoán phân loại cho một ví dụ mới. Thuật toán Naive Bayes tính xác suất cho các yếu tố, sau đó chọn kết quả với xác suất cao nhất.

c) Các bước thực hiện thuật toán phân lớp Naive Bayes

Bước 1: Huấn luyện Naive Bayes (dựa vào tập dữ liệu), tính P(C_i) và P(x_i| C_i). Với:

- P(C_i): xác suất phân lớp i
- P(x_i| C_i): xác suất thuộc tính thứ k mang giá trị x_k khi biết mẫu X thuộc phân lớp i

Bước 2: Phân lớp X^{new}= ($x_1, x_2, \dots, x_n$), ta cần phải tính xác suất thuộc từng phân lớp khi đã biết trước X^{new}. X^{new} được gán vào lớp có xác suất lớn nhất.



Hình 3.25: Mô tả bước xây dựng mô hình phân lớp

d) Ưu và nhược điểm của Naive Bayes Classifiers

- **Ưu điểm:**
  - Việc dự đoán lớp của tập dữ liệu huấn luyện dễ dàng và nhanh chóng, hoạt động tốt trong dự đoán nhiều lớp.
  - Giả định tính độc lập được giữ nguyên, phân loại Naive Bayes hoạt động tốt hơn so với các mô hình khác.
  - Hoạt động tốt trong trường hợp các biến đầu vào phân loại so với các biến số. Đối với biến số, phân phối chuẩn được giả định (đường cong hình chuông).
- **Nhược điểm:**
  - Nếu biến phân loại có 1 danh mục (trong tập dữ liệu huấn luyện), không quan sát thấy trong tập dữ liệu huấn luyện thì mô hình sẽ gán xác suất bằng 0 và sẽ không thể đưa ra dự đoán.
  - Naive Bayes giả định về các yếu tố dự đoán độc lập. Trong cuộc sống, hầu như không thể có được một tập hợp các yếu tố dự đoán hoàn toàn độc lập.

e) Ứng dụng của Naive Bayes Classifiers

Thuật toán Naive Bayes Classifiers được áp dụng vào các loại ứng dụng sau:

- Real time Prediction: Naive Bayes Classifiers thích hợp áp dụng nhiều vào các ứng dụng chạy thời gian thực như hệ thống cảnh báo phát hiện sự cố,...
- Multi class Prediction: Nhờ vào định lý Bayes mở rộng ta có thể ứng dụng vào các loại ứng dụng đa dự đoán, tức là ứng dụng có thể đoán nhiều giả thuyết mục tiêu.

Định lý Bayes mở rộng:

$$P(A|B) = \frac{P(B_1|A) \text{ x } P(B_2|A) \text{ x } P(B_3|A) \text{ x } \dots \text{ x } P(B_n|A) \text{ x } P(A)}{P(B_1) \text{ x } P(B_2) \text{ x } P(B_3) \text{ x } \dots \text{ x } P(B_n)}$$

- Text classification/ Span filtering/ Sentiment Analysis: Naive Bayes cũng thích hợp cho các hệ thống phân loại văn bản hay ngôn ngữ tự nhiên vì tính chính xác của nó lớn hơn các thuật toán khác. Và các hệ thống phân tích tâm lý thị trường cũng có thể áp dụng thuật toán này để tiến hành phân tính tâm lý người dùng ưa chuộng hay không ưa chuộng các loại sản phẩm nào từ việc phân tích các thói quen và hành động của khách hàng.

3.5.4.2. Thực hành

Cho bộ dữ liệu huấn luyện sau:

Dựa vào các yếu tố về "Mây", "Áp suất" và "Gió" của các đối tượng để dự đoán xem trời có xảy ra mưa hay không.

Đối tượng	Mây	Áp suất	Gió	Kết quả
1	Ít	Cao	Bắc	Không mưa
2	Nhiều	Cao	Bắc	Mưa
3	Ít	Thấp	Bắc	Không mưa

4	Nhiều	Thấp	Bắc	Mưa
5	Nhiều	Trung bình	Bắc	Mưa
6	Ít	Cao	Nam	Không mưa
7	Nhiều	Cao	Nam	Mưa
8	Nhiều	Thấp	Nam	Không mưa
9	Nhiều	Trung bình	Bắc	Mưa
10	Nhiều	Cao	Nam	Mưa

Bảng 3.22: Bộ dữ liệu huấn luyện Naive Bayes

Và bộ dữ liệu kiểm định sau:

Đối tượng	Mây	Áp suất	Gió	Kết quả
1	Ít	Thấp	Nam	?
2	Nhiều	Cao	Bắc	?
3	Nhiều	Trung bình	Bắc	?
4	Ít	Cao	Nam	?
5	Ít	Trung bình	Nam	?
6	Ít	Thấp	Bắc	?
7	Nhiều	Thấp	Nam	?
8	Nhiều	Trung bình	Nam	?
9	Ít	Trung bình	Bắc	?
10	Nhiều	Cao	Nam	?

Bảng 3.23: Bộ dữ liệu kiểm định Naive Bayes

Dưới đây là cách sử dụng Naive Bayes trong phân lớp dữ liệu:

Đối với bộ dữ liệu này, sử dụng mô hình Multinomial Naive Bayes (dùng cho dữ liệu rời rạc). Với $P(x_1|c) = \frac{N_{ci}}{N_c}$

Trong đó:

- N_{ci} là tổng số lần xuất hiện của thuộc tính i trong data của lớp c
- N_c là tổng số data của lớp c

Ta sẽ lập bảng tính xác suất đối với bộ dữ liệu huấn luyện, sau đó sẽ tính toán và phân loại các dòng dữ liệu xem biến mục tiêu Kết quả là Mưa hay Không mưa bằng định lý Bayes. Cuối cùng nhập bộ dữ liệu vào orange để kiểm tra tính chính xác của kết quả tính được.

MÂY	Mưa	Không mưa
Ít	0	34
Nhiều	66	14

Bảng 3.24: Xác suất theo thuộc tính "Mây"

ÁP SUẤT	Mưa	Không mưa
Thấp	16	24
Cao	36	24
Trung bình	26	0

Bảng 3.25: Xác suất theo thuộc tính "Áp suất"

GIÓ	Mưa	Không mưa
Bắc	46	24
Nam	26	24

Bảng 3.26: Xác suất theo thuộc tính "Gió"

X	Mưa	Không mưa
P(X)	610	410

Bảng 3.27: Xác suất theo thuộc tính "Kết quả"

Trước khi tiến hành tính toán ở bộ dữ liệu kiểm định, ta thấy có 1 số thuộc tính có xác suất = 0, việc này sẽ làm ảnh hưởng đến kết quả tính toán bởi 0 nhân với bất kì số nào cũng bằng 0. Để khắc phục tình trạng này, ta sẽ sử dụng kỹ thuật Laplace Correction

$$P_{LAP,k}(x|y) = \frac{c(x,y) + k}{c(y) + k.|x|}$$

Với:

- $P_{LAP,k}(x|y)$: xác suất sau khi áp dụng Laplace Correction
- $c(x,y)$: số lần xuất hiện giá trị x trong lớp y
- $c(y)$: tổng số lượng quan sát trong lớp y
- k: hằng số dương thường được đặt là 1
- $|x|$: số lượng giá trị có thể nhận trong biến cần ước lượng xác suất

Ta sẽ tính toán trên bộ dữ liệu kiểm định và cho ra kết quả cột Kết quả.

Đối tượng	Mây	Áp suất	Gió	Kết quả
1	Ít	Thấp	Nam	?

- **Xác suất của đối tượng [1] nếu Kết quả là Mưa:**

P([1]|Mưa) = P(Ít|Mưa) x P(Thấp|Mưa) x P(Nam|Mưa)

$= \frac{0+1}{6+2} \text{ x } \frac{1+1}{6+3} \text{ x } \frac{2+1}{6+2} = \frac{1}{96}$

- **Xác suất của đối tượng [1] nếu Kết quả là Không mưa:**

P([1]|Không mưa) = P(Ít| Không mưa) x P(Thấp| Không mưa) x P(Nam| Không mưa)

$= \frac{3+1}{4+2} \text{ x } \frac{2+1}{4+3} \text{ x } \frac{2+1}{4+2} = \frac{1}{7}$

- **Xác suất của đối tượng [1] xảy ra:**

P([1]) = P([1]|Mưa) x P(Mưa) + P([1]|Không mưa) x P(Không mưa)

$= \frac{1}{96} \text{ x } \frac{6+1}{10+2} + \frac{1}{7} \text{ x } \frac{4+1}{10+2} = \frac{529}{8064}$

- **Xác suất của Kết quả là Mưa nếu [1] xảy ra:**

$$P(\text{Mưa}|[1]) = \frac{P([1]|\text{Mưa}) \text{ x } P(\text{Mưa})}{P([1])} = \frac{\frac{1}{96} \text{ x } \frac{6+1}{10+2}}{\frac{529}{8064}} = 0.0926$$

- **Xác suất của Kết quả là Không mưa nếu [1] xảy ra:**

$$P(\text{Không mưa}|[1]) = \frac{P([1]|\text{Không mưa}) \text{ x } P(\text{Không mưa})}{P([1])} = \frac{\frac{1}{7} \text{ x } \frac{4+1}{10+2}}{\frac{529}{8064}} = 0.9074$$

➔ 0.9074 > 0.0926 ➔ Với đối tượng [1] thì kết quả là **Không mưa**

Thực hiện tương tự với các dòng dữ liệu còn lại trong bộ dữ liệu kiểm định ta được kết quả như sau:

Đối tượng	Mây	Áp suất	Gió	Kết quả	P(Không Mưa)	P(Mưa)
1	Ít	Thấp	Nam	Không mưa	0.9074	0.0926
2	Nhiều	Cao	Bắc	Mưa	0.173	0.826
3	Nhiều	Trung bình	Bắc	Mưa	0.085	0.915
4	Ít	Cao	Nam	Không mưa	0.831	0.169
5	Ít	Trung bình	Nam	Không mưa	0.685	0.314
6	Ít	Thấp	Bắc	Không mưa	0.854	0.145
7	Nhiều	Thấp	Nam	Mưa	0.411	0.588
8	Nhiều	Trung bình	Nam	Mưa	0.134	0.865
9	Ít	Trung bình	Bắc	Không mưa	0.566	0.433
10	Nhiều	Cao	Nam	Mưa	0.258	0.741

Bảng 3.28: Kết quả xác suất phân lớp Naive Bayes

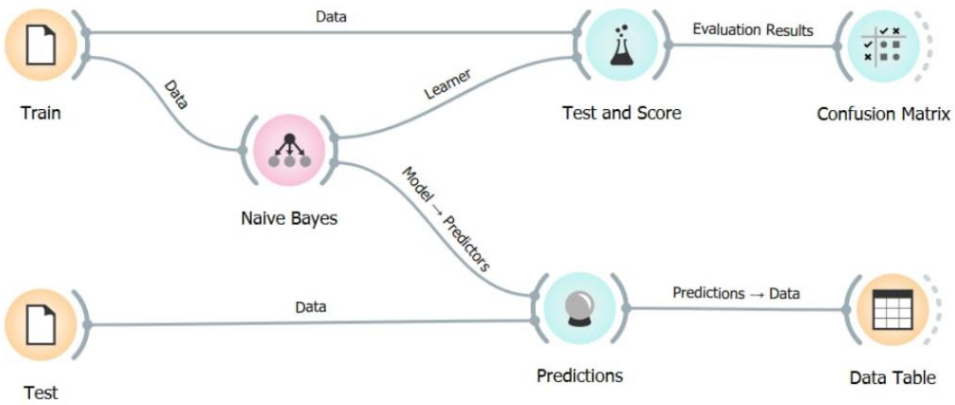
Để tăng độ tin cậy, thực hiện chạy bộ dữ liệu trên Orange để kiểm tra độ chính xác của kết quả trên:

	Naive Bayes (1)	Mây	Áp suất	Gió	Kết quả
1	0.92 : 0.08 → Không mưa	Ít	Thấp	Nam	?
2	0.20 : 0.80 → Mưa	Nhiều	Cao	Bắc	?
3	0.10 : 0.90 → Mưa	Nhiều	Trung bình	Bắc	?
4	0.85 : 0.15 → Không mưa	Ít	Cao	Nam	?
5	0.72 : 0.28 → Không mưa	Ít	Trung bình	Nam	?
6	0.88 : 0.12 → Không mưa	Ít	Thấp	Bắc	?
7	0.46 : 0.54 → Mưa	Nhiều	Thấp	Nam	?
8	0.16 : 0.84 → Mưa	Nhiều	Trung bình	Nam	?
9	0.61 : 0.39 → Không mưa	Ít	Trung bình	Bắc	?
10	0.30 : 0.70 → Mưa	Nhiều	Cao	Nam	?

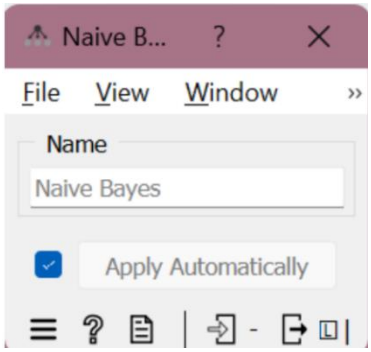
Hình 3.26: Kết quả so sánh với Orange Naive Bayes

Mặc dù có sự chênh lệch đôi chút về kết quả xác suất tuy nhiên kết quả phân lớp dữ liệu không có sự khác biệt.

3.5.4.3. Xây dựng mô hình Naive Bayes – Dữ liệu nhóm



Hình 3.27: Xây dựng mô hình theo Naive Bayes



Hình 3.28: Naive Bayes Widget

Naive Bayes là phương pháp phân loại theo xác suất dựa trên định lý Bayes, tương đối đơn giản, hiện tại phần mềm Orange không hỗ trợ điều chỉnh các thông số của Naive Bayes widget.

3.5.5. Kỹ thuật Logistic Regression – Lê Như Thi

3.5.5.1. Lý thuyết

a) Định nghĩa

Hồi quy logistic là một phương pháp thống kê được sử dụng để dự đoán xác suất của một biến phụ thuộc rời rạc dựa trên các biến độc lập. Trong hồi quy logistic, biến phụ thuộc thường là một biến nhị phân (có hai lựa chọn: 0 hoặc 1), như có hoặc không, đạt mục tiêu hay không đạt mục tiêu.

Cụ thể, hồi quy logistic sử dụng hàm logistic để biểu diễn mối quan hệ giữa các biến độc lập và xác suất của biến phụ thuộc. Hàm logistic giúp chuyển đổi giá trị liên tục của biến độc lập thành một phân phối xác suất nằm trong khoảng từ 0 đến 1.

Mô hình hồi qui *Logistic* là sự tiếp nối ý tưởng của hồi quy tuyến tính vào các bài toán phân loại. Từ đầu ra của hàm tuyến tính chúng ta đưa vào hàm *Sigmoid* để tìm ra phân phối xác suất của dữ liệu. Lưu ý rằng hàm *Sigmoid* chỉ được sử dụng trong bài toán phân loại nhị phân.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Phương pháp này thường được sử dụng để dự đoán xác suất của một sự kiện xảy ra hoặc không xảy ra dựa trên các biến đầu vào.

Ví dụ: Dự báo xác suất để một người không trả được nợ. Tính xác suất vỡ nợ/không vỡ nợ.

- Gọi biến phụ thuộc Vỡ nợ là y (y=1: vỡ nợ, y=0: không vỡ nợ)
- Dự báo P(y=1)
- P(y=0) = 1 - P(y=1)
- Các biến độc lập là $x_1, x_2, x_3, x_4 \dots x_k$ ví dụ:
    - Thu nhập hàng tháng
    - Tuổi
    - Trình độ học vấn
- Sử dụng hàm logistic: để tính xác suất trong khoảng từ (0,1).

$$P = \frac{1}{1 + e^{-(b_o + b_1X_1 + b_2X_2 + \cdots + b_kX_k)}}$$

- Cuối cùng kết quả của kỹ thuật logistic sẽ lấy giá trị sau khi dùng hàm logistic so với giá trị ngưỡng (thường là 0,5) để làm tròn đến các giá trị gần nhất là 0 hoặc 1 nhằm đưa ra kết luận cuối cùng là người đó vỡ nợ/không vỡ nợ.

b) Các bước thực hiện

Bước 1: Thu thập dữ liệu

Thu thập dữ liệu về biến phụ thuộc (binary outcome) và biến độc lập mà bạn muốn sử dụng để dự đoán biến phụ thuộc. Đảm bảo dữ liệu phù hợp và được làm sạch.

Bước 2: Chuẩn bị dữ liệu

Tiền xử lý dữ liệu bằng cách loại bỏ dữ liệu trống, xử lý ngoại lệ, mã hóa biến độc lập (nếu cần) và chia tập dữ liệu thành tập huấn luyện và tập kiểm tra.

Bước 3: Xây dựng mô hình hồi quy logistic.

Sử dụng hàm logistic để xây dựng mô hình hồi quy logistic. Mô hình này có dạng:

$$P = \frac{1}{1 + e^{-(b_o + b_1X_1 + b_2X_2 + \cdots + b_kX_k)}}$$

Trong đó:

- P là xác suất sự kiện xảy ra
- $X_1, X_2, \ldots, X_p$ là các biến độc lập
- $b_0, b_1, b_2, \ldots, b_p$ là các hệ số hồi quy mô hình cần được ước tính.

Bước 4: Ước tính các hệ số

Trong mô hình hồi quy logistic, hệ số (hay còn gọi là trọng số) là các giá trị được ước lượng cho mỗi biến độc lập trong mô hình. Mỗi biến độc lập được nhân với một hệ số tương ứng để dự đoán xác suất của biến phụ thuộc nhị phân.

Sử dụng dữ liệu để ước tính các hệ số (b0, b1, b2 ,…,bp) bằng cách tối ưu hóa hàm loss function, thường là hàm cross-entropy, thông qua các thuật toán tối ưu hóa như Gradient Descent.

Bước 5: Đánh giá mô hình

Sử dụng tập kiểm tra để đánh giá hiệu suất của mô hình hồi quy logistic. Các phép đo thường bao gồm confusion matrix, precision, recall, F1-score, ROC curve, và AUC.

Bước 6: Tinh chỉnh và sử dụng mô hình

Tùy chỉnh mô hình nếu cần thiết, bằng cách điều chỉnh siêu tham số hoặc thay đổi biến độc lập. Sau đó, bạn có thể sử dụng mô hình đã huấn luyện để dự đoán xác suất hoặc phân loi cho các điểm dữ liệu mới.

c) Ưu và nhược điểm của Logistic Regression

- **Ưu điểm:**

  - Phân loại rõ ràng: Mô hình hồi quy logistic thích hợp cho các bài toán phân loại, đặc biệt là khi có hai nhóm phân loại (binary classification).
  - Dễ hiểu và triển khai: Mô hình này dựa trên ý tưởng đơn giản của hồi quy tuyến tính, làm cho nó dễ hiểu và triển khai trong các tình huống thực tế.

- Không yêu cầu dữ liệu phân phối chuẩn: Logistic regression không đòi hỏi dữ liệu phải tuân theo phân phối chuẩn, điều này giúp nó phù hợp với nhiều loại dữ liệu thực tế.
- Ít cần thiết kỹ thuật tiền xử lý: Mặc dù việc chuẩn hóa dữ liệu có thể cần thiết trong một số trường hợp, nhưng logistic regression ít đòi hỏi các kỹ thuật tiền xử lý phức tạp so với một số mô hình phân loại khác.

- **Nhược điểm:**
  - Giả sử đặc biệt của dữ liệu: Logistic regression giả định một mối quan hệ tuyến tính giữa các biến đầu vào và đầu ra, điều này có thể là một hạn chế trong trường hợp dữ liệu không tuân theo mô hình tuyến tính.
  - Không thể xử lý mối quan hệ phi tuyến tính: Mô hình logistic regression không thể xử lý được mối quan hệ phi tuyến tính giữa các biến đầu vào và đầu ra một cách hiệu quả.
  - Dễ bị overfitting: Khi có quá nhiều biến đầu vào hoặc dữ liệu không đủ, mô hình logistic regression có thể dễ dàng bị overfitting, làm giảm khả năng tổng quát hóa của nó đối với dữ liệu mới.
  - Không xử lý được các biến đầu vào liên tục: Logistic regression mặc định làm việc tốt với các biến đầu vào phân loại hoặc biến đầu vào liên tục đã được rời rạc hóa trước đó. Trong một số trường hợp, việc xử lý các biến đầu vào liên tục có thể làm giảm hiệu suất của mô hình.

d) Ứng dụng hàm Logistic

Hồi quy logistic có một số ứng dụng thực tế trong nhiều ngành công nghiệp khác nhau:

Sản xuất: Các công ty sản xuất áp dụng phân tích hồi quy logistic để ước tính xác suất xảy ra sự cố ở bộ phận trong máy móc. Sau đó, họ sẽ lên lịch bảo trì dựa trên xác suất đã ước tính này để giảm thiểu sự cố trong tương lai.

Chăm sóc sức khỏe: Các nhà nghiên cứu y khoa lên kế hoạch điều trị và chăm sóc dự phòng bằng cách dự đoán khả năng mắc bệnh ở bệnh nhân. Họ sử dụng các mô hình hồi quy logistic để so sánh tác động của tiền sử gia đình hoặc của bộ gen lên bệnh tật.

Tài chính: Các công ty tài chính phải phân tích các giao dịch tài chính để đề phòng gian lận, xem xét các đơn xin vay và đơn bảo hiểm để đề phòng rủi ro. Những vấn đề này phù hợp với mô hình hồi quy logistic bởi chúng có kết quả cụ thể, chẳng hạn như rủi ro cao hoặc rủi ro thấp và gian lận hoặc không gian lận.

Bộ phận tiếp thị: Các công cụ quảng cáo trực tuyến sử dụng mô hình hồi quy logistic để dự đoán xem người dùng sẽ nhấp vào một quảng cáo hay không. Kết quả là, các nhà tiếp thị có thể phân tích phản ứng của người dùng đối với những từ ngữ và hình ảnh khác nhau, tạo ra các quảng cáo hiệu suất cao có khả năng thu hút khách hàng.

3.5.5.2. Thực hành

Cho bộ dữ liệu huấn luyện sau:

Biến phụ thuộc Kết quả với 0 là không mắc bệnh và 1 là mắc bệnh, các biến độc lập bao gồm Tuổi, Tỷ lệ cholesterol, BMI, Huyết áp, Tỷ lệ Glucose.

TUỔI	TỶ LỆ CHOLESTEROL	BMI	HUYẾT ÁP	TỶ LỆ GLUCOSE	KẾT QUẢ
39	195	26.97	80	77	0
46	250	28.73	95	76	0
48	245	25.34	75	70	0
61	225	28.58	65	103	1
46	285	23.1	85	85	0
43	228	20.3	77	99	0
63	205	33.11	60	85	1
45	313	21.68	79	78	0
38	221	21.35	95	70	1
46	294	26.31	98	64	0
38	195	23.26	75	78	0
50	254	22.91	75	76	0
46	291	23.38	80	89	1

Bảng 3.29: Bộ dữ liệu huấn luyện Logistic Regression

Và bộ dữ liệu kiểm định sau:

TUỔI	TỶ LỆ CHOLESTEROL	BMI	HUYẾT ÁP	TỶ LỆ GLUCOSE	KẾT QUẢ
52	260	26.36	76	79	?
43	225	23.61	93	88	?
46	294	26.31	98	64	?

41	332	31.31	65	84	?
48	232	22.37	64	72	?

Bảng 3.30: Bộ dữ liệu kiểm định Logistic Regression

Dưới đây là cách sử dụng mô hình Logistic regression để dự đoán xác suất mắc bệnh/không mắc bệnh tiểu đường.

Ta có hàm logistic như sau:

$$P = \frac{1}{1 + e^{-(b_o + b_1X_1 + b_2X_2 + \cdots + b_kX_k)}}$$

Gọi biến phụ thuộc là Y (Y = 1: mắc bệnh, Y = 0: không mắc bệnh)

Các biến độc lập x1, x2, x3, x4 , x5 lần lượt là Tuổi, Tỷ lệ Cholesterol, BMI, Huyết áp, Tỷ lệ Glucose.

Xác định hệ số b0, b1, b2, b3, b4, b5 bằng cách sử dụng phần mềm IBM SPSS STATISTICS để tính giá trị hệ số ước lượng

Variables in the Equation

		B	S.E.	Wald	df	Sig.	Exp(B)
Step 1[a]	Ty_le_Glucoso	,054	,087	,378	1	,538	1,055
	Huyet_ap	,119	,141	,716	1	,397	1,127
	BMI	-,205	,401	,261	1	,610	,815
	Ty_le_Cholesterol	-,028	,032	,771	1	,380	,973
	Tuoi	,303	,291	1,088	1	,297	1,355
	Constant	-17,125	18,329	,873	1	,350	,000

a. Variable(s) entered on step 1: Ty_le_Glucoso, Huyet_ap, BMI, Ty_le_Cholesterol, Tuoi.

Hình 3.29: Hệ số ước lượng thông qua SPSS

Ta có: Cột B là các hệ số ước tính tương ứng với từng biến độc lập.

Ta chọn ngưỡng là 0.5 về xác suất để đưa ra kết quả 0 và 1 để dự báo kết quả quan sát.

Thực hiện tính toán với hàng đầu tiên:

TUỔI	TỶ LỆ CHOLESTEROL	BMI	HUYẾT ÁP	TỶ LỆ GLUCOSE	KẾT QUẢ
52	260	26.36	76	79	?

Áp dụng công thức hàm logistic tính toán xác suất xảy ra:

$$P = \frac{1}{1 + e^{-17,125 + 0,303\ x\ 52 + -0,028\ x\ 260 + -0,205\ x\ 26,36 + 0,119\ x\ 76 + 0,054\ x\ 79)}}$$

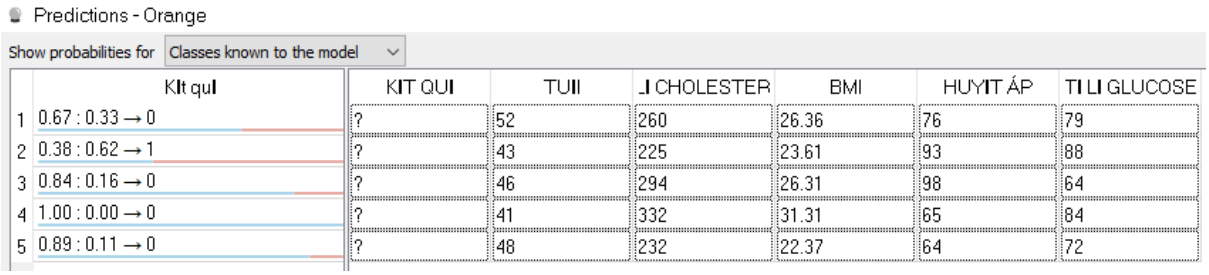
Xác suất là P = 0,32239 < ngưỡng 0,5 nên suy ra Kết quả là 0 có ý nghĩa là người này có khả năng không mắc bệnh.

Áp dụng tương tự cho các dòng dữ liệu còn lại ta được bảng kết quả sau:

TUỔI	TỶ LỆ CHOLESTEROL	BMI	HUYẾT ÁP	TỶ LỆ GLUCOSE	P	KẾT QUẢ
52	260	26.36	76	79	0,3224	?
43	225	23.61	93	88	0,6417	?
46	294	26.31	98	64	0,1552	?
41	332	31.31	65	84	0,0003	?
48	232	22.37	64	72	0,1035	?

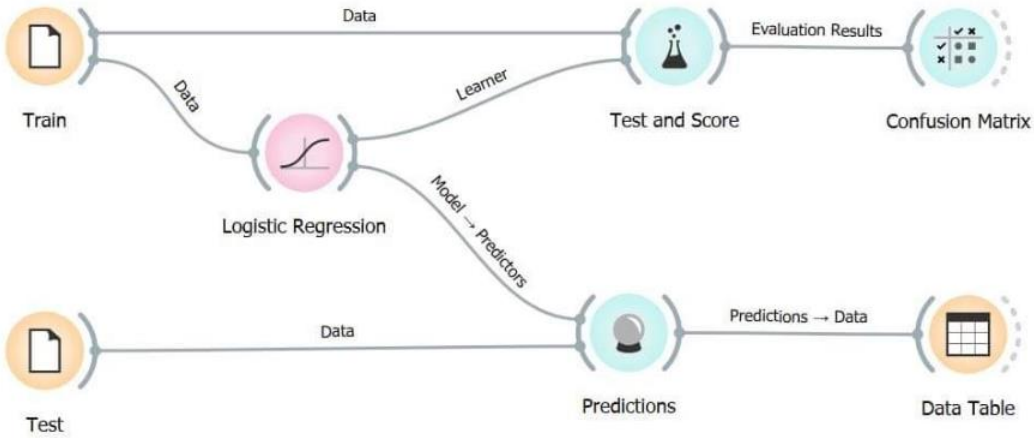
Bảng 3.31: Kết quả xác suất phân lớp thủ công Logistic Regression

Để tăng độ tin cậy, thực hiện chạy bộ dữ liệu trên Orange để kiểm tra độ chính xác của kết quả trên: kết quả có sai số về xác suất tuy nhiên kết quả phân loại không có gì khác biệt.

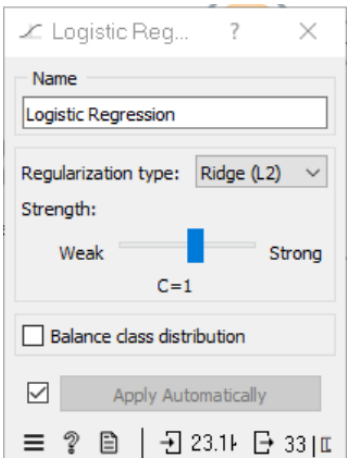


Hình 3.30: Kết quả so sánh Orange Logistic Regression

3.5.5.3. Xây dựng mô hình theo Logistic Regression – Dữ liệu nhóm



Hình 3.31: Xây dựng mô hình theo Logistic Regression



Hình 3.32: Logistic Regression Widget

Logistic Regression hay còn gọi là Hồi quy logit là một công cụ học máy cho phép người dùng lựa chọn biến phụ thuộc và các biến độc lập, thuật toán hiển thị các thông số hệ số hồi quy, xác suất, các độ đo và dự đoán kết quả phân loại nhị phân dựa trên các biến độc lập.

Các thông số có thể điều chỉnh:

- **Regularization type:** giúp kiểm soát sự phức tạp của mô hình bằng cách thêm một thành phần phạt vào hàm mất mát (loss function) của mô hình, làm cho mô hình trở nên đơn giản hơn và giảm thiểu nguy cơ overfitting. Có hai thành phần phạt là: L1 Regularization và L2 Regularization. **Regularization** có ảnh hưởng đến tính toán của hệ số hồi quy và điều này có thể làm thay đổi giá trị và phân phối của các hệ số hồi quy trong quá trình huấn luyện mô hình.
- **Strength (tham số C):** Tham số C trong hồi quy logistic là một hệ số regularization dương, và giá trị của nó đặc trưng cho mức độ của regularization được áp dụng vào mô hình.
  + Giá trị của C càng lớn, mức độ regularization càng yếu: có nghĩa là mô hình sẽ có xu hướng phân loại các mẫu huấn luyện chính xác hơn, nhưng cũng có nguy cơ cao hơn về overfitting dữ liệu huấn luyện.
  + Giá trị của C càng nhỏ, mức độ regularization càng mạnh, giúp giảm thiểu nguy cơ overfitting nhưng có thể làm giảm hiệu suất của mô hình trên dữ liệu kiểm tra.
  + Thông thường tham số C được đặt bằng 1.
- **Balance class distribution:** Trong hồi quy logit, cân bằng phân phối lớp (balance class distribution) có ảnh hưởng đến việc ước lượng mô hình và đánh giá hiệu suất của nó. Nó cũng phụ thuộc vào mục tiêu và tập dữ liệu. Ở trường hợp này ta không dùng cân bằng phân phối lớp.

3.5.6. Kỹ thuật k-Nearest Neighbors (kNN) – Trần Thế Hào

3.5.6.1. Lý thuyết

a) Khái niệm

kNN (K-Nearest Neighbors) là một trong những thuật toán học có giám sát đơn giản nhất được sử dụng nhiều trong khai phá dữ liệu và học máy. Ý tưởng của thuật toán này là nó

không học một điều gì từ tập dữ liệu học (nên kNN được xếp vào loại lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán nhãn của dữ liệu mới.

Lớp (nhãn) của một đối tượng dữ liệu mới có thể dự đoán từ các lớp (nhãn) của k hàng xóm gần nó nhất

b) Ý tưởng

Thuật toán kNN cho rằng những dữ liệu tương tự nhau sẽ tồn tại **gần nhau** trong một không gian, từ đó công việc của chúng ta là sẽ tìm k điểm gần với dữ liệu cần kiểm tra nhất. Việc tìm khoảng cách giữa 2 điểm cũng có nhiều công thức có thể sử dụng, tùy trường hợp mà chúng ta lựa chọn cho phù hợp. Đây là 3 cách cơ bản để tính khoảng cách 2 điểm dữ liệu x, y có k thuộc tính. Khoảng cách có thể được tính theo các chuẩn Euclidean, Manhattan hoặc Minkowski…

Distance functions

Euclidean $$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Manhattan $$\sum_{i=1}^{k}|x_i - y_i|$$

Minkowski $$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$$

Trong đó, x_i và y_i lần lượt là tọa độ của điểm cần phân loại và điểm lân cận, k là số điểm lân cận được chọn.

Số k là siêu tham số của mô hình (hyperparameter), các giá trị khác nhau của k có thể dẫn đến các kết luận khác nhau. Nếu k là số chẵn, có thể không có phân loại rõ ràng. Chọn giá trị cho k quá nhỏ sẽ dẫn đến tỉ lệ lỗi cao và độ nhạy đối với các điểm dữ liệu bất thường mang tính cục bộ. Nhưng chọn giá trị cho k quá lớn sẽ làm giảm đi tính chất khái niệm láng giềng gần nhất vì lấy trung bình quá nhiều kết quả.

c) Các bước thực hiện

Để thực hiện bài toán kNN cần 6 bước chính như sau:

1. Ta có D là tập các điểm dữ liệu đã được gắn nhãn và A là dữ liệu chưa được phân loại.
2. Đo khoảng cách (Euclidean, Manhattan, Minkowski, Minkowski hoặc Trọng số) từ dữ liệu mới A đến tất cả các dữ liệu khác -đã được phân loại trong D.
3. Chọn k (k là tham số mà bạn định nghĩa) khoảng cách nhỏ nhất.
4. Kiểm tra danh sách các lớp có khoảng cách ngắn nhất và đếm số lượng của mỗi lớp xuất hiện.
5. Lấy đúng lớp (lớp xuất hiện nhiều lần nhất).

6. Lớp của dữ liệu mới là lớp mà bạn đã nhận được ở bước 5.

d) Ưu và nhược điểm của kNN

- **Ưu điểm**
  - Đơn giản và dễ giải thích
  - Không dựa trên bất kỳ giả định nào, vì thế nó có thể được sử dụng trong các bài toán phi tuyến tính.
  - Hoạt động tốt trong trường hợp phân loại với nhiều lớp
  - Sử dụng được trong cả phân loại và hồi quy
- **Nhược điểm**
  - Trở nên rất chậm khi số lượng điểm dữ liệu tăng lên vì mô hình cần lưu trữ tất cả các điểm dữ liệu.
  - Tốn bộ nhớ
  - Nhạy cảm với các dữ liệu bất thường (nhiễu)

e) Ứng dụng

Về ứng dụng của thuật toán KNN phải kể đến như:

+ Trong y tế: xác định bệnh lý của người bệnh mới dựa trên dữ liệu lịch sử của các bệnh nhân có cùng bệnh lý có cùng các đặc điểm đã được chữa khỏi trước đây, hay xác định loại thuốc phù hợp giống ví dụ chúng tôi trình bày ở trên.
+ Trong lĩnh vực ngân hàng: xác định khả năng khách hàng chậm trả các khoản vay hoặc rủi ro tín dụng do nợ xấu dựa trên phân tích Credit score; xác định xem liệu các giao dịch có hành vi phạm tội, lừa đảo hay không.
+ Trong giáo dục: phân loại các học sinh theo hoàn cảnh, học lực để xem xem cần hỗ trợ gì cho những học sinh ví dụ như hoàn cảnh sống khó khăn nhưng học lực lại tốt.
+ Trong thương mại điện tử: phân loại khách hàng theo sở thích cụ thể để hỗ trợ personalized marketing hay xây dựng hệ thống khuyến nghị, dựa trên dữ liệu từ website, social media.
+ Trong kinh tế nói chung: giúp dự báo các sự kiện kinh tế trong tương lai, dự báo tình hình thời tiết trong nông nghiệp, xác định xu hướng thị trường chứng khoán để lên kế hoạch đầu tư thích hợp.

3.5.6.2. Thực hành

Cho bộ dữ liệu huấn luyện sau:

Đối tượng	Giống loài	Chiều dài lá (Cm)	Chiều rộng lá (Cm)	Chiều Dài Cánh Hoa (Cm)	Chiều Rộng Cánh Hoa (Cm)
1	Iris-setosa	51	35	14	2
2	Iris-setosa	49	30	14	2

3	Iris-setosa	47	32	13	2
4	Iris-setosa	46	31	15	2
5	Iris-setosa	48	34	19	2
6	Iris-versicolor	49	24	33	10
7	Iris-versicolor	50	20	35	10
8	Iris-versicolor	50	23	33	10
9	Iris-virginica	76	30	66	21
10	Iris-virginica	77	38	67	22
11	Iris-virginica	77	30	61	23
12	Iris-virginica	79	38	64	20

Bảng 3.32: Bộ dữ liệu huấn luyện kNN

Các cột trong bộ dữ liệu này là:

+ Đối tượng
+ Giống loài
+ Chiều dài lá (Cm)
+ Chiều rộng lá (Cm)
+ Chiều Dài Cánh Hoa (Cm)
+ Chiều Rộng Cánh Hoa (Cm)

Và bộ dữ liệu kiểm định sau:

Đối tượng	**Giống loài**	**Chiều dài lá (Cm)**	**Chiều rộng lá (Cm)**	**Chiều Dài Cánh Hoa (Cm)**	**Chiều Rộng Cánh Hoa (Cm)**
1	?	52	18	32	8
2	?	45	30	15	2
3	?	45	32	12	3
4	?	51	34	13	1
5	?	75	20	50	20
6	?	65	25	30	15

Bảng 3.33: Bộ dữ liệu kiểm định kNN

Dưới đây là cách sử dụng kNN trong phân lớp dữ liệu:

Đối với bộ dữ liệu này, sử dụng công thức Euclidean:

Euclidean $$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Trong đó:

- x_i và y_i lần lượt là tọa độ của điểm cần phân loại và điểm lân cận,
- k là số điểm lân cận được chọn.

Ta sẽ lập bảng tính khoảng cách với bộ dữ liệu huấn luyện, sau đó sẽ lựa chọn k điểm dữ liệu huấn luyện gần nhất với mỗi dữ liệu kiểm định nhằm đưa ra kết luận các dòng dữ liệu xem biến mục tiêu Giống loài là Iris-setosa, Iris-versicolor hay là Iris-virginica . Cuối cùng nhập bộ dữ liệu vào Orange để kiểm tra tính chính xác của kết quả tính được.

Ta sẽ tính toán trên bộ dữ liệu kiểm định và cho ra kết quả cột Giống loài.

Đối tượng	**Giống loài**	**Chiều dài lá (Cm)**	**Chiều rộng lá (Cm)**	**Chiều Dài Cánh Hoa (Cm)**	**Chiều Rộng Cánh Hoa (Cm)**
1	?	52	18	32	8

Khoảng cách giữa Đối tượng kiểm định [1] với Đối tượng huấn luyện [1]:

$$D1 = \sqrt{(52 - 51)^2 + (18 - 35)^2 + (32 - 14)^2 + (8 - 2)^2} = 25,5$$

Tương tự ta tính toán khoảng cách giữa Đối tượng kiểm định [1] với từng Đối tượng huấn luyện còn lại, ta được:

D1 = 25,5	D7 = 4,59
D2 = 22,65	D8 = 5,84
D3 = 24,86	D9 = 45,23
D4 = 23,03	D10 = 49,46
D5 = 21,85	D11 = 42,84
D6 = 7,08	D12 = 47,93

Tính toán khoảng cách đối với bộ huấn luyện

Tiếp theo ta chọn k: Xác định số lượng nearest neighbor ta muốn sử dụng trong quá trình tìm kiếm, trong trường hợp này chọn k = 5 – lấy ra 5 Đối tượng dữ liệu huấn luyện gần Đối tượng kiểm định [1] nhất, ta được bảng sau:

STT	Distance
1	D7 = 4,59
2	D8 = 5,84
3	D6 = 7,08
4	D5 = 21,85
5	D2 = 22,65

Bảng 3.34: Xác định lượng Nearest Neighbor

Đếm số lượng của mỗi lớp xuất hiện trong bảng 5 khoảng cách gần nhất. Lấy đúng lớp (lớp xuất hiện nhiều lần nhất): Trong 5 Đối tượng huấn luyện ta nhận thấy có 3 đối tượng đầu có Giống loài là Iris-versicolor, 2 đối tượng sau có Giống loài là Iris-setosa.

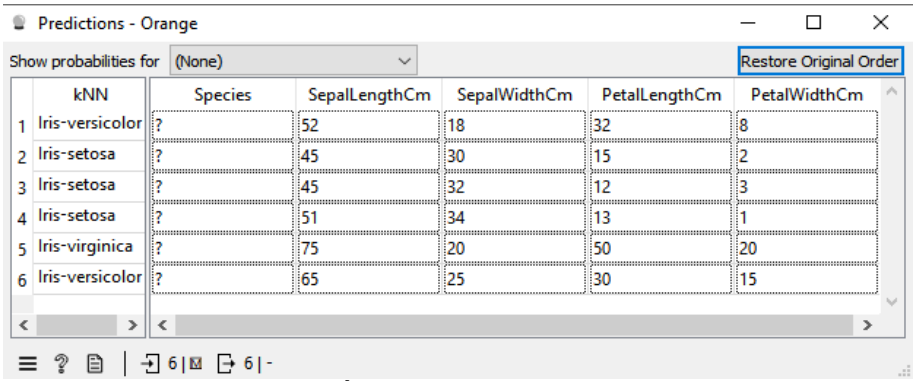
➔ Giống loài của Đối tượng kiểm định [1] là Iris- versicolor.

Thực hiện tương tự với các Đối tượng kiểm định còn lại trong bộ dữ liệu kiểm định ta được kết quả như sau:

Đối tượng	**Giống loài**	**Chiều dài lá (Cm)**	**Chiều rộng lá (Cm)**	**Chiều Dài Cánh Hoa (Cm)**	**Chiều Rộng Cánh Hoa (Cm)**
1	Iris-versicolor	52	18	32	8
2	Iris-setosa	45	30	15	2
3	Iris-setosa	45	32	12	3
4	Iris-setosa	51	34	13	1
5	Iris-virginica	75	20	50	20
6	Iris-versicolor	65	25	30	15

Bảng 3.35: Kết quả phân lớp thủ công kNN

Để tăng độ tin cậy, thực hiện chạy bộ dữ liệu trên Orange để kiểm tra độ chính xác của kết quả trên:

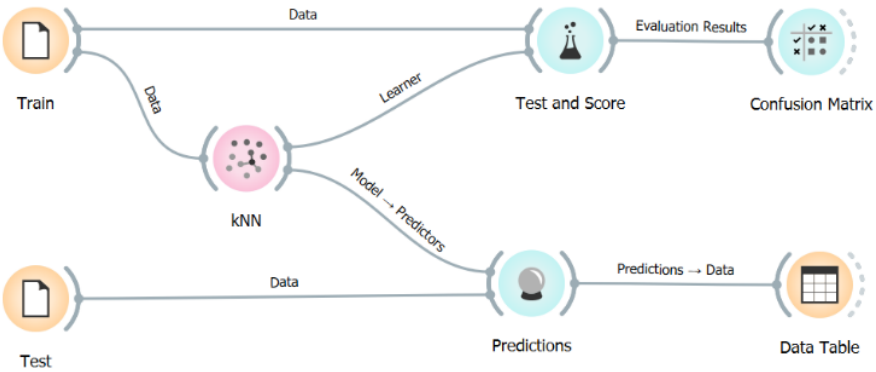


	kNN	Species	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
1	Iris-versicolor	?	52	18	32	8
2	Iris-setosa	?	45	30	15	2
3	Iris-setosa	?	45	32	12	3
4	Iris-setosa	?	51	34	13	1
5	Iris-virginica	?	75	20	50	20
6	Iris-versicolor	?	65	25	30	15

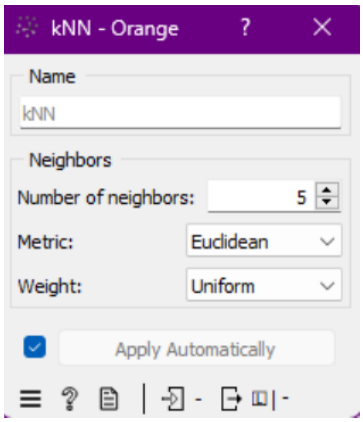
Hình 3.33: Kết quả so sánh Orange kNN

➔ Kết quả phân lớp dữ liệu không có sự khác biệt.

3.5.6.3. Xây dựng mô hình theo kNN – Dữ liệu nhóm



Hình 3.34: Xây dựng mô hình theo kNN



Hình 3.35: kNN Widget

kNN widget sử dụng thuật toán kNN đi tìm đầu ra của một điểm dữ liệu mới bằng cách *chỉ* dựa trên thông tin của k điểm dữ liệu trong training set gần nó nhất (k-lân cận)

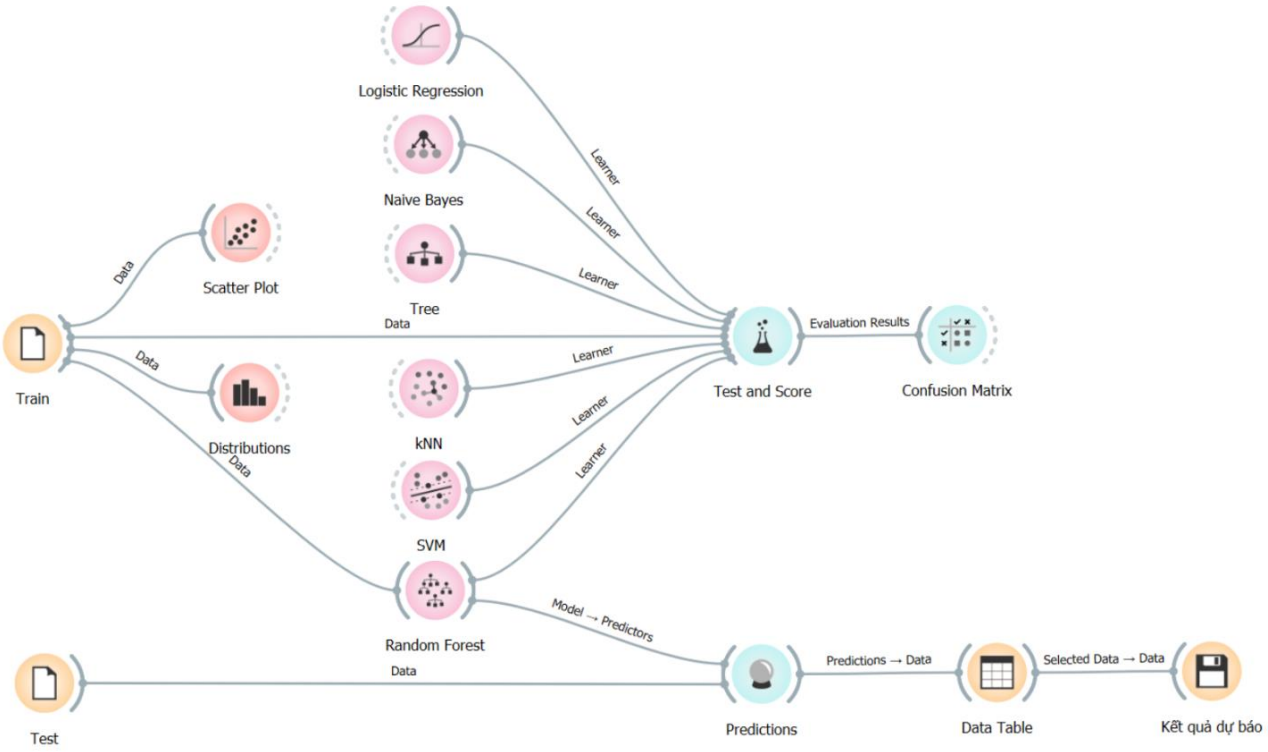
Thông số có thể điều chỉnh:

- Number of neighbors - Số lượng lân cận gần nhất
- Metric - Số liệu:
  - Euclidean ("đường thẳng", khoảng cách giữa hai điểm)

- Manhattan (tổng chênh lệch tuyệt đối của tất cả các thuộc tính)
- Maximal (sự khác biệt tuyệt đối lớn nhất giữa các thuộc tính)
- Mahalanobis (khoảng cách giữa điểm và phân phối).

- Weights - Trọng số:
  - Uniform: tất cả các điểm trong vùng lân cận đều có trọng số như nhau.
  - Distance: các lân cận gần hơn của điểm truy vấn có ảnh hưởng lớn hơn các lân cận ở xa hơn.

3.6. Xây dựng mô hình



Hình 3.36: Mô hình phân lớp dữ liệu để dự báo tình trạng đặt phòng của khách sạn

- **Bước 1:** Chọn dữ liệu file "Hotel Reservations", và chọn cột "booking_status" làm target cho file "Train" và file "Test".
- **Bước 2:** Mở file "Hotel Reservations". Nối file Train với Scatter Plot và Distributions để quan sát thuộc tính.
- **Bước 3:** Nối file "Train" với 6 phương pháp SVM, Tree, Logistic Regression, Naive Bayes, Random Forest, kNN với Test and Score để huấn luyện mô hình. Trong Test and Score ta dùng Cross Validation chia tập train ban đầu thành k phần tương đương nhau, mỗi phần được gọi là một fold. Sau đó, sử dụng k-1 folds để huấn luyện mô hình. Nối Test and Score với file"Train" và Confusion Matrix để thực hiện đánh giá kết quả.
- **Bước 4:** Liên kết phương pháp phân lớp tốt nhất với file "Train" và Predictions. Đồng thời nối file "Test" là dữ liệu kiểm thử với Predictions để đánh giá và dự báo dữ liệu đầu vào.
- **Bước 5:** Xuất kết quả ra Data Table và lưu thành file là "kết quả dự báo.xlsx"

3.7. Kết quả và đánh giá

❖ **Đánh giá dựa trên kết quả Test and Score:**

Xem xét các chỉ số và lựa chọn mô hình phù hợp nhất cho nghiên cứu. Trong nghiên cứu này, phương pháp đánh giá mô hình phân lớp với Cross Validation với Number of fold là 5 (= 5) để đánh giá với tính năng vượt trội hơn và tránh trùng lặp giữa các tập kiểm thử.

1. *Chỉ số AUC (Area Under the Curve):*
   - AUC đo lường khả năng phân biệt giữa các lớp trong mô hình phân loại.
   - Giá trị AUC nằm trong khoảng từ 0 đến 1. Một AUC gần 1 cho thấy mô hình hoạt động tốt trong việc phân loại.
2. *Chỉ số F1:*
   - F1 là một phép đo kết hợp giữa độ chính xác và độ phủ của mô hình.
   - Giá trị F1 nằm trong khoảng từ 0 đến 1. F1 gần 1 cho thấy mô hình có độ chính xác cao và độ phủ tốt.

❖ **Đánh giá dựa trên kết quả Confusion Matrix:**

Sai lầm loại 1: Dự báo rằng phòng không bị hủy nhưng thực tế thì phòng bị hủy.

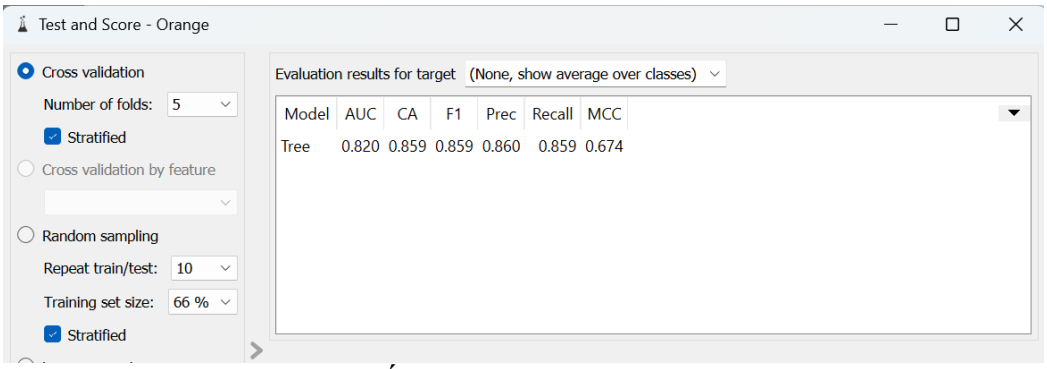
Sai lầm loại 2: Dự báo rằng phòng hủy nhưng thực tế thì phòng không bị hủy.

Khi dự báo khách hàng không hủy đặt phòng mà trên thực tế khách hàng lại hủy phòng. Vì điều này sẽ làm cho khách sạn thiệt hại doanh thu, khi dự báo sai công suất phòng dẫn đến việc khách sạn không sử dụng hết tất cả các phòng đã dự tính. Dẫn đến thiệt hại doanh thu khách sạn và gây dư thừa phòng trống. Ảnh hưởng đến hoạt động của khách sạn, việc hủy đặt phòng ảnh hưởng trực tiếp đến hoạt động của khách sạn. Các phòng trống không được sử dụng tối ưu, và việc quản lý phòng trở nên phức tạp hơn.

3.7.1. Theo kỹ thuật cá nhân

a) Tree

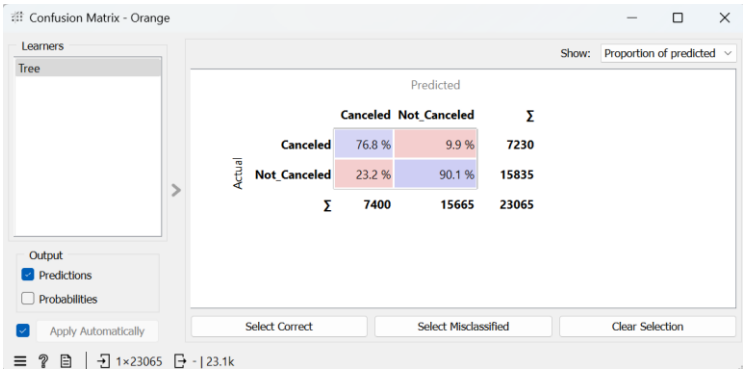
❖ **Đánh giá dựa trên kết quả Test and Score:**



Hình 3.37: Kết quả Test and Score kỹ thuật Tree

Theo lý thuyết, chỉ số AUC càng lớn (càng tiến về 1) thì mô hình càng tốt. Các chỉ số khác của phương pháp Tree đều cao, trong đó chỉ số AUC là 0.820. Có thể xem mô hình phân lớp bằng Tree ở bộ dữ liệu này tương đối khá.

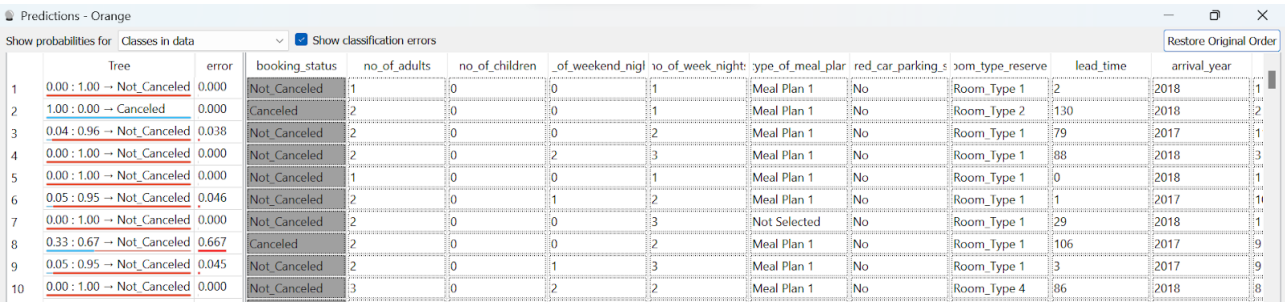
❖ **Đánh giá dựa trên kết quả Confusion Matrix:**



Hình 3.38: Kết quả Confusion Matrix của Tree

- Dự báo phòng bị huỷ và thực tế phòng bị hủy: 76.8%
- Dự báo phòng không bị hủy nhưng thực tế phòng bị hủy: 9.9%
- Dự báo phòng bị hủy nhưng thực tế phòng không bị hủy: 23.2%
- Dự báo phòng không bị hủy và thực tế phòng không bị hủy: 90.1%

❖ **Kết quả dự báo bằng Tree:**



	Tree	error	booking_status	no_of_adults	no_of_children	_of_weekend_nigl	no_of_week_night:	ype_of_meal_plar	red_car_parking_s	om_type_reserve	lead_time	arrival_year	
1	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	2	2018	1
2	1.00 : 0.00 → Canceled	0.000	Canceled	2	0	0	1	Meal Plan 1	No	Room_Type 2	130	2018	2
3	0.04 : 0.96 → Not_Canceled	0.038	Not_Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	79	2017	1
4	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	2	0	2	3	Meal Plan 1	No	Room_Type 1	88	2018	3
5	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	0	2018	1
6	0.05 : 0.95 → Not_Canceled	0.046	Not_Canceled	2	0	1	2	Meal Plan 1	No	Room_Type 1	1	2017	1
7	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	2	0	0	3	Not Selected	No	Room_Type 1	29	2018	1
8	0.33 : 0.67 → Not_Canceled	0.667	Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	106	2017	9
9	0.05 : 0.95 → Not_Canceled	0.045	Not_Canceled	2	0	1	3	Meal Plan 1	No	Room_Type 1	3	2017	9
10	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	3	0	2	2	Meal Plan 1	No	Room_Type 4	86	2018	8

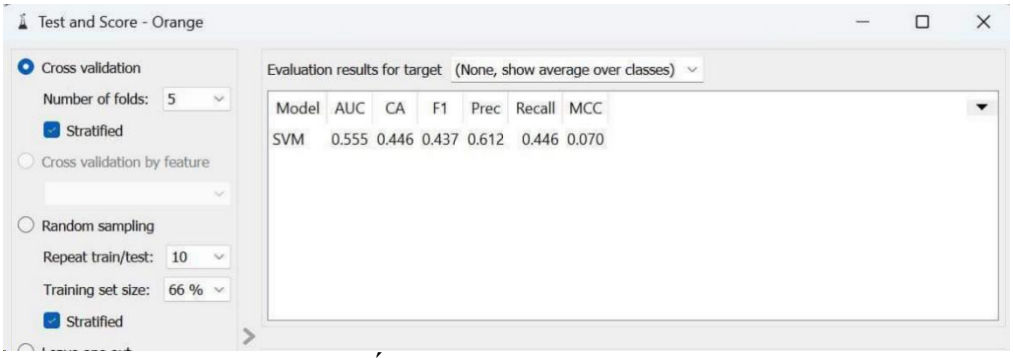
Hình 3.39: Kết quả dự báo bằng kỹ thuật Tree

❖ **Đánh giá phương pháp:**

Trong phương pháp Tree, các chỉ số đánh giá đều khá cao trong đó chỉ số AUC là 0.820. Có thể xem mô hình phân lớp bằng Tree ở bộ dữ liệu này khá tốt. Bên cạnh đó, mô hình có tổng số sai lầm loại 1 và 2 tương đối thấp (9.9% và 23.2%). Với kết quả trên, có thể thấy rằng việc áp dụng phương pháp Tree vào bài toán dự báo này là phù hợp với mức độ tin cậy cao về chỉ số Test and Score và Confusion Matrix, từ đó đưa ra được những dự báo tốt, giúp cải thiện hoạt động kinh doanh của khách sạn.

b) SVM

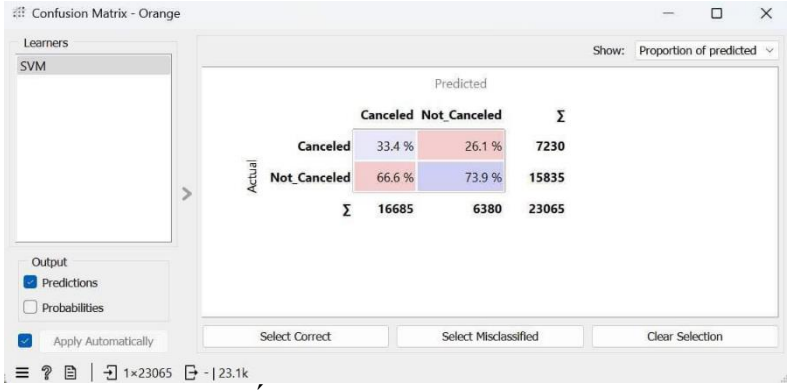
❖ **Đánh giá dựa trên kết quả Test and Score:**



Hình 3.40: Kết quả Test and Score kỹ thuật SVM

Theo lý thuyết, chỉ số AUC càng lớn (càng tiến về 1) thì mô hình càng tốt. Các chỉ số khác của phương pháp SVM đều thấp. Trong trường hợp này, AUC là 0.555, không quá cao, có thể cần xem xét cải thiện mô hình đồng thời chỉ số F1 là 0.437 cũng không đạt hiệu suất tốt.

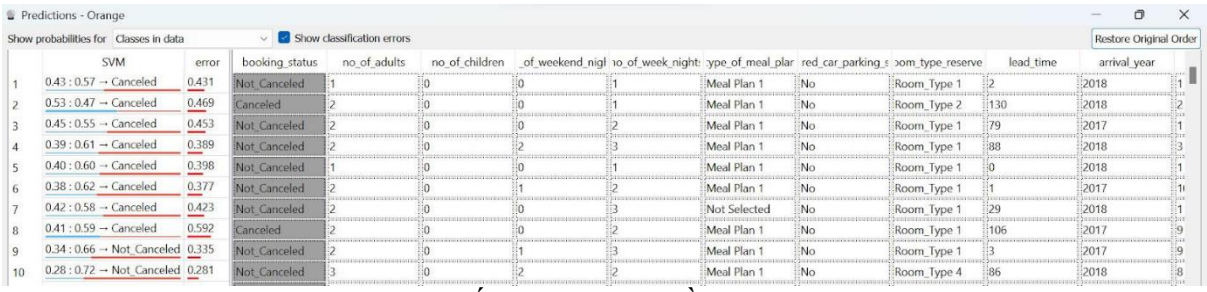
❖ **Đánh giá dựa trên kết quả Confusion Matrix:**



Hình 3.41: Kết quả Confusion Matrix của SVM

- Dự báo phòng bị huỷ và thực tế phòng bị hủy: 33.4%
- Dự báo phòng không bị hủy nhưng thực tế phòng bị hủy: 26.1%
- Dự báo phòng bị hủy nhưng thực tế phòng không bị hủy: 66.6%
- Dự báo phòng không bị hủy và thực tế phòng không bị hủy: 73.9%

❖ **Kết quả dự báo bằng SVM:**



	SVM	error	booking_status	no_of_adults	no_of_children	_of_weekend_nigl	no_of_week_night	ype_of_meal_plar	red_car_parking_s	oom_type_reserve	lead_time	arrival_year	
1	0.43 : 0.57 → Canceled	0.431	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	2	2018	1
2	0.53 : 0.47 → Canceled	0.469	Canceled	2	0	0	1	Meal Plan 1	No	Room_Type 2	130	2018	2
3	0.45 : 0.55 → Canceled	0.453	Not_Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	79	2017	1
4	0.39 : 0.61 → Canceled	0.389	Not_Canceled	2	0	2	3	Meal Plan 1	No	Room_Type 1	88	2018	3
5	0.40 : 0.60 → Canceled	0.398	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	0	2018	1
6	0.38 : 0.62 → Canceled	0.377	Not_Canceled	2	0	1	2	Meal Plan 1	No	Room_Type 1	1	2017	1
7	0.42 : 0.58 → Canceled	0.423	Not_Canceled	2	0	0	3	Not Selected	No	Room_Type 1	29	2018	1
8	0.41 : 0.59 → Canceled	0.592	Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	106	2017	9
9	0.34 : 0.66 → Not_Canceled	0.335	Not_Canceled	2	0	1	3	Meal Plan 1	No	Room_Type 1	3	2017	9
10	0.28 : 0.72 → Not_Canceled	0.281	Not_Canceled	3	0	2	2	Meal Plan 1	No	Room_Type 4	86	2018	8

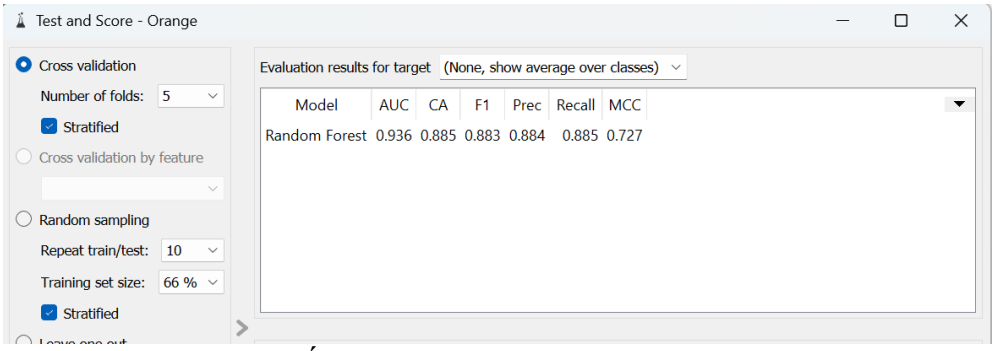
Hình 3.42: Kết quả dự báo bằng kỹ thuật SVM

❖ **Đánh giá phương pháp:**

Qua các kết quả đánh giá với AUC là 0,555 chỉ số này không cao, tương tự các chỉ số khác cũng tương đối thấp, có thể việc sử dụng mô hình này cần xem xét lại. Ngoài ra mô hình có tổng số sai lầm loại 1 và 2 khá cao (26.1% và 66.6%), với kết quả này, nếu áp dụng mô hình vào bài toán dự báo này có thể là không thực sự phù hợp , với độ tin cậy thấp về cả chỉ số Test and Score và Confusion Matrix có thể đưa ra nhiều dự báo sai gây ảnh hưởng đến hoạt động kinh doanh của khách sạn.

c) Random Forest

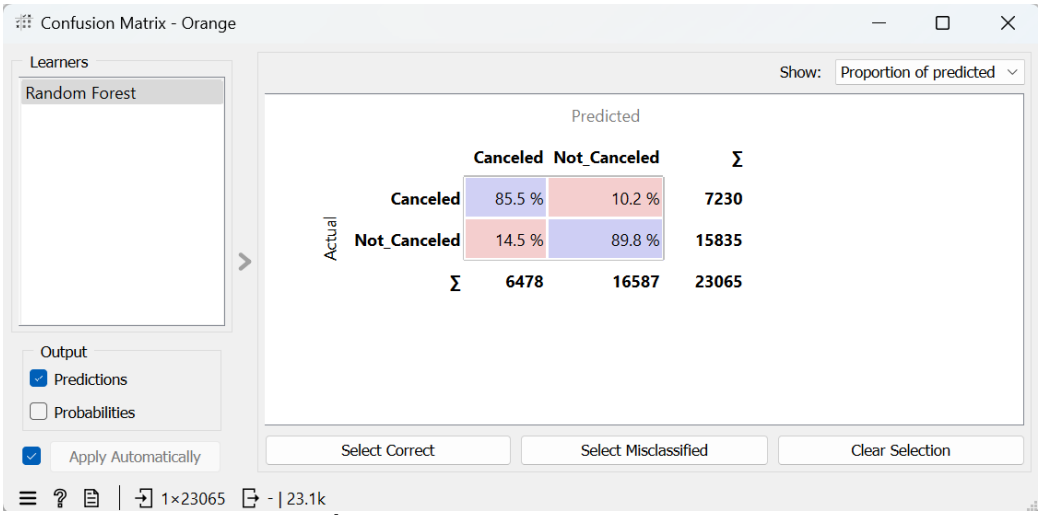
❖ **Đánh giá dựa trên kết quả Test and Score:**



Hình 3.43: Kết quả Test and Score của Random Forest

Theo lý lý thuyết, chỉ số AUC càng lớn (càng tiến về 1) thì mô hình càng tốt. Các chỉ số khác của phương pháp Random Forest tương đối cao, trong đó AUC cao nhất (0.936). Có thể xem mô hình phân lớp bằng Random Forest ở bộ dữ liệu này tương đối tốt.

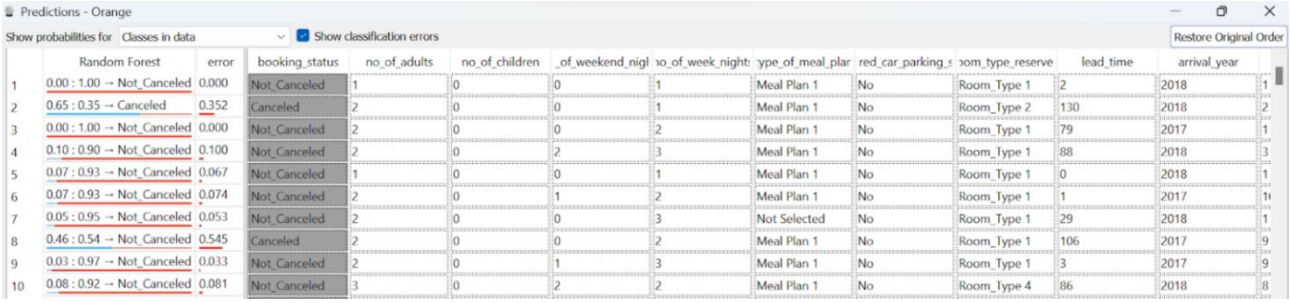
❖ **Đánh giá dựa trên kết quả Confusion Matrix:**



Hình 3.44: Kết quả Confusion Matrix của Random Forest

- Dự báo phòng bị huỷ và thực tế phòng bị hủy: 85.5%
- Dự báo phòng không bị hủy nhưng thực tế phòng bị hủy: 10.2%
- Dự báo phòng bị hủy nhưng thực tế phòng không bị hủy: 14.5%
- Dự báo phòng không bị hủy và thực tế phòng không bị hủy: 89.8%

❖ **Kết quả dự báo bằng Random Forest:**



Predictions - Orange

Show probabilities for Classes in data | Show classification errors | Restore Original Order

	Random Forest	error	booking_status	no_of_adults	no_of_children	_of_weekend_nigl	o_of_week_night:	ype_of_meal_plar	red_car_parking_s	oom_type_reserve	lead_time	arrival_year	
1	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	2	2018	1
2	0.65 : 0.35 → Canceled	0.352	Canceled	2	0	0	1	Meal Plan 1	No	Room_Type 2	130	2018	2
3	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	79	2017	1
4	0.10 : 0.90 → Not_Canceled	0.100	Not_Canceled	2	0	2	3	Meal Plan 1	No	Room_Type 1	88	2018	3
5	0.07 : 0.93 → Not_Canceled	0.067	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	0	2018	1
6	0.07 : 0.93 → Not_Canceled	0.074	Not_Canceled	2	0	1	2	Meal Plan 1	No	Room_Type 1	1	2017	1
7	0.05 : 0.95 → Not_Canceled	0.053	Not_Canceled	2	0	0	3	Not Selected	No	Room_Type 1	29	2018	1
8	0.46 : 0.54 → Not_Canceled	0.545	Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	106	2017	9
9	0.03 : 0.97 → Not_Canceled	0.033	Not_Canceled	2	0	1	3	Meal Plan 1	No	Room_Type 1	3	2017	9
10	0.08 : 0.92 → Not_Canceled	0.081	Not_Canceled	3	0	2	2	Meal Plan 1	No	Room_Type 4	86	2018	8

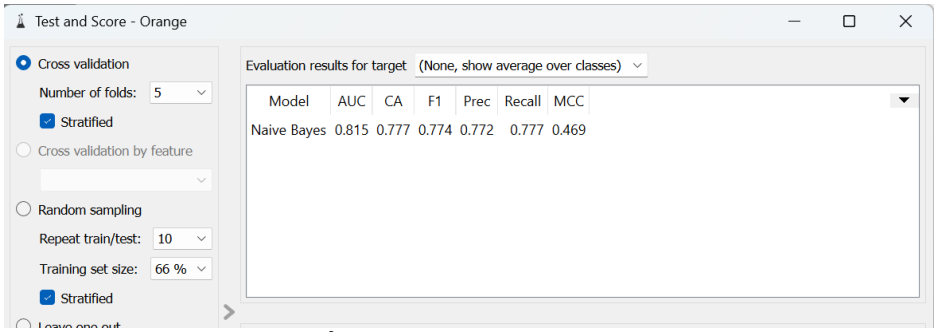
Hình 3.45: Kết quả dự báo bằng kỹ thuật Random Forest

❖ **Đánh giá phương pháp:**

Đối với phương pháp Random Forest, các chỉ số khác đều khá cao, trong đó chỉ số AUC là 0.936. Có thể xem mô hình phân lớp bằng Random Forest ở bộ dữ liệu này tốt. Cùng với đó là phương pháp này có phần trăm sai lầm loại 1 và 2 ở mức thấp (10.2% và 14.5%). Với kết quả khả quan này cho thấy việc áp dụng phương pháp Random Forest vào bài toán dự báo này là phù hợp, cùng với mức độ tin cậy của các chỉ số Test and Score và Confusion Matrix, từ đó đưa ra được những dự báo tốt, giúp cải thiện hoạt động kinh doanh của khách sạn.

d) Naive Bayes

❖ **Đánh giá dựa trên kết quả Test and Score:**



Test and Score - Orange

Cross validation
Number of folds: 5
Stratified
Cross validation by feature
Random sampling
Repeat train/test: 10
Training set size: 66 %
Stratified

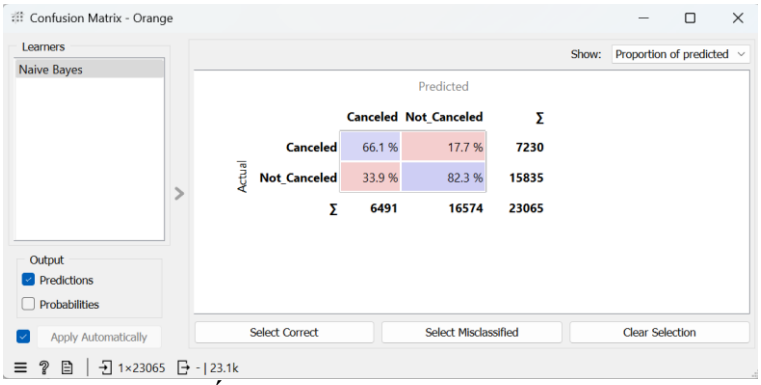
Evaluation results for target (None, show average over classes)

Model	AUC	CA	F1	Prec	Recall	MCC
Naive Bayes	0.815	0.777	0.774	0.772	0.777	0.469

Hình 3.46: Kết quả Test and Score của Naive Bayes

Theo lý lý thuyết, chỉ số AUC càng lớn (càng tiến về 1) thì mô hình càng tốt. Các chỉ số khác của phương pháp Naive Bayes tương đối cao, trong đó AUC cao nhất (0.815). Có thể xem mô hình phân lớp bằng Naive Bayes ở bộ dữ liệu này tương đối tốt.

❖ **Đánh giá dựa trên kết quả Confusion Matrix:**



Confusion Matrix - Orange

Learners: Naive Bayes

Show: Proportion of predicted

Predicted

Actual	Canceled	Not_Canceled	Σ
Canceled	66.1 %	17.7 %	7230
Not_Canceled	33.9 %	82.3 %	15835
Σ	6491	16574	23065

Output: Predictions, Probabilities

Apply Automatically

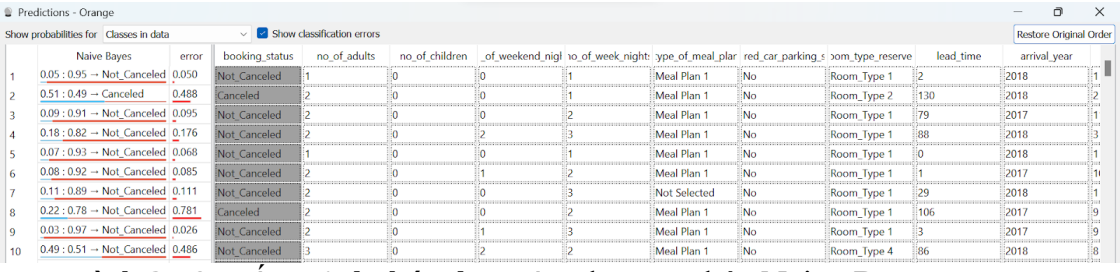
Select Correct | Select Misclassified | Clear Selection

1×23065 | 23.1k

Hình 3.47: Kết quả Confusion Matrix của Naive Bayes

- Dự báo phòng bị huỷ và thực tế phòng bị hủy: 66.1%
- Dự báo phòng không bị hủy nhưng thực tế phòng bị hủy: 17.7%
- Dự báo phòng bị hủy nhưng thực tết phòng không bị hủy: 33.9%
- Dự báo phòng không bị hủy và thực tế phòng không bị hủy: 82.3%

❖ **Kết quả dự báo bằng Naive Bayes:**



Predictions - Orange

Show probabilities for Classes in data | Show classification errors | Restore Original Order

	Naive Bayes	error	booking_status	no_of_adults	no_of_children	_of_weekend_nigl	o_of_week_night	ype_of_meal_plar	red_car_parking_s	oom_type_reserve	lead_time	arrival_year
1	0.05 : 0.95 → Not_Canceled	0.050	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	2	2018
2	0.51 : 0.49 → Canceled	0.488	Canceled	2	0	0	1	Meal Plan 1	No	Room_Type 2	130	2018
3	0.09 : 0.91 → Not_Canceled	0.095	Not_Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	79	2017
4	0.18 : 0.82 → Not_Canceled	0.176	Not_Canceled	2	0	2	3	Meal Plan 1	No	Room_Type 1	88	2018
5	0.07 : 0.93 → Not_Canceled	0.068	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	0	2018
6	0.08 : 0.92 → Not_Canceled	0.085	Not_Canceled	2	0	1	2	Meal Plan 1	No	Room_Type 1	1	2017
7	0.11 : 0.89 → Not_Canceled	0.111	Not_Canceled	2	0	0	3	Not Selected	No	Room_Type 1	29	2018
8	0.22 : 0.78 → Not_Canceled	0.781	Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	106	2017
9	0.03 : 0.97 → Not_Canceled	0.026	Not_Canceled	2	0	1	3	Meal Plan 1	No	Room_Type 1	3	2017
10	0.49 : 0.51 → Not_Canceled	0.486	Not_Canceled	3	0	2	2	Meal Plan 1	No	Room_Type 4	86	2018

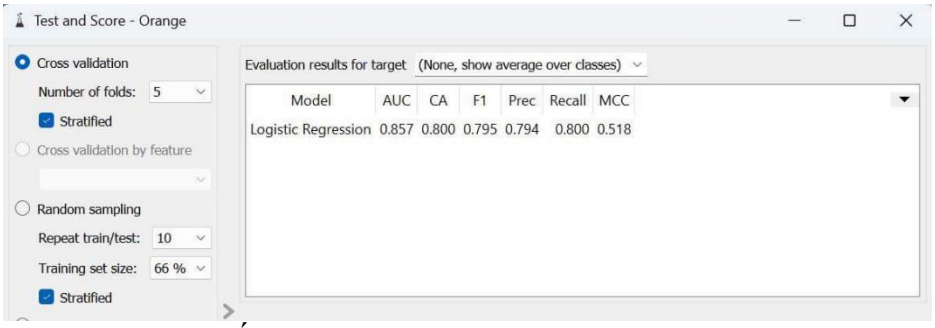
Hình 3.48: Kết quả dự báo dựa trên phương pháp Naive Bayes

❖ **Đánh giá phương pháp:**

Kỹ thuật Naive Bayes phân lớp cho bộ dữ liệu này với chỉ số AUC là 0.815 tương đối khá và các chỉ số còn lại cũng ở mức tạm. Song tỷ lệ sai lầm loại 1 và loại 2 là 17.7% và 33.9% , tổng hai sai lầm này không quá cao. Với các chỉ số chưa cao đồng đều nhau và tỷ lệ sai lầm tạm ổn tuy nhiên phương pháp này chưa thực sự phù hợp để mang lại độ chính xác cao nhất trong việc dự báo tình trạng đặt phòng. Cần xem xét điều chỉnh mô hình để có thể đưa ra được những dự báo tốt hơn giúp cải thiện hoạt động kinh doanh của khách sạn.

f) Logistic Regression

❖ **Đánh giá dựa trên kết quả Test and Score:**



Test and Score - Orange

Cross validation
Number of folds: 5
Stratified
Cross validation by feature
Random sampling
Repeat train/test: 10
Training set size: 66 %
Stratified

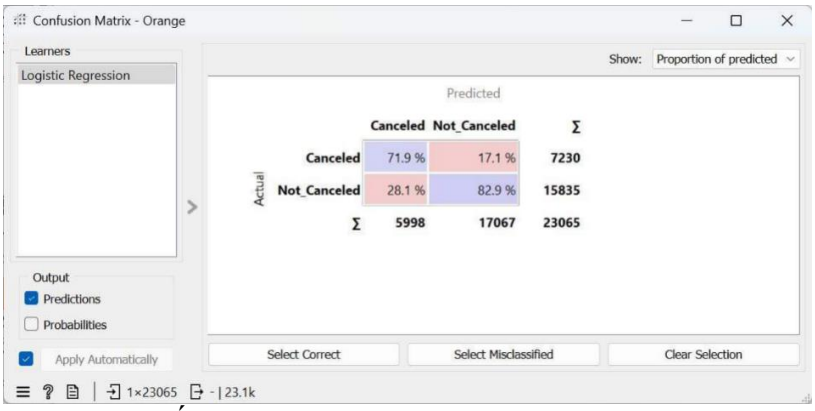
Evaluation results for target (None, show average over classes)

Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression	0.857	0.800	0.795	0.794	0.800	0.518

Hình 3.49: Kết quả Test and Score của Logistic Regression

Chỉ số AUC (Area Under the ROC Curve) là một phương pháp đánh giá hiệu suất của mô hình hồi quy logistic (hay bất kỳ mô hình phân loại nào khác). Trên lý thuyết nếu chỉ số AUC càng gần 1 thì mô hình càng tốt . Ở đây chỉ số AUC là 0,857 được xem là tốt cho mô hình Hồi quy Logit khi áp dụng cho bộ dữ liệu.

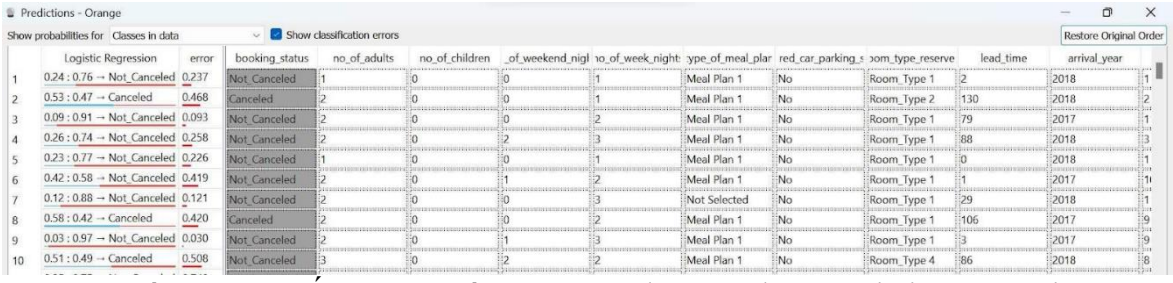
❖ **Đánh giá dựa trên kết quả Confusion Matrix:**



Hình 3.50: Kết quả Confusion Matrix của Logistic Regression

- Dự báo phòng bị huỷ và thực tế phòng bị hủy: 71.9%
- Dự báo phòng không bị hủy nhưng thực tế phòng bị hủy: 17.1%
- Dự báo phòng bị hủy nhưng thực tết phòng không bị hủy: 28.1%
- Dự báo phòng không bị hủy và thực tế phòng không bị hủy: 82.9%

❖ **Kết quả dự báo bằng Logistic Regression:**



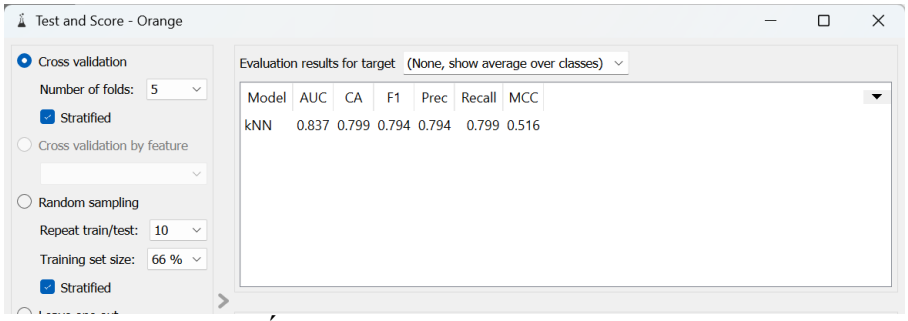
Hình 3.51: Kết quả dự báo dựa trên phương pháp Logistic Regression

❖ **Đánh giá phương pháp:**

Đối với mô hình Logistic Regression, các chỉ số đánh giá tương đối cao, trong đó có chỉ số AUC là 0,857. Có thể nhận xét mô hình Logistic Regression cho bộ dữ liệu này khá tốt. Bên cạnh đó mô hình có phần trăm sai lầm loại 1 và 2 tương đối thấp (17,1% và 28,1%). Với kết quả trên, có thể thấy rằng việc áp dụng phương pháp Hồi quy Logit vào bài toán dự báo này là phù hợp với mức độ tin cậy cao về chỉ số Test and Score và Confusion Matrix, từ đó đưa ra được những dự báo tốt, giúp cải thiện hoạt động kinh doanh của khách sạn.

g) kNN

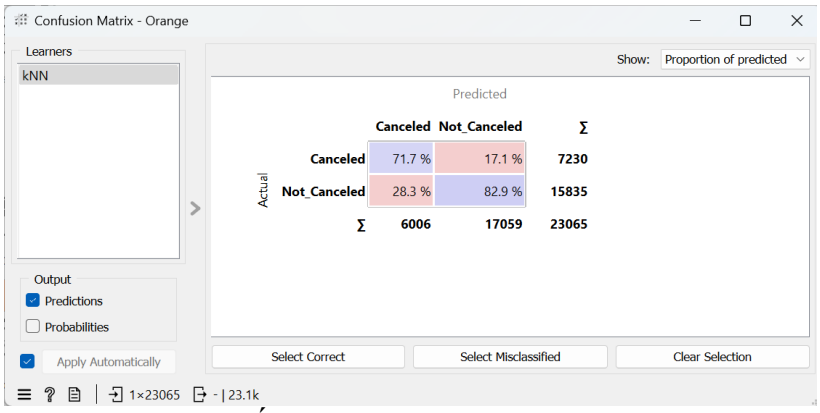
❖ **Đánh giá dựa trên kết quả Test and Score:**



Hình 3.52: Kết quả Test and Score của kNN

Theo lý lý thuyết, chỉ số AUC càng lớn (càng tiến về 1) thì mô hình càng tốt. Các chỉ số khác của phương pháp kNN tương đối cao, trong đó AUC cao nhất (0.837). Có thể xem mô hình phân lớp bằng kNN ở bộ dữ liệu này tương đối tốt.

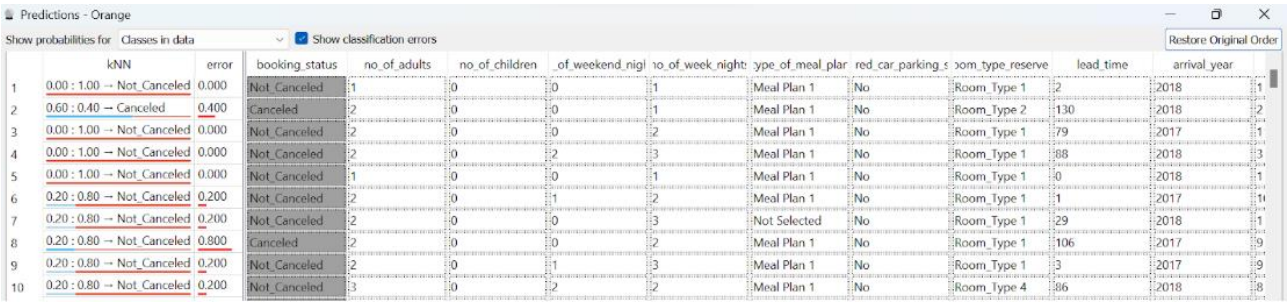
❖ **Đánh giá dựa trên kết quả Confusion Matrix:**



Hình 3.53: Kết quả Confusion Matrix của kNN

- Dự báo phòng bị huỷ và thực tế phòng bị hủy: 71.7%
- Dự báo phòng không bị hủy nhưng thực tế phòng bị hủy: 17.1%
- Dự báo phòng bị hủy nhưng thực tết phòng không bị hủy: 28.3%
- Dự báo phòng không bị hủy và thực tế phòng không bị hủy: 82.9%

❖ **Kết quả dự báo bằng kNN:**



	kNN	error	booking_status	no_of_adults	no_of_children	_of_weekend_nigl	no_of_week_night	ype_of_meal_plar	red_car_parking_s	oom_type_reserve	lead_time	arrival_year	
1	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	2	2018	1
2	0.60 : 0.40 → Canceled	0.400	Canceled	2	0	0	1	Meal Plan 1	No	Room_Type 2	130	2018	2
3	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	79	2017	1
4	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	2	0	2	3	Meal Plan 1	No	Room_Type 1	88	2018	3
5	0.00 : 1.00 → Not_Canceled	0.000	Not_Canceled	1	0	0	1	Meal Plan 1	No	Room_Type 1	0	2018	1
6	0.20 : 0.80 → Not_Canceled	0.200	Not_Canceled	2	0	1	2	Meal Plan 1	No	Room_Type 1	1	2017	1
7	0.20 : 0.80 → Not_Canceled	0.200	Not_Canceled	2	0	0	3	Not Selected	No	Room_Type 1	29	2018	1
8	0.20 : 0.80 → Not_Canceled	0.800	Canceled	2	0	0	2	Meal Plan 1	No	Room_Type 1	106	2017	9
9	0.20 : 0.80 → Not_Canceled	0.200	Not_Canceled	2	0	1	3	Meal Plan 1	No	Room_Type 1	3	2017	9
10	0.20 : 0.80 → Not_Canceled	0.200	Not_Canceled	3	0	2	2	Meal Plan 1	No	Room_Type 4	86	2018	8

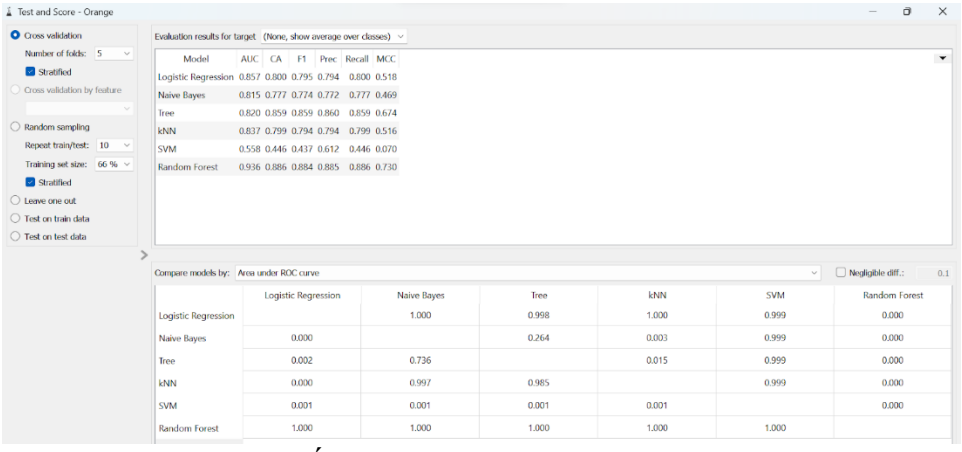
Hình 3.54: Kết quả dự báo dựa trên phương pháp kNN

❖ **Đánh giá phương pháp:**

Đối với phương pháp kNN, các chỉ số khác đều khá cao, trong đó chỉ số AUC là 0.837, điều này chứng tỏ mô hình phân loại sử dụng kNN trên bộ dữ liệu này là một lựa chọn tốt. Điểm đáng chú ý là tỷ lệ sai sót loại 1 và loại 2 đều ở mức tương đối thấp (lần lượt là 17.1% và 28.3%). Với kết quả khả quan này cho thấy việc áp dụng phương pháp kNN vào bài toán dự báo này là phù hợp, cùng với mức độ tin cậy của các chỉ số Test and Score và Confusion Matrix, cung cấp những dự báo chính xác và có giá trị, từ đó đóng góp vào việc tối ưu hoá hoạt động kinh doanh của khách sạn.

3.7.2. Kết quả và đánh giá chung

❖ **Đánh giá dựa trên kết quả Test and Score:**



Hình 3.55: Kết quả Test and Score của mô hình

Tại mục Evaluation results, ta cần chú ý kết quả định lượng của 6 mô hình SVM, Tree, Logistic Regression, Naive Bayes, Random Forest, kNN.

Qua đó ta có thể thấy: AUC: Random Forest = 0.936 > Logistic Regression = 0.857 > kNN = 0.837 > Tree = 0.820 > Naive Bayes = 0.815 > SVM = 0.558 (Giá trị này càng lớn thì mô hình càng tốt). Bên cạnh đó, phương pháp Random Forest có các chỉ số Accuracy, F1-Score, Precision và Recall là cao nhất trong cả 6 mô hình sử dụng.

➔ Nên sử dụng phương pháp Random Forest.

❖ **Đánh giá dựa trên kết quả Confusion Matrix:**

Hình 3.56: Kết quả Confusion Matrix của mô hình

Sai lầm loại 1: Dự báo rằng phòng không bị hủy nhưng thực tế thì phòng bị hủy.

Sai lầm loại 2: Dự báo rằng phòng hủy nhưng thực tế thì phòng không bị hủy.

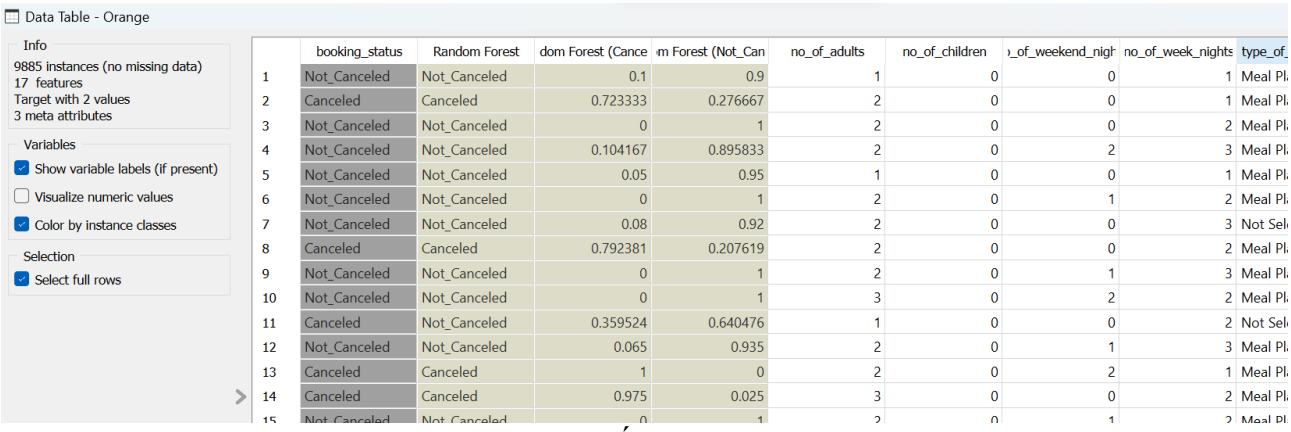
Các sai lầm này ảnh hưởng đến sự chuẩn bị, lên kế hoạch của khách sạn, đồng thời còn ảnh hưởng đến sự chuyên nghiệp của khách sạn. Và điều này dẫn tới những ảnh hưởng không nhỏ đối với doanh thu của khách sạn. Vì vậy, chúng ta sẽ đi tìm các dự đoán này ở mức thấp nhất.

Random Forest = 24.7% (10.2% + 14.5%) < Tree = 33.1% < Logistic Regression = 45.2% < kNN = 45.4% < Navie Bayes = 51.6% < SVM = 92.7%

➔ Nên sử dụng phương pháp Random Forest

➔ Dựa theo 2 kết quả của Test and score và Confusion Matrix, chúng ta sẽ chọn phương pháp Random Forest.

❖ **Kết quả dự báo:**



	booking_status	Random Forest	dom Forest (Cance	m Forest (Not_Can	no_of_adults	no_of_children	_of_weekend_nigh	no_of_week_nights	type_of_
1	Not_Canceled	Not_Canceled	0.1	0.9	1	0	0	1	Meal Pl
2	Canceled	Canceled	0.723333	0.276667	2	0	0	1	Meal Pl
3	Not_Canceled	Not_Canceled	0	1	2	0	0	2	Meal Pl
4	Not_Canceled	Not_Canceled	0.104167	0.895833	2	0	2	3	Meal Pl
5	Not_Canceled	Not_Canceled	0.05	0.95	1	0	0	1	Meal Pl
6	Not_Canceled	Not_Canceled	0	1	2	0	1	2	Meal Pl
7	Not_Canceled	Not_Canceled	0.08	0.92	2	0	0	3	Not Sel
8	Canceled	Canceled	0.792381	0.207619	2	0	0	2	Meal Pl
9	Not_Canceled	Not_Canceled	0	1	2	0	1	3	Meal Pl
10	Not_Canceled	Not_Canceled	0	1	3	0	2	2	Meal Pl
11	Canceled	Not_Canceled	0.359524	0.640476	1	0	0	2	Not Sel
12	Not_Canceled	Not_Canceled	0.065	0.935	2	0	1	3	Meal Pl
13	Canceled	Canceled	1	0	2	0	2	1	Meal Pl
14	Canceled	Canceled	0.975	0.025	3	0	0	2	Meal Pl
15	Not_Canceled	Not_Canceled	0	1	2	0	1	2	Meal Pl

Hình 3.57: Kết quả dự báo

❖ **Đánh giá kết quả bài toán:**

Theo bảng đánh giá kết quả, mô hình Random Forest cho ra kết quả cao nhất trong 6 mô hình: AUC (94%), Accuracy (89%), F-I score, Precision và Recall đều 88%-89% đều là những tỷ lệ rất cao. Bên cạnh đó Confusion Matrix của Random Forest có sai lầm loại 2 bằng 14,9% và sai lầm loại 1 là 10% thấp nhất trong 6 mô hình được áp dụng cho bài toán.

CHƯƠNG IV: KẾT LUẬN

4.1. Kết luận

Trong bối cảnh sự phát triển vượt bậc của ngành du lịch hiện nay, việc đặt phòng trực tuyến đã trở thành một phần không thể thiếu của trải nghiệm du lịch. Sự tiện lợi và nhanh chóng của việc này không chỉ giúp khách hàng tiết kiệm thời gian và chi phí di chuyển đến khách sạn, mà còn mở ra một thế giới của sự lựa chọn và tiện ích.

Tuy nhiên, cùng với sự tiến triển của công nghệ đến từ việc đặt phòng trực tuyến là sự gia tăng của tình trạng hủy phòng qua các trang mạng của các doanh nghiệp khách sạn. Sự phổ biến của hiện tượng này không chỉ gây ra phiền toái mà còn đặt ra thách thức lớn cho việc quản lý khách sạn. Việc xử lý các trường hợp hủy phòng đột ngột không chỉ tốn kém thời gian và công sức mà còn có thể ảnh hưởng đến uy tín và hình ảnh của khách sạn.

Cuối cùng, nghiên cứu này đã thực hiện được những mục tiêu đề ra là giải quyết các bài toán liên quan đến vấn đề được đặt ra ban đầu, các doanh nghiệp khách sạn cần thiết lập chính sách hủy phòng rõ ràng và minh bạch, cung cấp thông tin chi tiết và dễ hiểu về các điều khoản và điều kiện hủy phòng và hoàn tiền. Bằng cách này, họ có thể giảm thiểu sự bất tiện và lo lắng cho khách hàng và đồng thời tăng cường niềm tin và sự hài lòng của họ.

Hơn nữa, mô hình cũng được xây dựng khá hoàn chỉnh để xác định tính chính xác của vấn đề, từ đó hỗ trợ quá trình quản lý khách sạn: điển hình là việc có thể cải thiện quy trình đặt phòng và tăng cường chất lượng dịch vụ khách hàng, đây là cách hiệu quả để giảm thiểu tình trạng hủy phòng và tăng khả năng hoàn thành mục tiêu đặt phòng trong tương lai. Bằng cách này, các doanh nghiệp khách sạn không chỉ có thể thu hút được nhiều khách hàng hơn mà còn có thể duy trì mối quan hệ lâu dài và bền vững với họ.

4.2. Hạn chế

Trong quá trình làm bài, nhóm tác giả không thể tránh khỏi một số hạn chế do các yếu tố chủ quan, cũng như khách quan như sau:

- Dữ liệu chưa được kiểm nghiệm về độ chính xác: Tính chính xác của dữ liệu chưa được xác thực hoặc kiểm định, có thể gây ra sai lệch trong kết quả phân tích.
- Tại Việt Nam, dữ liệu về khách hàng tại các dữ liệu khách sạn là một dạng bảo mật thông tin, vì vậy, sinh viên chưa thực sự có cơ hội làm việc với dữ liệu trong nước và dữ liệu thực tế.
- Thời gian học phần ngắn, hạn chế về thời gian hoàn thành và sinh viên vẫn còn nhiều thiếu sót trong thực hành các bước huấn luyện dữ liệu cũng như làm việc với dữ liệu.

4.3. Giải pháp

Với những hạn chế gặp phải như trên, nhóm xin đề xuất một số hướng phát triển tiếp theo của đề tài:

- Tăng cường thu thập dữ liệu: Khách sạn nên tập trung vào việc thu thập dữ liệu từ nhiều nguồn khác nhau, không chỉ là dữ liệu từ các kênh trực tuyến mà còn từ các

kênh offline. Điều này sẽ cung cấp một cơ sở dữ liệu đa dạng và phong phú hơn, giúp cải thiện tính chính xác của mô hình dự đoán.

- Nghiên cứu dài hạn: Cần tiến hành nghiên cứu qua nhiều năm và nhiều thời kỳ để hiểu rõ hơn về sự thay đổi của các chỉ số về đặt phòng và xu hướng của khách hàng. Quá trình này đòi hỏi sự kiên nhẫn và phải được thực hiện với sự hỗ trợ từ các khách sạn.
- Cập nhật và theo dõi mô hình thường xuyên: Cần thường xuyên cập nhật và kiểm tra mô hình dự đoán để đảm bảo rằng nó vẫn giữ được tính chính xác và có thể áp dụng trong thực tế. Sự liên tục trong việc điều chỉnh và cập nhật mô hình là cần thiết để đảm bảo tính hiệu quả và khả năng áp dụng trong tương lai.

TÀI LIỆU THAM KHẢO

[1]. *Mô hình cây quyết định (decision tree) — Deep AI KhanhBlog*. (n.d.).

[2]. *Bài giảng Máy học nâng cao - Chương 5: Naive Bayes Classification - Trịnh Tấn Đạt – Tài liệu, ebook, giáo trình, hướng dẫn*. (n.d.).

[3]. Bài Học 10 Phút. (2016, October 30). *Mô hình hồi quy logistic (hồi quy logit)* [Video]. YouTube.

[4]. *Data Mining là gì? Các công cụ khai phá dữ liệu phổ biến nhất hiện nay*. (n.d.).

[5]. Dvms.Vn. (n.d.). *THUẬT TOÁN KNN VÀ VÍ DỤ ĐƠN GIẢN TRONG NGÀNH NGÂN HÀNG*.

[6]. GfG. (2023, June 10). *Support Vector Machine (SVM) algorithm*. GeeksforGeeks.

[7]. Hop, N. T. (2024, March 21). *KNN (K-Nearest Neighbors) #1*. Viblo.

[8]. *Khái niệm về phương pháp random forest trong cuộc cách mạng machine learning và định hướng ứng dụng trong lĩnh vực viễn thám - Luận văn, đồ án, luan van, do an*. (n.d.).

[9]. Nguyễn Văn Tuấn. (2014, November 15). *Bài giảng 43: Mô hình hồi qui logistic* [Video]. YouTube.

[10]. Orange Data Mining. (2023a, September 15). *Logistic regression* [Video]. YouTube.

[11]. Orange Data Mining. (2023b, October 16). *Logistic Regression Nomogram* [Video]. YouTube.

[12]. Phạm M. (2020, May 11). Thuật toán K láng giềng gần nhất (K-Nearest Neighbor - KNN) là gì? *Vietnambiz*.

[13]. *Random Forest algorithm — Machine Learning cho dữ liệu dạng bảng*. (n.d.).

[14]. Saini, A. (2024, January 5). *Decision Tree – a Step-by-Step guide*. Analytics Vidhya.

[15]. Sim, N. D. (2024, March 21). *# Phân lớp bằng Random Forests trong Python*. Viblo.

[16]. ThongKe.Club. (2022, September 6). Ứng dụng của thuật toán rừng ngẫu nhiên - Random Forests - Phân tích xử lý dữ liệu. *Phân Tích Xử Lý Dữ Liệu*.

[16]. Toàn, P. V. (2024, March 21). *Support Vector Machine trong học máy - Một cái nhìn đơn giản hơn*. Viblo.

[17]. Trituenhantao.Io. (2024, February 11). *Cây quyết định (Decision tree)*. Trí Tuệ Nhân Tạo.

[18]. Trung, H. C. (2024, March 21). *Giới thiệu về Support Vector Machine (SVM)*. Viblo.

[19]. Vu, T. (2017, August 8). *Bài 32: Naive Bayes classifier*. Tiep Vu's Blog.

[20]. Vu, T. (2018, January 14). *Bài 34: Decision Trees (1): Iterative Dichotomiser 3*. Tiep Vu's Blog.