Khoa Học Dữ Liệu

Tiểu luận cuối kỳ

Predicting Movie Ratings

Nhóm 5

Nguyễn Phan Nhật Hoàng

Lê Văn Thịnh

Phan Khánh Ngân

Bảng phân công nhiệm vụ

Thành viên	Công việc
Nguyễn Phan Nhật Hoàng	- Thu thập dữ liệu - Mô tả dữ liệu
Lê Văn Thịnh	- Xử lý dữ liệu • Xử lý dữ liệu trống. • Biến đổi phân loại dữ liệu. • Thêm các đặc trưng mới. • Xử lý ngoại lệ. • Chuẩn hóa dữ liệu. - Trực quan hóa dữ liệu sau khi xử lý dữ liệu.
Phan Khánh Ngân	- Thực hiện lựa chọn đặc trưng bằng thuật toán RFE. - Xây dựng hai mô hình hồi quy (RF và SVM). - Triển khai thuật toán lựa chọn tham số tối ưu. - Đánh giá, so sánh hai mô hình hồi quy (RF và SVM).

Giới thiệu

1. Mục tiêu

Xây dựng hệ thống dự đoán IMDb Rating của các bộ phim trước khi ra mắt.

- Đầu vào chính là các pre-released features của phim (genre, directors, writers, stars, runtime, votes, released, ratings)
- Đầu ra chính là IMDb rating dự báo

2. Giải pháp tổng quan

Nghiên cứu, xử lý các dữ liệu theo các phương pháp phù hợp, ứng dụng các mô hình học máy vào quá trình huấn luyện. Từ đó có thể đưa ra các mô hình có độ tin cậy cao áp dụng vào hệ thống dự đoán IMDb rating của phim, hỗ trợ các nhà làm phim trong việc đưa ra quyết định có nên hay không sản xuất một bộ phim.

Thu thập và mô tả dữ liệu

1. Thu thập dữ liệu
2. Mô tả dữ liệu

1. Thu thập dữ liệu

1.1. Nguồn dữ liệu

IMDb là trang web hàng đầu về thông tin phim và ngành công nghiệp điện ảnh. Nó cung cấp một cơ sở dữ liệu phong phú về các bộ phim, diễn viên và đạo diễn. IMDb cũng cho phép người dùng đánh giá và thảo luận với cộng đồng yêu thích điện ảnh.

1.2. Công cụ thu thập

BeautifulSoup là một thư viện Python mạnh mẽ được sử dụng để phân tích và trích xuất dữ liệu từ các trang web. Nó cung cấp các công cụ linh hoạt để truy cập và xử lý các thành phần HTML và XML trong một cách dễ dàng. Với BeautifulSoup, chúng ta có thể lấy thông tin từ các thẻ HTML, thuộc tính, văn bản và cấu trúc của một trang web. Thư viện này giúp đơn giản hóa quá trình web scraping và phân tích dữ liệu từ trang web.

1. Thu thập dữ liệu

1.3. Các thức sử dụng công cụ

- Sử dụng thư viện requests để gửi yêu cầu GET đến trang web IMDb.
- Sử dụng thư viện BeautifulSoup để phân tích nội dung HTML trả về từ yêu cầu.
- Xác định các phần tử trên trang web chứa thông tin cần thu thập (ví dụ: tiêu đề phim, năm phát hành, đánh giá IMDb, số phiếu bầu, thể loại, thời lượng, v.v.).
- Trích xuất thông tin từ các phần tử và lưu trữ vào các biến hoặc mảng.
- Lặp lại quá trình trên các trang khác nhau để thu thập thông tin từ nhiều trang.

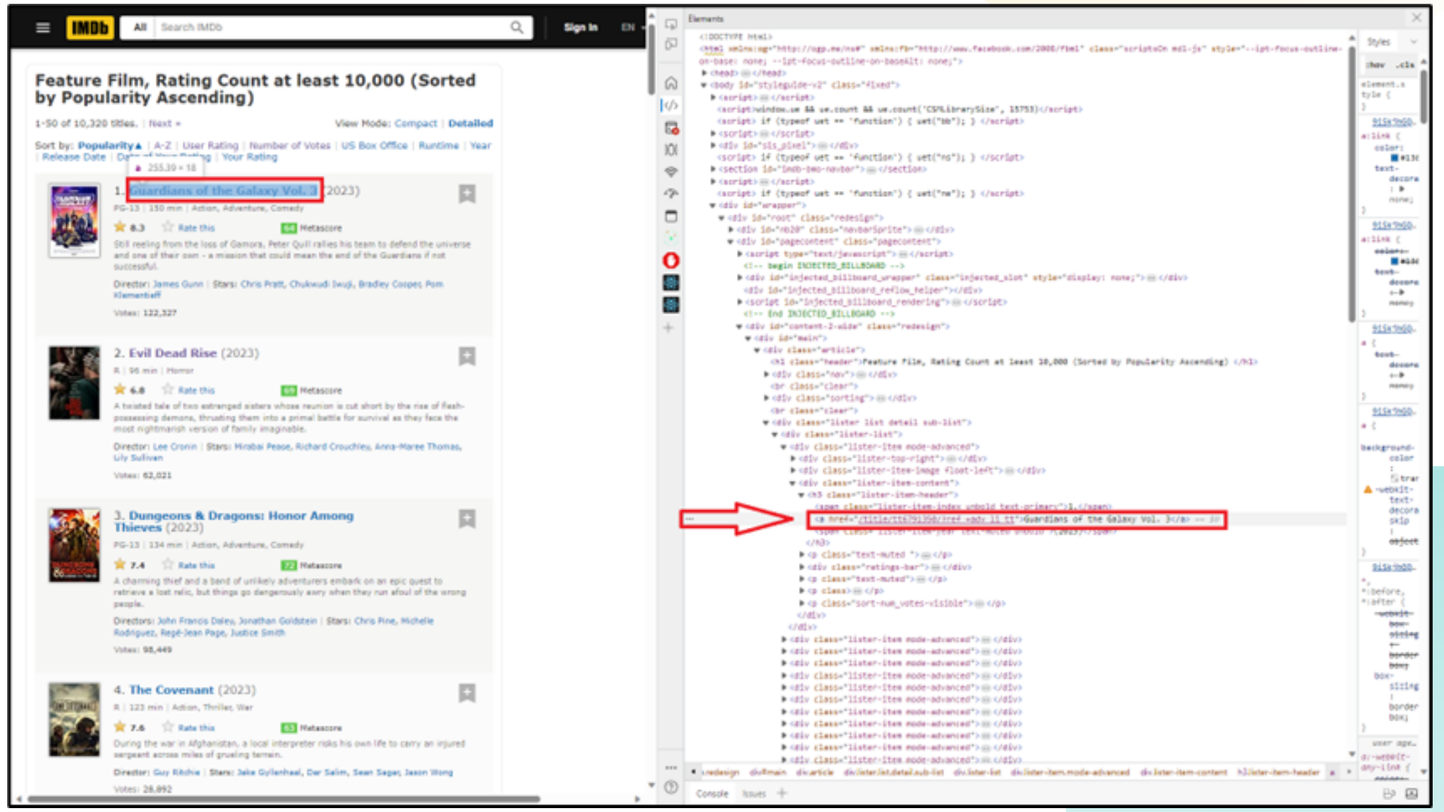
1.4. Đầu ra và đầu vào của quá trình thu thập

- Đầu vào: URL của trang web IMDb, các tham số truy vấn để tìm kiếm, lọc và sắp xếp danh sách phim theo yêu cầu.
- Đầu ra: Dữ liệu thu thập được từ trang web IMDb, bao gồm thông tin về các bộ phim như title, releaded, imdb score, genre, runtime, votes, rating, directors, stars, v.v. Dữ liệu này được lưu trữ trong các cấu trúc dữ liệu như mảng, DataFrame sau đó được lưu vào file (ví dụ: CSV, Excel) để sử dụng và phân tích sau này.

1. Thu thập dữ liệu

1.5. Ví dụ minh họa

- Thực hiện hàm get() của thư viện requests để lấy được nội dung HTML của trang. Tiếp đó, sử dụng hàm BeautifulSoup() để phân tích trang HTML.
- Thực hiện các hàm find(), find_all(), ... của thư viện BeautifulSoup để truy xuất đến element trong DOM. Từ đó, lấy ra nội dung của element(ở hình bên là ta lấy được tên của phim).
- Thực hiện tương tự với các thông tin khác như là năm phát hành, đánh giá IMDb, số phiếu bầu, thể loại, ...
- Dữ liệu được lưu trữ trong một DataFrame và xuất ra file CSV để sử dụng và phân tích



2. Mô tả dữ liệu

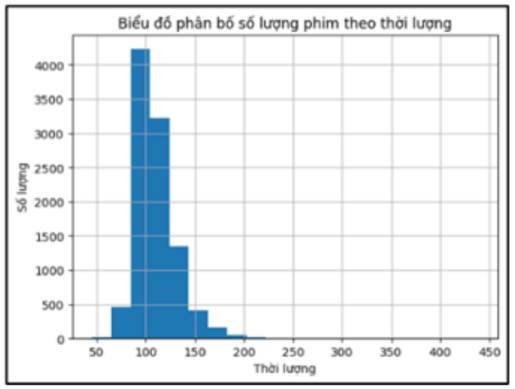
2.1. Thống kê tổng quan về tập dữ liệu

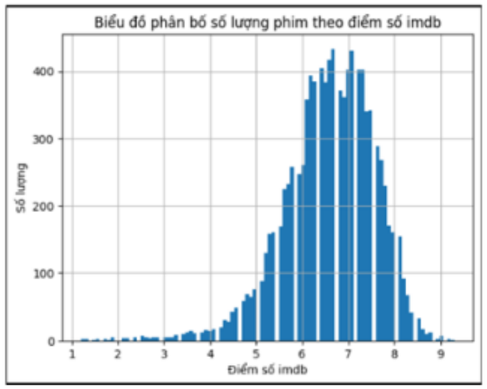
STT	Đặc trưng	Mô tả	Kiểu dữ liệu
1	title	Tên của phim	object
2	year	Năm phim ra mắt	int64
3	genre	Các thể loại của phim	object
4	runtime	Thời lượng của phim	int64
5	imdb	Điểm số IMDb	float64
6	votes	Số lượt bình chọn trên trang IMDb của phim	int64
7	released	Ngày ra mắt của phim	object
8	budget	Kinh phí của phim	float64
9	companies	Các công ty đầu tư	object
10	gross	Doanh thu của phim	float64
11	diretors	Các đạo diễn tham gia	object
12	writers	Các biên kịch tham gia	object
13	stars	Các diễn viên chính tham gia đóng phim	object
14	origins	Những nơi công chiếu phim đầu tiên	object
15	rating	Giới hạn độ tuổi(R, PG-13,…)	object

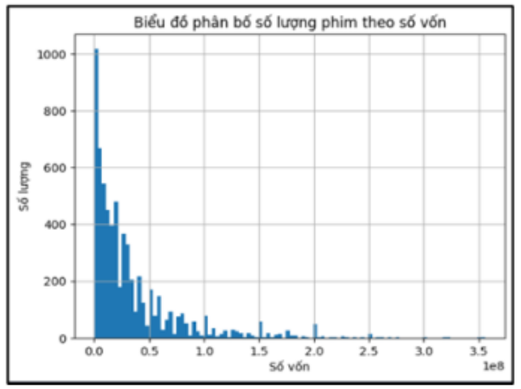
	BIG DS	SMALL DS
Tổng dữ liệu (records)	10000	1000
Dữ liệu trống trong đặc trưng “rating”	277	4
Dữ liệu trống trong đặc trưng “budgets”	3460	128
Dữ liệu trống trong đặc trưng “gross”	1225	39

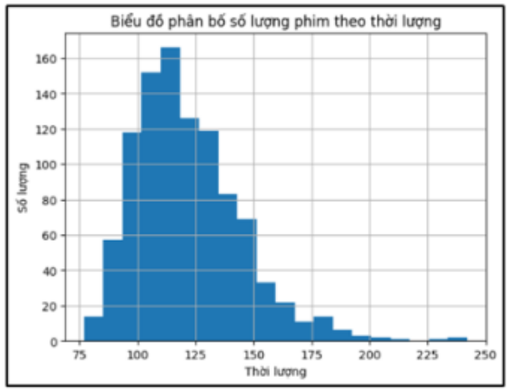
2. Mô tả dữ liệu

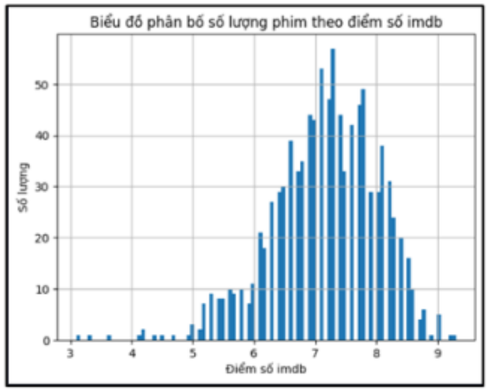
2.2. Mô tả trực quan một số đặc trưng đáng chú ý

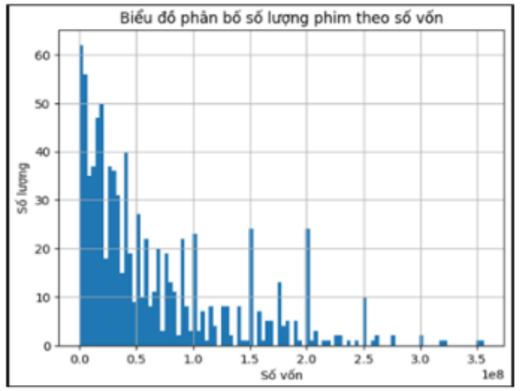


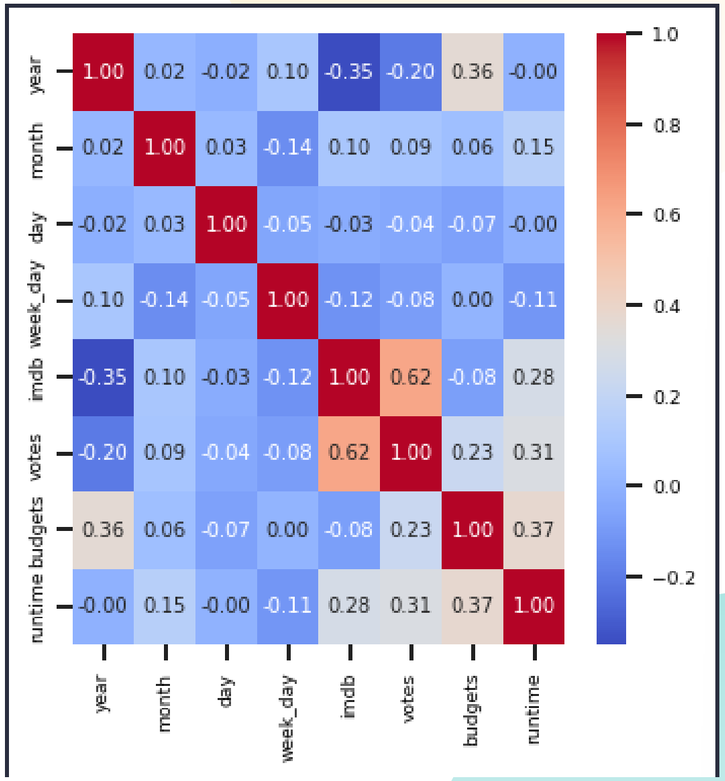












Nhận xét:

- Thời lượng phim từ 80 – 140 phút có số lượng cao. Điều này liên quan đến thói quen khi đi xem phim của mọi người, các phim quá dài làm cho người xem chán nản, phim quá ngắn thì không đủ xây dựng nhân vật và tình tiết
- Điểm IMDB có phổ điểm rộng 1 – 9, nhưng tập rất nhiều ở mức điểm 6 – 8. Đây là mức điểm khá tốt đối với một bộ phim.
- Kinh phí bỏ ra cho một bộ phim cao nhất gần 350.000.000 dollar, đa số thì bé hơn 50.000.000 dollar.
- Ngoài "votes" có độ tương quan với IMDB rating là 0.62 ra thì các đặc trưng còn lại có độ tương quan khá thấp.

Trích xuất đặc trưng

1. Làm sạch dữ liệu.
2. Xử lý ngoại lệ
3. Chuẩn hóa dữ liệu
4. Lựa chọn đặc trưng

1. Làm sạch dữ liệu

1.1. Xử lý dữ liệu trống

Tập dữ liệu gồm các trường chứa dữ liệu trống đó là budgets, rating và gross. Ta sẽ xóa các dòng chứa giá trị trống đi để đảm bảo được tính chính xác của dữ liệu.

1.2. Xử lý rating, origins

Có rất nhiều rating và origins đã được thu thập trong dataset.
Trong bài toán này ta sẽ giữ lại 5 rating phổ biến nhất là R, PG-13, PG, G, NC-17. Ta tiến hành thay thế tất cả các rating trong dataset bằng 5 rating phổ biến như sau:

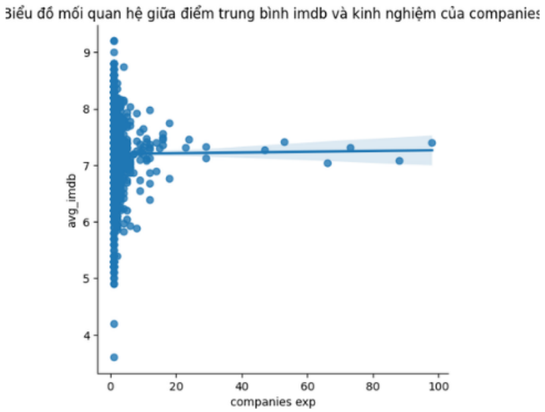
- 13+ = TV-14 = **PG-13**, 16+ = C16 = TV-MA = Not Rated = Passed = Unrated = **R**.
- M = M/PG = GP = TV-PG = **PG**, X = **NC-17**, Approved = TV-G = **G**.

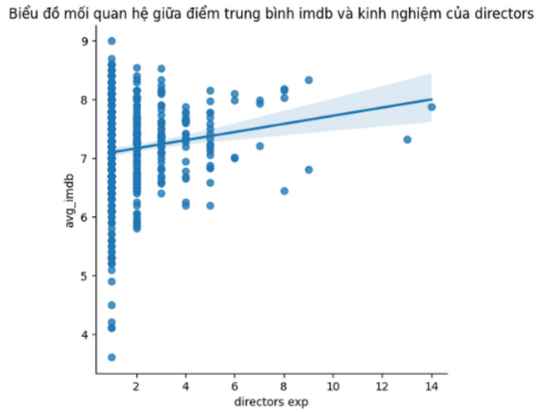
Orgins của các bộ phim chiếm đa số ở United States và United Kingdom, còn lại sẽ nằm rải rác ở các quốc gia khác. Do đó ta sẽ chia lại lớp cho trường Orgins như sau: US (United State), UK (United Kingdom), USUK (phim có origin gồm cả United State và United Kingdom), Others.

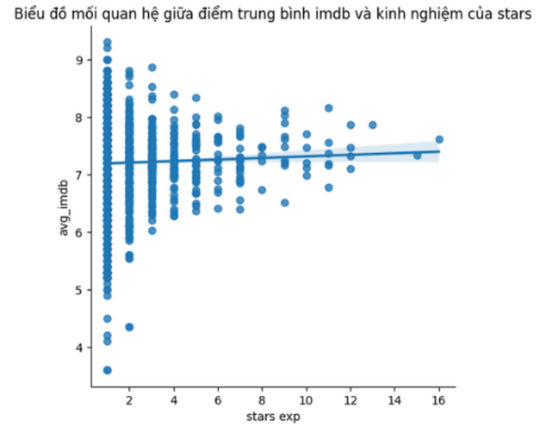
1. Làm sạch dữ liệu

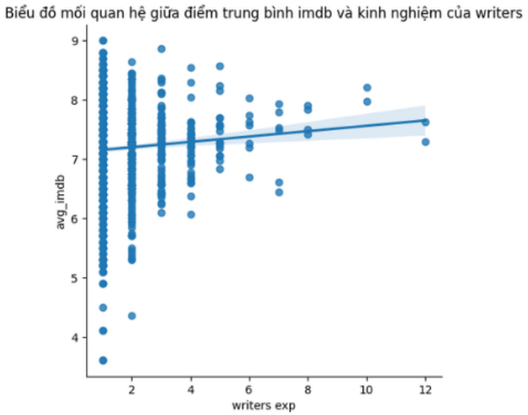
1.3. Xử lý companies, stars, writers, directors

Ta tiến hành tạo thêm 4 đặc trưng mới dựa trên 4 đặc trưng cũ: companies, stars, writers, directors (gọi tắt là creators). Đặc trưng mới này chính là kinh nghiệm của creators tương ứng - được tính bằng cách thống kê số lượng phim mà creators đó đã tham gia sản xuất trong dataset.









Từ các đồ thị trên, ta kết luận được nếu creators của một bộ phim có càng nhiều kinh nghiệm thì điểm imdb của phim sẽ càng lớn. Điều này cho thấy rằng kinh nghiệm của creators cũng sẽ ảnh hưởng đến điểm imdb của phim.

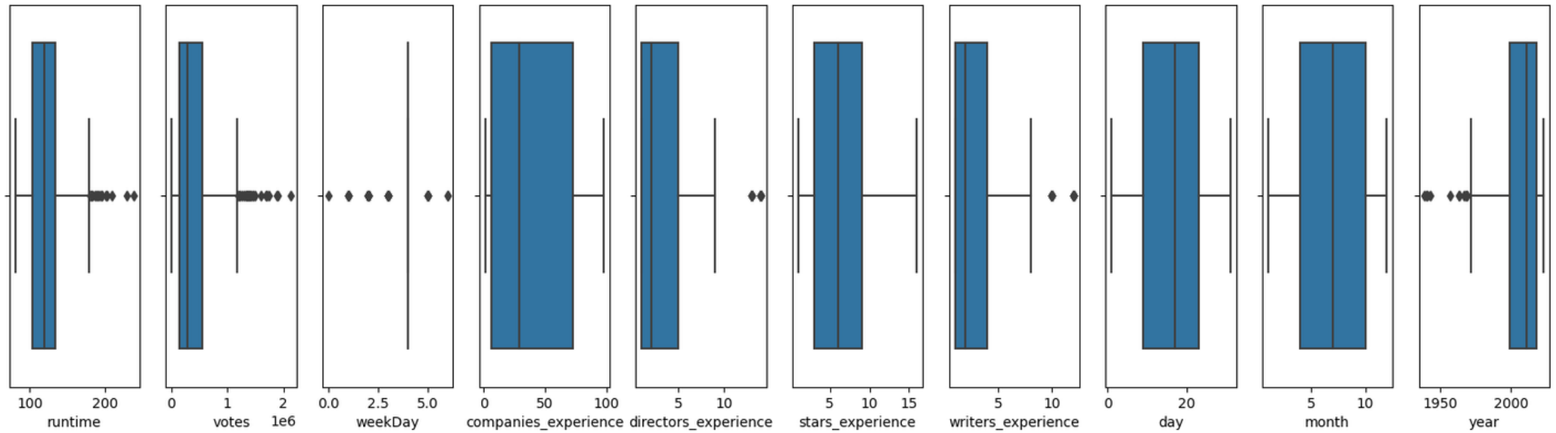
1. Làm sạch dữ liệu

1.4. Xử lý releaseds, genres

Ta sẽ tách releaseds thành 4 cột mới là: day, month, year, weekDay.

Gernes bao gồm nhiều thể loại của một bộ phim. Xử lý đặc trưng này bằng cách biến đổi mỗi danh mục gernes thành một con số riêng biệt, sử dụng Label Encoder.

2. Xử lý ngoại lệ



Từ các đồ thị boxplot trên, ta có thể nhận thấy 2 trường có nhiều ngoại lệ nhất đó chính là runtime và votes, nên ta sẽ tiến hành xử lý ngoại lệ cho 2 trường này.

3. Chuẩn hóa dữ liệu

Để cải thiện độ chính xác, tốc độ thực thi của mô hình, cũng như giúp mô hình học máy dễ dàng hội tụ và tạo kết quả tốt hơn, dữ liệu phải cần được chuẩn hóa. Trong các phương pháp chuẩn hóa dữ liệu, phương pháp Min Max Scaler được sử dụng phổ biến và áp dụng hiệu quả với dữ liệu gốc của bài toán – dữ liệu có phân phối không chuẩn. Phương pháp Min Max Scaler chuyển dữ liệu gốc vào một phạm vi cụ thể như [0, 1] bởi công thức:

$$\mathrm{x_{std}} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

$$\mathrm{x_{scaled}} = \mathrm{x_{std}} * (\max - min) + \min$$

Trong đó: min = 0 và max = 1.

4. Lựa chọn đặc trưng: sử dụng Recursive Feature Elimination (RFE)

RFE hoạt động dựa vào một external estimator (SVR, RFR,...). Estimator này sẽ được huấn luyện trên tập đặc trưng ban đầu, sau đó tầm quan trọng các các đặc trưng sẽ được tính toán để loại bỏ đặc trưng ít quan trọng nhất. Quá trình này sẽ được đệ quy cho đến khi nào tập đặc trưng đưa vào ban dầu đạt đến số lượng mong muốn.

Thư viện scikit-learn cung cấp lớp sklearn.feature_selection.RFE để có thể cài đặt thuật toán này. Một vài tham số sử dụng:

Tham số	Ý nghĩa
estimator	Mô hình học máy có giám sát, cung cấp phương thức “fit” để tính toán “feature importance”.
n_features_to_select	Số lượng đặc trưng cần lựa chọn.
step	Là giá trị lớn hơn 1, biểu thị số lượng đặc trưng muốn loại bỏ sau mỗi lần lặp.

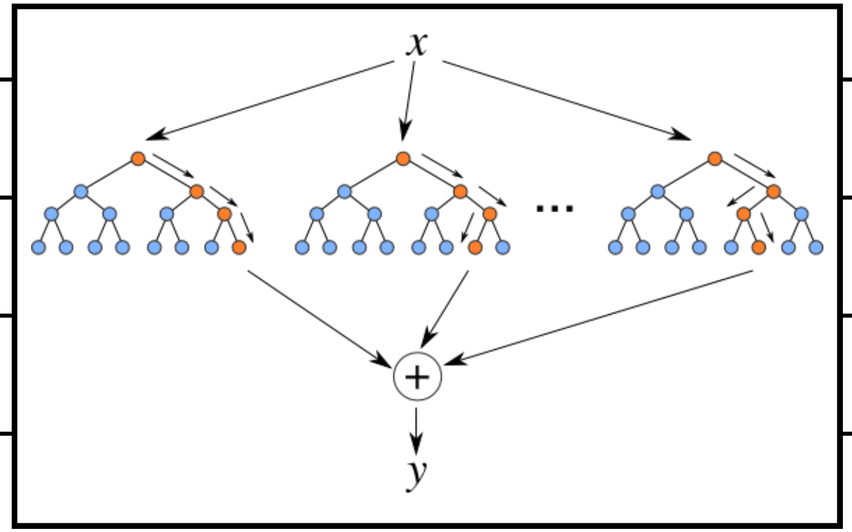
Mô hình hóa dữ liệu

1. Mô hình RandomForests
2. Mô hình Support Vector Machines
3. Kết quả huấn luyện

1. Random Forests (Random Forest Regression)

Random Forest là mô hình sử dụng nhiều cây quyết định phân loại khác nhau trên nhiều tập dữ liệu con khác nhau của tập dữ liệu ban đầu. Mô hình sử dụng phương pháp trung bình để cải thiện độ chính xác dự đoán và kiểm soát hiện tượng over-fitting. Mô hình học máy Random Forest được sử dụng cho cả bài toán phân lớp và bài toán hồi quy.

Sử dụng lớp RandomForestRegression từ module sklearn.ensemble để xây dựng hệ thống dự đoán hồi quy.



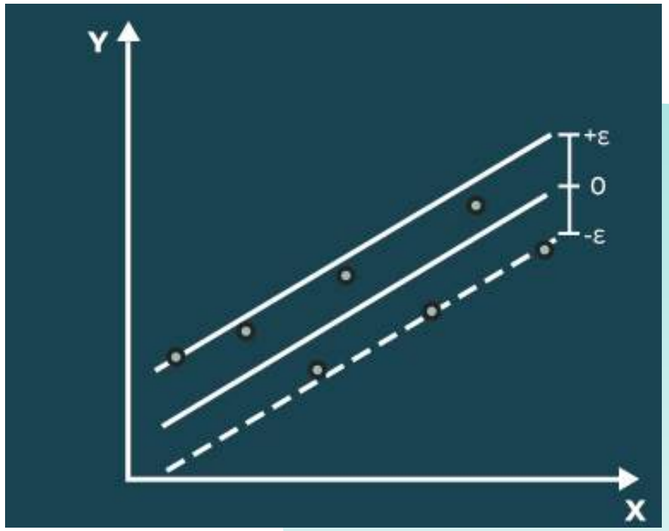
Tham số	Ý nghĩa
n_estimators	Số lượng cây trong mô hình forest.
max_depth	Độ sâu tối đa của cây.
min_samples_split	Số lượng mẫu tối thiểu để chia một nút trong cây.
min_samples_leaf	Số lượng mẫu tối thiểu trong mỗi lá của cây.
max_features	Số lượng đặc trưng được xem xét khi tìm kiếm phân chia tốt nhất.
random_state	Giá trị đảm bảo kết quả được tái tạo như nhau trong quá trình huấn luyện mô hình.

2. Support Vector Machines (Support Vector Regression)

SVMs là một trong những mô hình học máy có giám sát được sử dụng trong các bài toán phân lớp, hồi quy và dự đoán ngoại lệ. Đối với bài toán hồi quy, có thể sử dụng Support Vector Regression (SVR) với ý tưởng tương tự như bài toán phân lớp. Mục tiêu của SVR là tìm ra hàm xấp xỉ mối quan hệ giữa các biến đầu vào với một biến đầu ra mang giá trị liên tục, với lỗi dự đoán tối thiểu.

Thư viện scikit.learn cung cấp lớp sklearn.svm.SVR để có thể xây dựng mô hình dự đoán hồi quy cho bài toán Predicting move ratings. Một vài tham số sử dụng để huấn luyện:

Tham số	Ý nghĩa
kernel	Chỉ định loại kernel để tính toán trước kernel matrix.
C	Tham số tổng quát hóa (giúp mô hình tránh overfitting) với penalty là l2 squared.
epsilon	Tham số xác định phạm vi chấp nhận được cho sai số của các mẫu huấn luyện, quyết định mức độ linh hoạt của đường biên hỗ trợ (support boundary) xung quanh các điểm dữ liệu huấn luyện.

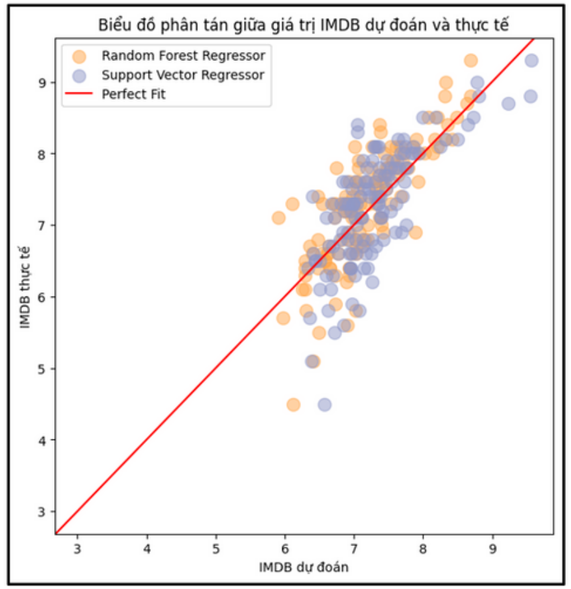


3. Kết quả huấn luyện

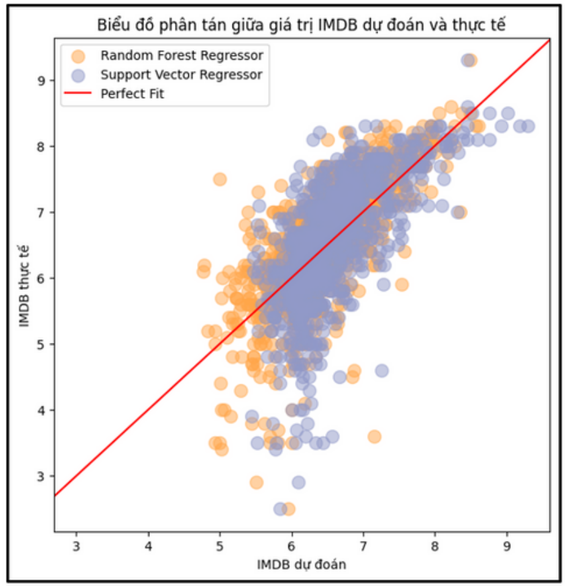
Trước lựa chọn đặc trưng:

Metrics	Small Dataset (1000 mẫu)		Big Dataset (10000 mẫu)	
	RF	SVM	RF	SVM
MAE	0.39	0.42	0.47	0.52
RMSE	0.51	0.55	0.64	0.70
R2	0.61	0.55	0.55	0.45

Small DS



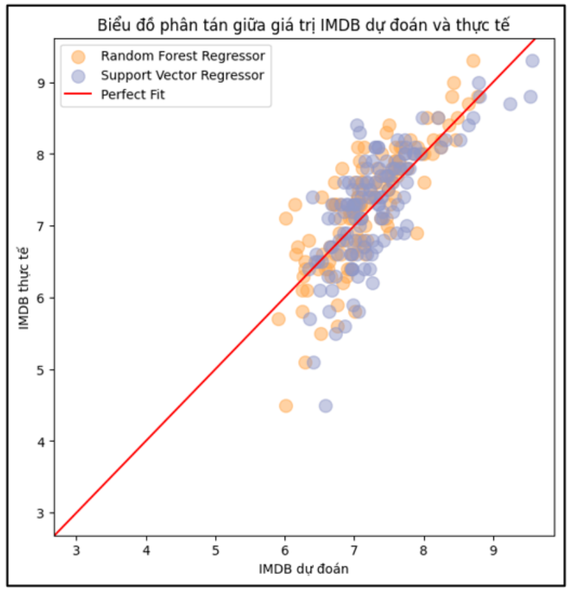
Big DS



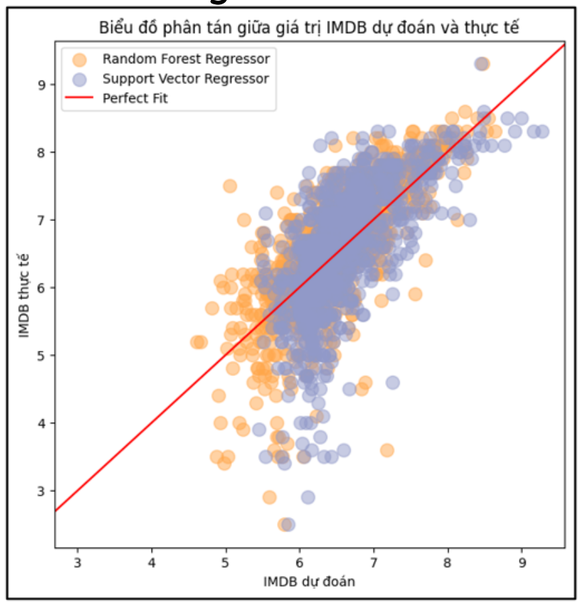
Sau lựa chọn đặc trưng:

Metrics	Small Dataset (1000 mẫu)		Big Dataset (10000 mẫu)	
	RF	SVM	RF	SVM
MAE	0.39	0.42	0.47	0.52
RMSE	0.51	0.55	0.63	0.70
R2	0.61	0.55	0.56	0.45

Small DS



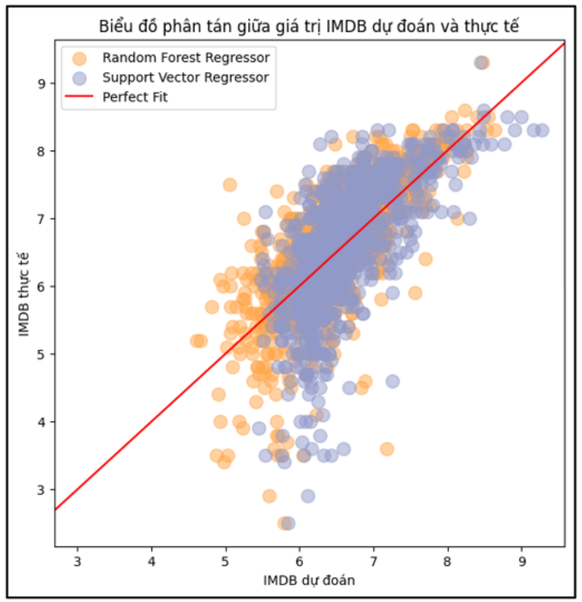
Big DS

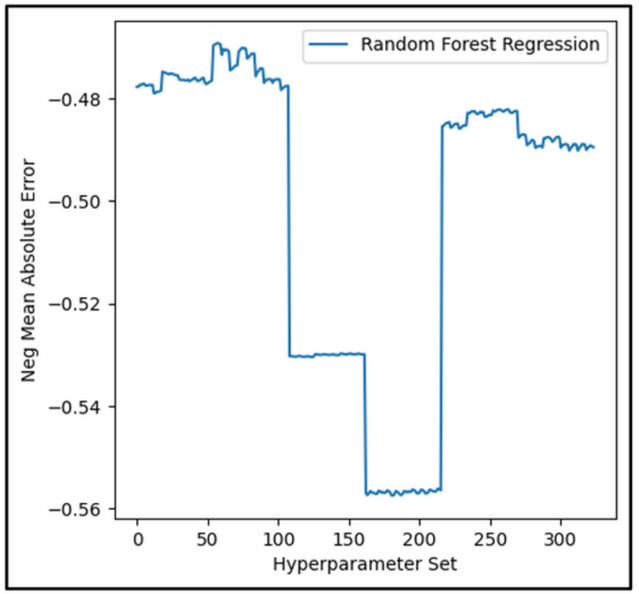


2. Kết quả huấn luyện

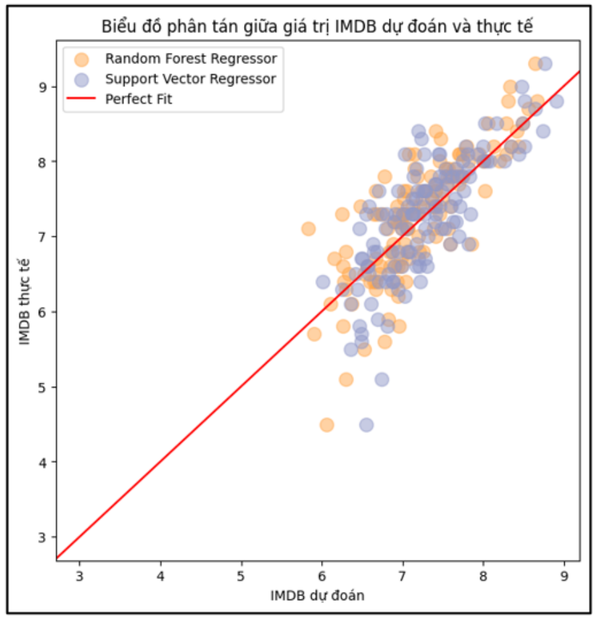
Sau tối ưu tham số:

Metrics	Small Dataset (1000 mẫu)		Big Dataset (10000 mẫu)	
	RF	SVM	RF	SVM
MAE	0.38	0.40	0.46	0.46
RMSE	0.50	0.51	0.61	0.63
R2	0.63	0.61	0.58	0.56

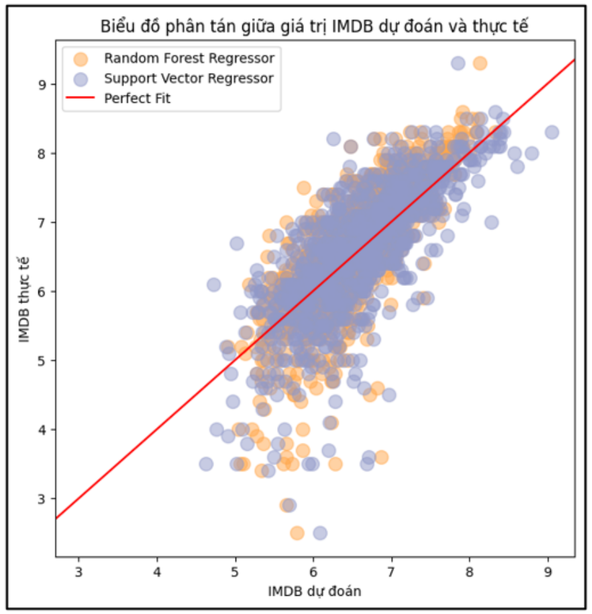


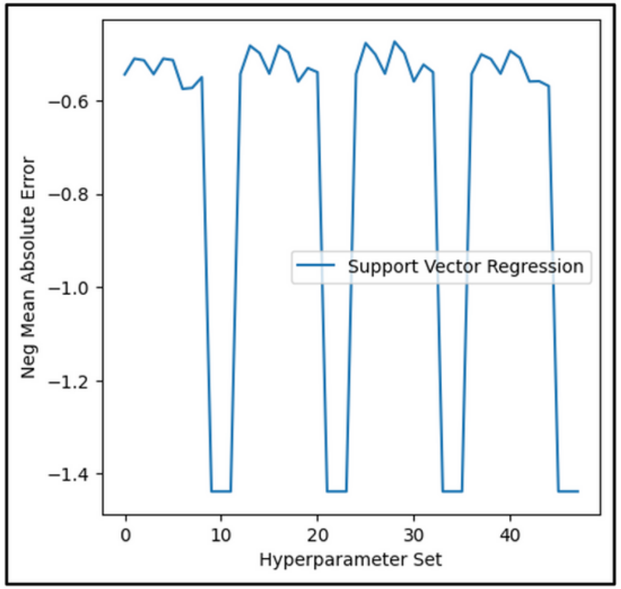


Small DS



Big DS





2. Kết quả huấn luyện

Kết quả trên tập Test:

Metrics	Small Dataset (1000 mẫu)		Big Dataset (10000 mẫu)	
	RF	SVM	RF	SVM
MAE	0.39	0.37	0.47	0.46
RMSE	0.53	0.51	0.64	0.65
R2	0.60	0.63	0.61	0.59

Kết quả trên tập Validation:

Metrics	Small Dataset (1000 mẫu)		Big Dataset (10000 mẫu)	
	RF	SVM	RF	SVM
MAE	0.38	0.40	0.46	0.46
RMSE	0.50	0.51	0.61	0.63
R2	0.63	0.61	0.58	0.56

Đánh giá:

Với tập dữ liệu hoàn toàn mới, cả hai mô hình đều cho ra kết quả dự đoán khá tốt và tương đồng nhau, điều này cho thấy mô hình không gặp các vấn đề như overfitting và underfitting. Mô hình khớp được ~60% dữ liệu và có khả năng dự đoán tốt IMDB rating của bộ phim. Trong đó nhận thấy rõ mô hình SVM cho kết quả tốt hơn trên tập Test và mô hình RF cho kết quả tốt hơn trên tập Validation.

Kết luận:

Bên cạnh những thành quả đạt được, một số vấn đề cần cải thiện:

- Sự đa dạng dữ liệu chưa đồng nhất, phân bố điểm IMDB rating của tập dữ liệu dùng để huấn luyện chưa đồng đều, có rất ít các bộ phim có điểm IMDB rating dưới 5 khiến mô hình cho kết quả dự đoán với sai số lớn.
- Sử dụng thuật toán lựa chọn đặc trưng chưa phù hợp, không góp phần cải thiện mô hình dự đoán.


Thank
you!