|  |  |  |
| --- | --- | --- |
|  | **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  **KHOA CÔNG NGHỆ THÔNG TIN** |  |

TIỂU LUẬN

KHOA HỌC DỮ LIỆU

**TÊN ĐỀ TÀI**

HỆ THỐNG DỰ ĐOÁN ĐIỂM ĐÁNH GIÁ PHIM CHIẾU RẠP

Giảng viên chấm: ………………………………………………..

|  |  |  |
| --- | --- | --- |
| HỌ VÀ TÊN SINH VIÊN | LỚP HỌC PHẦN | ĐIỂM BẢO VỆ  (do GV chấm ghi) |
| Nguyễn Thành Long | 18N15 |  |
| Dương Văn Chính | 18N15 |  |
| Lê Đặng Trường Đạt | 18N15 |  |

ĐÀ NẴNG, 06/2021

# TÓM TẮT

Phim ảnh từ lâu đã trở thành một phương tiện giải trí trong cuộc sống. Với thế giới hiện nay, đã có rất nhiều bộ phim đã và đang được thực hiện trên nhiều phương diện và thể loại. Vì vậy, trong bài tập lớn lần này, chúng em muốn phát triển một hệ thống Dự đoán điểm đánh giá phim chiếu rạp dựa vào mô hình học máy, từ đó giúp người xem có thể đưa ra những lựa chọn tốt nhất. Sau khi thu thập dữ liệu, chúng em tiến hành trực quan hóa và xử lý dữ liệu, từ đó làm cơ sở để chọn ra các đặc trưng phù hợp. Cuối cùng, bằng việc chọn được hai mô hình (*Linear Regression và Neural Network*), chúng em thu được kết quả khá khả quan khi mô hình thu được có tính tổng quát. Các giá trị đánh giá (*metrics*): MSE: ~ 0.005, RMSE: ~0.07, R2\_score: ~0.5.

# BẢNG PHÂN CÔNG NHIỆM VỤ

|  |  |  |
| --- | --- | --- |
| Sinh viên thực hiện | Các nhiệm vụ | Tự đánh giá theo 3 mức  (Đã hoàn thành/Chưa hoàn thành/Không triển khai) |
| Nguyễn Thành Long | Thu thập dữ liệu: mã nguồn.  Trực quan và xử lý dữ liệu: phân tích độ tương quan và trích chọn các đặc trưng.  Mô hình hóa: xây dựng mô hình Neural Network. | Đã hoàn thành. |
| Dương Văn Chính | Thu thập dữ liệu: xử lý các mẫu thu thập bị lỗi.  Trực quan và xử lý dữ liệu: xử lý dữ liệu trống, chuẩn hóa dữ liệu.  Mô hình hóa: xây dựng mô hình Linear Regression. | Đã hoàn thành. |
| Lê Đặng Trường Đạt | Thu thập dữ liệu: tìm chọn chủ đề và nguồn thu thập.  Trực quan và xử lý dữ liệu: mô tả tổng quát dữ liệu, trực quan dữ liệu.  Mô hình hóa: đánh giá và so sánh hiệu quả hai mô hình sau khi đã huấn luyện. | Đã hoàn thành. |

**MỤC LỤC**

[TÓM TẮT 2](#_Toc77738380)

[BẢNG PHÂN CÔNG NHIỆM VỤ 3](#_Toc77738381)

[1. Giới thiệu 5](#_Toc77738382)

[2. Thu thập và mô tả dữ liệu 5](#_Toc77738383)

[2.1 Thu thập dữ liệu 5](#_Toc77738384)

[2.1.1 Nguồn thu thập 5](#_Toc77738385)

[2.1.2 Công cụ thu thập 5](#_Toc77738386)

[2.1.3 Cách thức sử dụng công cụ, đầu vào và đầu ra 5](#_Toc77738387)

[2.2. Mô tả dữ liệu 7](#_Toc77738388)

[2.2.1 Thông tin dữ liệu 7](#_Toc77738389)

[2.2.2 Mô tả trực quan đặc trưng 8](#_Toc77738390)

[3. Trích xuất đặc trưng 12](#_Toc77738391)

[3.1 Lựa chọn đặc trưng 12](#_Toc77738392)

[3.2 Xử lý dữ liệu trống 12](#_Toc77738393)

[3.3 Chuẩn hóa dữ liệu 14](#_Toc77738394)

[4. Mô hình hóa dữ liệu 16](#_Toc77738395)

[4.1 Linear Regression ( Hồi quy tuyến tính ) 16](#_Toc77738396)

[4.2 Neural Network 16](#_Toc77738397)

[4.2.1 Áp dụng Neural Network vào bài toán Dự đoán điểm đánh giá phim: 17](#_Toc77738398)

[4.3 Kết quả đạt được 17](#_Toc77738399)

[4.3.1 Linear Regression 18](#_Toc77738400)

[4.3.2 Neural Network 18](#_Toc77738401)

[4.4 Đánh giá mô hình 19](#_Toc77738402)

[4.4.1 Độ đo (Metrics) 19](#_Toc77738403)

[4.4.2 So sánh mô hình 19](#_Toc77738404)

[5. Kết luận 20](#_Toc77738405)

[6. Tài liệu tham khảo 20](#_Toc77738406)

# 1. Giới thiệu

Bài toán đặt ra là Dự đoán điểm đánh giá của phim chiếu rạp dựa trên dữ liệu thu thập được từ trang web. Đầu tiên, chúng ta cần chọn ra trang web phù hợp cho việc thu thập dữ liệu ( đủ dữ liệu > 1000 mẫu ). Sau khi thu thập được dữ liệu, ta thống kê, mô tả các đặc trưng của dữ liệu, đánh giá độ tương quan giữa các đặc trưng, từ đó có thể trích chọn đặc trưng phù hợp cho đầu vào của quá trình huấn luyện mô hình. Tiếp theo, để có một bộ dữ liệu chuẩn, ta cần xử lý các trường hợp dữ liệu trống, dữ liệu ngoại lai,… Cuối cùng ta chọn mô hình phù hợp (2 mô hình) để huấn luyện dữ liệu và so sánh kết quả đạt được.

# 2. Thu thập và mô tả dữ liệu

## 2.1 Thu thập dữ liệu

**2.1.1 Nguồn thu thập**

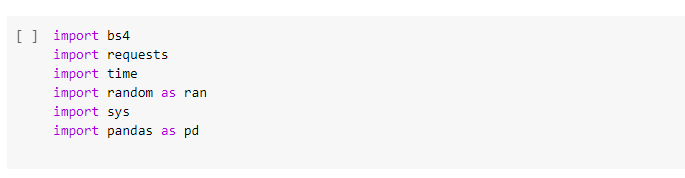
* + [*https://www.imdb.com*](https://www.imdb.com)*:*  **Internet Movie Database** (**IMDb** - cơ sở dữ liệu điện ảnh trên **Internet**) là một trang cơ sở dữ liệu trực tuyến về điện ảnh thế giới. Nó cung cấp thông tin về phim, diễn viên, đạo diễn, nhà làm phim, ... và tất cả những người, công ty trong lĩnh vực sản xuất phim, phim truyền hình và cả trò chơi video.

**2.1.2 Công cụ thu thập**

* + IDE: *Google Colab*
  + Ngôn ngữ lập trình: *Python 3.8*
  + Thư viện: *beautifulsoup4*, *request*

**2.1.3 Cách thức sử dụng công cụ, đầu vào và đầu ra**

1. Truy cập *Google Colab* và kết nối với server.
2. Khai báo thư viện sử dụng.



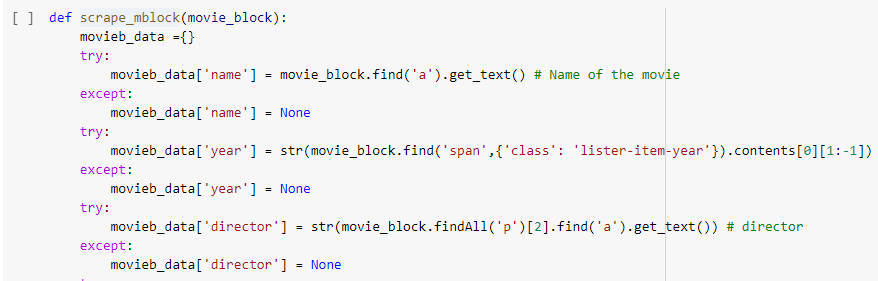
*Hình 1: Khai báo thư viện cho việc thu thập dữ liệu*

1. Khởi tạo đường dẫn đến trang web cần thu thập dữ liệu.
2. Viết mã sử dụng thư viện *beautifulsoup4* và *request* để lấy nội dung trang web dưới dạng *HTML*.



*Hình 2: Khai báo đường dẫn và lấy nội dung trang web dưới dạng HTML.*

1. Sử dụng các hàm *findAll*, *find*, *get\_text*, *get*,… của thư viện *beautifulsoup4*,… để lấy nội dụng dữ liệu trong các *<tab>.*

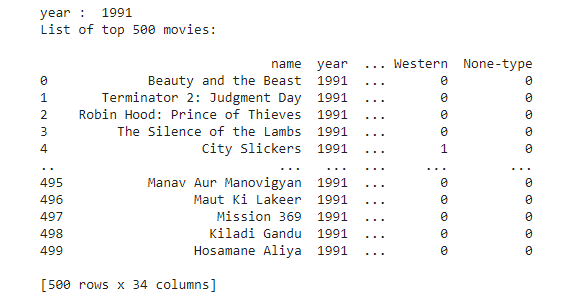


*Hình 3: Lấy dữ liệu bằng cách sử dụng các hàm findAll, find, get\_text, get, ...*

1. Đầu vào là đường dẫn của trang web hiển thị danh sách các phim chiếu rạp trong năm được chọn và số lượng phim cần thu thập dữ liệu của mỗi năm.
2. Đầu ra là một danh sách bao gồm các danh sách con chứa thông tin của mỗi bộ phim từ năm 1991 đến năm 2020 và sau đó được lưu thành file định dạng đuôi .csv.



*Hình 4: Đầu vào là đường dẫn base\_scraping\_link và đầu ra là biến kiểu list films*



*Hình 5: Đầu ra dưới dạng pandas.DataFrame bao gồm thông tin 500 bộ phim trong năm 1991*

## 2.2. Mô tả dữ liệu

**2.2.1 Thông tin dữ liệu**

* + Số mẫu: 12161
  + Số đặc trưng của 1 mẫu: 34
  + Mô tả đặc trưng

**MÔ TẢ CÁC ĐẶC TRƯNG CỦA DỮ LIỆU**

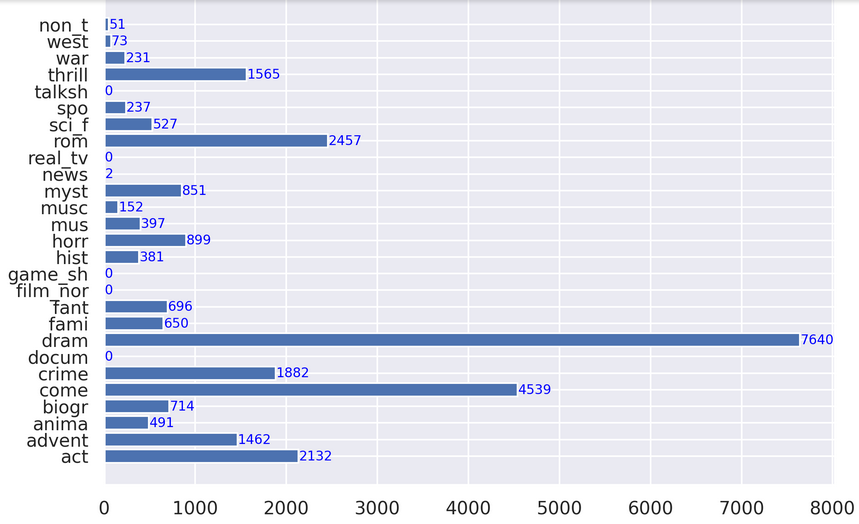
|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Kiểu dữ liệu | Số mẫu dữ liệu trống |
| name | Tên bộ phim | object | 0 |
| year | Năm xuất bản | int64 | 0 |
| director | Đạo diễn | object | 11 |
| rating | Điểm rating | float64 | 0 |
| m\_score | Điểm Mcore | float64 | 3887 |
| vote | Số người đánh giá | float64 | 1134 |
| gross | Chi phí sản xuất | float64 | 1134 |
| act | Hành động | int64 | 0 |
| advent | Thám hiểu | int64 | 0 |
| anima | Hoạt hình | int64 | 0 |
| biogr | Sinh học | int64 | 0 |
| come | Hài | int64 | 0 |
| crime | Tội phạm | int64 | 0 |
| docum | Tài liệu | int64 | 0 |
| dram | Tâm lý | int64 | 0 |
| fami | Gia đình | int64 | 0 |
| fant | Viễn tưởng | int64 | 0 |
| film\_nor | Phim nor cổ điển | int64 | 0 |
| game\_sh | Truyền hình | int64 | 0 |
| hist | Lịch sử | int64 | 0 |
| horr | Kinh dị | int64 | 0 |
| mus | Ca nhạc | int64 | 0 |
| musc | Nhạc sĩ | int64 | 0 |
| myst | Bí ẩn | int64 | 0 |
| news | Thời sự | int64 | 0 |
| real\_tv | Thực tế | int64 | 0 |
| rom | Tình cảm | int64 | 0 |
| sci\_f | Khoa học viễn tưởng | int64 | 0 |
| spo | Thể thao | int64 | 0 |
| talksh | Đối thoại | int64 | 0 |
| thrill | Giật gân | int64 | 0 |
| war | Chiến trân | int64 | 0 |
| west | Cao bồi miền Tây | int64 | 0 |
| non\_t | Không có thể loại | int64 | 0 |

*Bảng 1: Thông tin các đặc trưng dữ liệu*

**2.2.2 Mô tả trực quan đặc trưng**

* + *Thể loại*

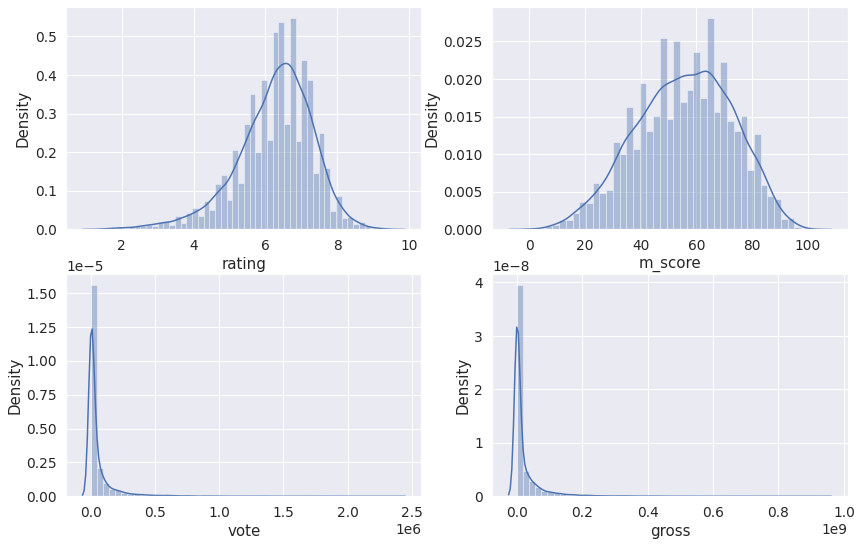
Ta thống kế 27 thể loại phimbằng biểu đồ cột để có một cái nhìn tổng quan về sự phân bố số lượng phim ở các thể loại.

****

*Hình 6: Thống kê số lượng phim qua thể loại*

* *Nhận xét: Không có bộ phim nào thuộc thể loại Talkshow (talksh), Real\_TV (real\_tv), Game\_Show(game\_sh), Film\_Nor (film\_nor), Documentary (docum) ( 0%). Phim thuộc các thể loại Thrill (thrill), Romance (rom), Crime (crime), Comedy (come), Drama (dram), Action (act) có số lượng vượt trội, trong đó phim thuộc thể loại Drama (dram) chiếm số lượng cao nhất ( ~ 63 % ).*
* *Mô tả các đặc trưng số học*

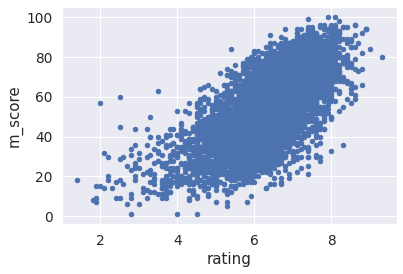
Các đặc trưng số học bao gồm ‘rating’, ‘m\_score’, ‘vote’, ‘gross’. Chúng ta sử dụng biểu đồ mật độ để quan sát sự phân bố của các giá trị của mỗi đặc trưng.

****

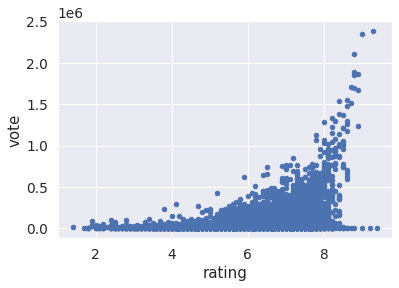
*Hình 7: Thống kê tần suất xuất hiện giá trị của các biến số.*

* ***Nhận xét*** *: Các giá trị của hai biến ‘ rating ‘ và ‘m\_score’ phân bố khá đều và có khả năng tuân theo luật phân phối chuẩn. Bên cạnh đó, hai biến ‘vote’ và ‘gross’ lại thể hiện các giá trị phân bố gần giá trị 0 hơn, thậm chí có nhiều giá trị bằng 0.*
* *Mối quan hệ giữa các biến số*

Bây giờ chúng ta cần quan sát và hình dung được mối quan hệ giữa các biến số thông qua biểu đồ scatter.

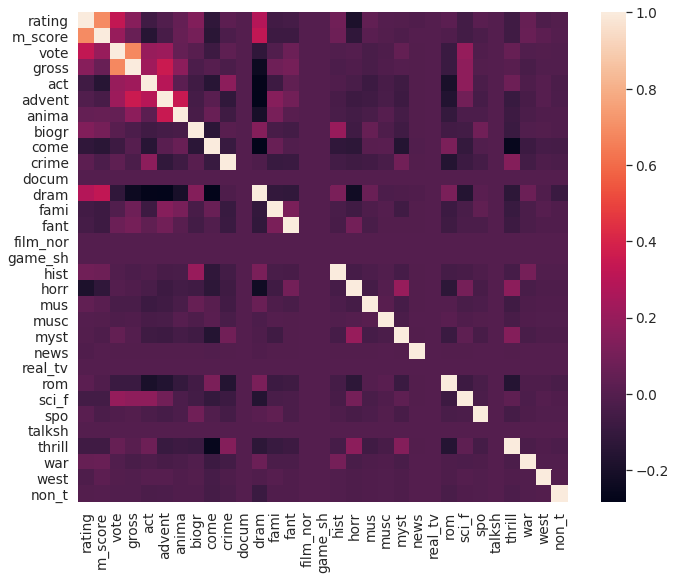
****

*Hình 8: Mối quan hệ giữa ‘rating’ và ‘m\_score’.*

**

*Hình 9: Mối quan hệ giữa ‘rating’ và ‘vote’.*

* ***Nhận xét:*** *Có vẻ như ‘rating’ và ‘m\_score’ có quan hệ gần gũi với nhau, có thể là mối quan hệ* ***Tuyến tính****, còn ‘rating’ và ‘vote’ có thể là mối quan hệ* ***Hàm số mũ****. Tương tự, ta có thể suy ra ‘rating’ và ‘gross’ có mối hạn hệ* ***Hàm số mũ****, vì ‘gross’ và ‘vote’ phân bố khá tương đồng nhau.*
* *Độ tương quan giữa các đặc trưng*

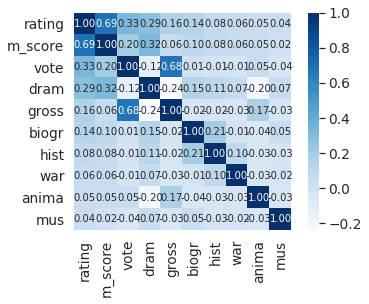


*Hình 10: Bản đồ nhiệt thể hiện độ tương quan giữa các đặc trưng*

* ***Nhận xét:*** *Dễ dàng nhận thấy, các đặc trưng ‘rating’, ‘m\_score’, ‘vote’, ‘dram’, ‘gross’ khá tương quan với nhau, đây là cơ sở cho việc lựa chọn đặc trưng ở phần tiếp theo.*

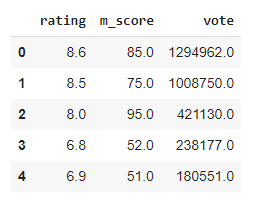
# 3. Trích xuất đặc trưng

**3.1 Lựa chọn đặc trưng**



*Hình 11: Bản đồ nhiệt chi tiết của 10 đặc trưng có độ tương quan cao nhất đối với ‘rating’*

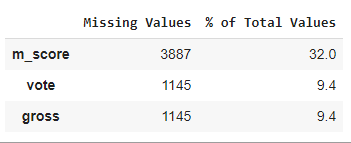
* + *Nhận xét: Qua biểu đồ nhiệt trên, ta có thể nhận thấy ‘m\_score’, ‘vote’, ‘dram’, ‘gross’ có mối quan hệ tương đối gần gũi với ‘rating’ hơn các đặc trưng khác, trong đó giá trị độ tương quan lần lượt là 0.69, 0.33, 0.29, 0.16.*
  + *Nhìn một cách tổng quát, các đặc trưng ‘vote’ (0.33), ‘dram’ (0.29), ‘gross’ (0.16) không thể coi là có mối quan hệ đủ tốt với đặc trưng ‘rating’.Vì vậy, ta không có nhiều sự lựa chọn trong tình huống này và có thể tạm chấp nhận việc lựa chọn hai đặc trưng ‘m\_score’ và ‘ vote’ để huấn luyện dữ liệu.*

**

*Hình 12: Dữ liệu được chọn*

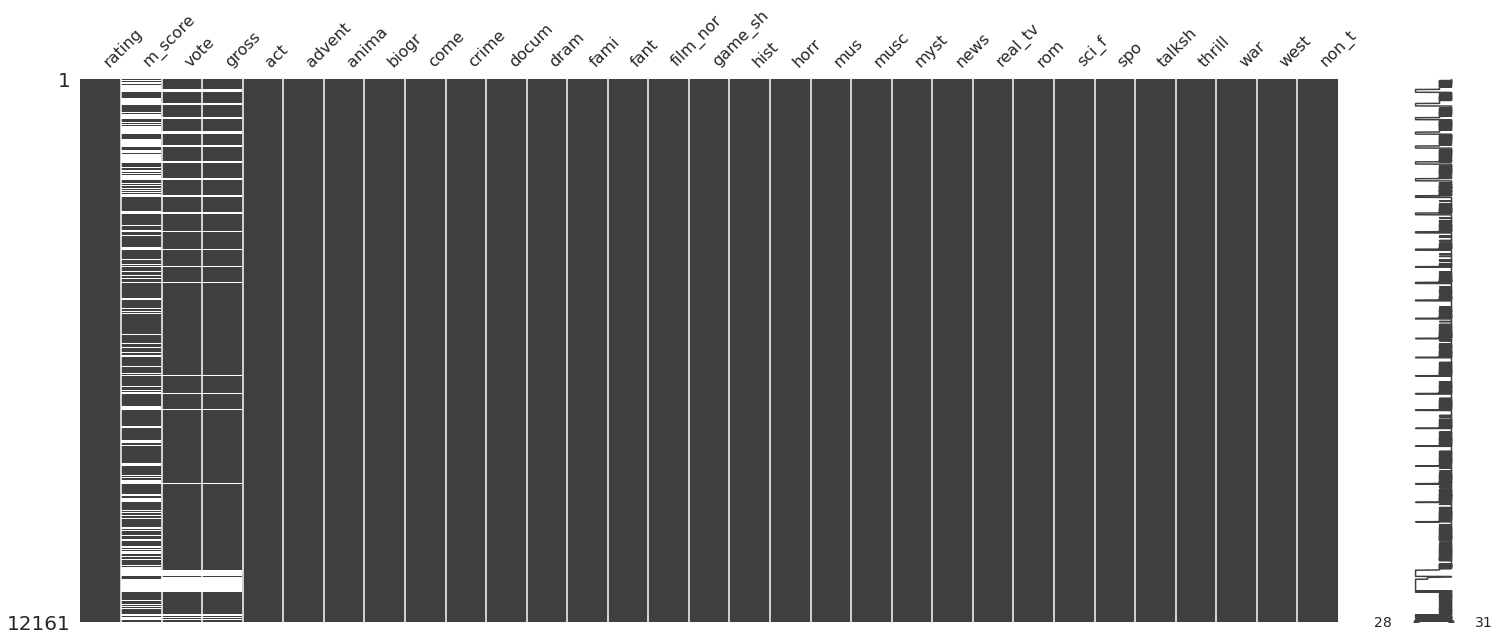
**3.2 Xử lý dữ liệu trống**

Đầu tiên, chúng ta tìm và thống kê số lượng mẫu có dữ liệu trống



*Hình 13: Thống kê số mẫu trống của mỗi đặc trưng*

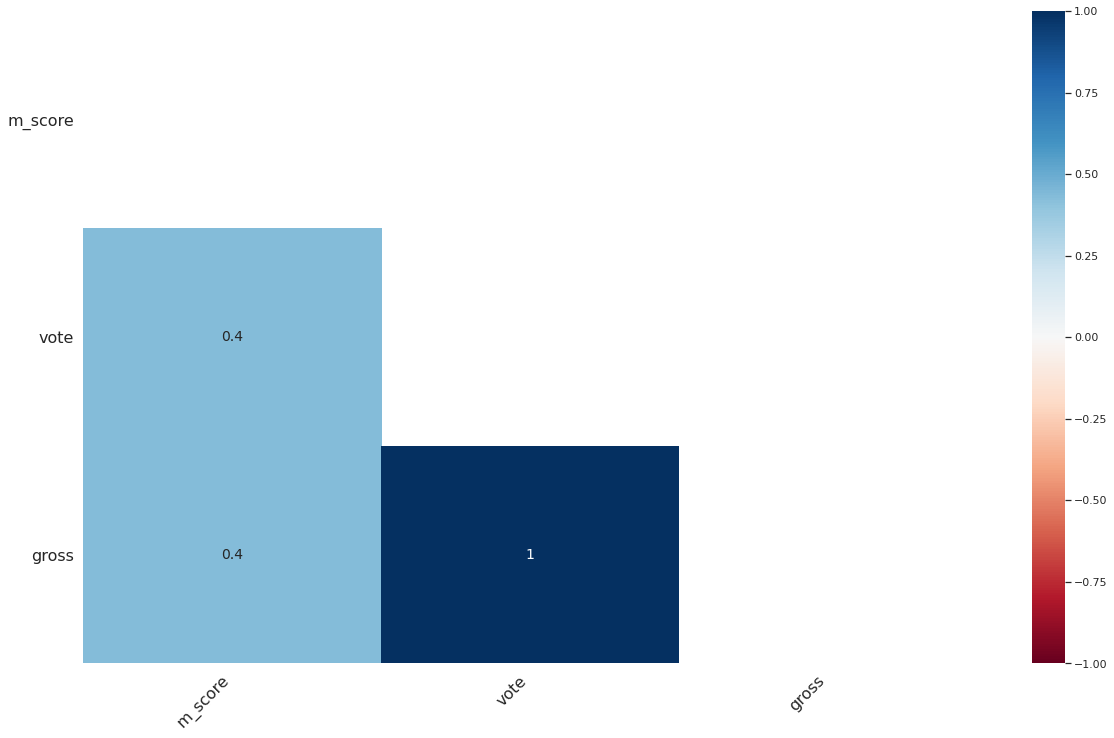
Tiếp theo, quan sát chúng ta quan sát ma trận



*Hình 14: Trực quan dữ liệu trống dựa trên ma trận.*

* ***Nhận xét:*** *Dễ thấy, số mẫu dữ liệu trống nhiều nhất nằm ở cột ‘m\_score’, theo sau đó là ‘gross’, ‘ vote’. Đặc biệt, ‘gross’ và ‘vote’ có số lượng mẫu trống là như nhau. Nên ta có chọn một trong hai đặc trưng trên để tìm mối liên hệ với cột ‘m\_score’.*

Để xử lý dữ liệu, ta cần tìm sự tương quan dữ liệu trống của các đặc trưng

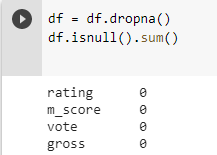


*Hình 15: Độ tương quan số lượng dữ liệu trống.*

* ***Nhận xét:*** *Giá trị 0.4 cho thấy mối liên kết yếu giữa ‘gross’, ‘vote’ và ‘m\_score’; Giá trị 1 cho thấy mối liên kết rất tốt giữa ‘vote’ và ‘gross’. Điều này chứng tỏ, ‘vote’ và ‘gross’ sẽ cùng xảy ra hiện tượng mất dữ liệu cùng nhau, nhưng lại không ảnh hưởng quá nhiều đến ‘m\_score’.*

Qua các khảo sát trên, ta thấy rằng khi mất dữ liệu ở ‘vote’, ‘gross’ thì ‘m\_score’ có thể mất dữ liệu hoặc không (hình 14). Và cụ thể hơn là độ tương quan giữa ‘m\_score’ và ‘vote’, ‘gross’ chỉ là 0.4. Vậy có thể kết luận, trường hợp này là mất dữ liệu hoàn toàn ngẫu nhiêu - MCAR (Missing Completely At Random).

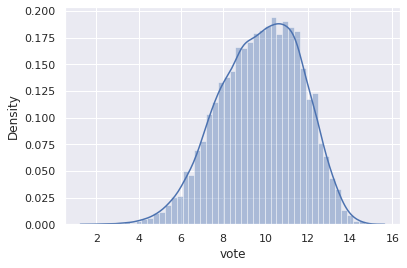
* ***Phương pháp xử lý dữ liệu trống: Loại bỏ các mẫu có dữ liệu trống***

******

*Hình 16: Loại bỏ các mẫu có dữ liệu trống bằng hàm pandas.dropna().*

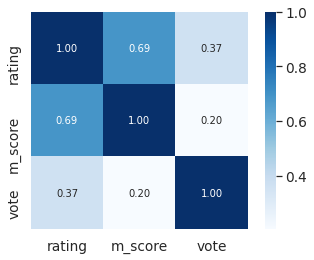
**3.3 Chuẩn hóa dữ liệu**

Trước khi chuẩn hóa, ta hãy quan sát hình 12 và hình 7. Có thể nhận thấy, giá trị ở cột ‘vote’ khá cao và đồ thị không phân bố tuân theo luật phân phối chuẩn. Để mô hình huấn luyện đạt kết quả tốt hơn, ta cần thực hiện một bước nhỏ, đó là lấy logarit các giá trị của cột ‘vote’.



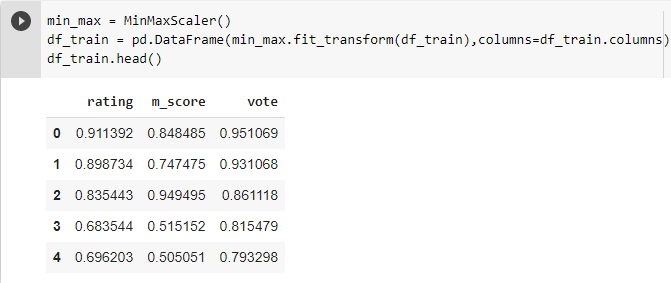
*Hình 17: Biểu đồ mật độ của đặc trưng ‘vote’ sau khi lấy logarit.*

Trực quan bước này bằng bản đồ nhiệt của độ tương quan, ta thấy giá trị độ tương quan giữa ‘rating’-‘vote’ đã tăng lên 0.04 ( so sánh với hình 10 ).



*Hình 18: Biểu đồ nhiệt độ tương quan.*

Tiếp theo, ta chuẩn hóa dữ liệu bằng phương pháp chuẩn hóa *min-max*



*Hình 19: Dữ liệu được chuẩn hóa bằng phương pháp min-max.*

# 4. Mô hình hóa dữ liệu

**4.1 Linear Regression ( Hồi quy tuyến tính )**

“Hồi quy tuyến tính” là một phương pháp thống kê để hồi quy dữ liệu với biến phụ thuộc có giá trị liên tục trong khi các biến độc lập có thể có một trong hai giá trị liên tục hoặc là giá trị phân loại. Nói cách khác “Hồi quy tuyến tính” là một phương pháp để dự đoán biến phụ thuộc, dựa trên giá trị của biến độc lập. Nó có thể được sử dụng cho các trường hợp chúng ta muốn dự đoán một số lượng liên tục. Ví dụ, dự đoán giao thông ở một cửa ung bán lẻ, dự đoán thời gian người ung dừng lại một trang nào đó hoặc số trang đã truy cập vào một website nào đó v.v…

Ta xây dựng hàm mất mát dựa trên việc lấy bình phương của hiệu:

*L*()=

Trong đó là lable được dự đoán, là label đúng. Tối ưu hàm mất mát này bằng cách giải phương trình đạo hàm bằng 0 để tìm ra giá trị trọng số làm cho hàm mất mát có giá trị nhỏ nhất. Quá trình này ta có thể dễ dàng thực hiện bằng việc sử dụng thư viện *scikit-learn.*

**4.2 Neural Network**

Neural Network là một hệ thống tính toán lấy cảm hứng từ sự hoạt động của các neural hệ thần kinh. Tuy nhiên Neural Network chỉ là lấy cảm hứng từ bộ não cách nó hoạt động, chứ không phải là bắt chước toàn bộ các chức năng của nó. Việc chính của chúng ta là dùng mô hình đấy đi giải quyết bài toán chúng ta cần. Mạng Neural Network là sự kết hợp của những tầng perceptron hay còn gọi là perceptron đa tầng. Và mỗi một mạng Neural Network thường bao gồm 3 kiểu tầng là:

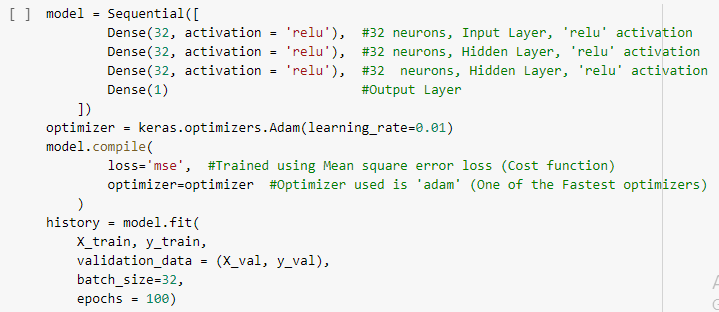
* Tầng input layer (tầng vào): Tầng này nằm bên trái cùng của mạng, thể hiện cho các đầu vào của mạng.
* Tầng hidden layer (tầng ẩn): Tầng này nằm giữa tầng vào và tầng ra nó thể hiện cho quá trình suy luận logic của mạng.
* Tầng output layer (tầng ra): Là tầng bên phải cùng và nó thể hiện cho những đầu ra của mạng.

Tại mỗi tầng, số lượng nút mạng có thể khác nhau còn tùy vào bài toán hoặc cách giải quyết. Tuy nhiên, khi làm việc người ta sẽ để các tầng ẩn số với số lượng nơ-ron khác nhau. Ngoài ra, những nơ-ron nằm ở tầng thường sẽ liên kết đôi với nhau để tạo thành mạng kết nối đầy đủ nhất. Khi đó, người dùng có thể tính toán được kích cỡ của mạng dựa vào tầng và số lượng nơ ron.

**4.2.1 Áp dụng Neural Network vào bài toán Dự đoán điểm đánh giá phim:**

Bằng việc sử dụng thư viện Tensorflow ta có thể dễ dàng xây dựng mô hình Neural Network bao gồm:

* 4 lớp Dense
* 32 neurals mỗi 3 lớp đầu tiên
* 3 lớp đầu sử dụng hàm kích hoạt ‘relu’
* Lớp Dense cuối cùng thể hiện đầu ra tuyến tính với 1 neural
* Sử dụng thuật toán Adam để tối ưu mô hình, trong đó learning rate là 0.01
* Hàm mất mát Mean Square Error
* Số epochs bằng 100
* Batch size bằng 32
* Tập train, test và validation được chia theo tỉ lệ 7:2:1

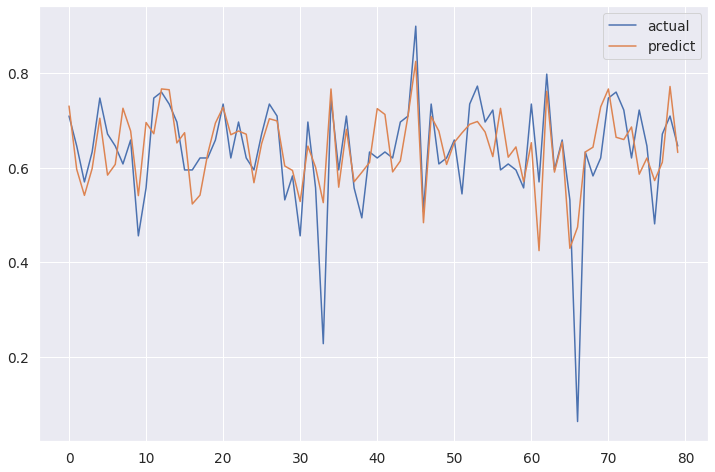


*Hình 20: Xây dựng mô hình trước khi huấn luyện*

**4.3 Kết quả đạt được**

**4.3.1 Linear Regression**

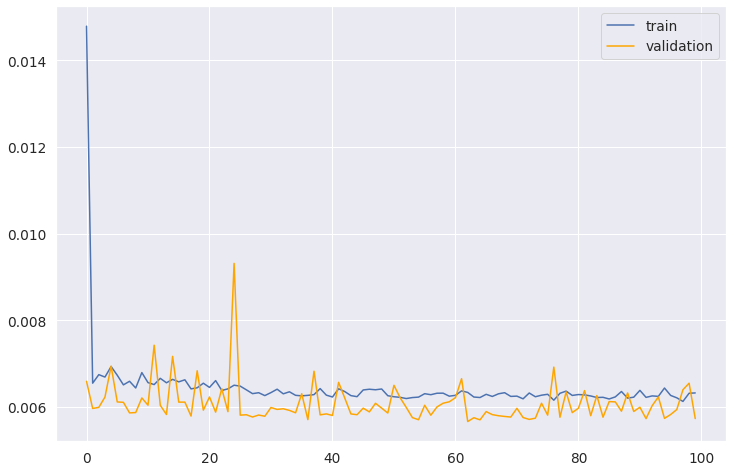
Đồ thị thể hiện sự chênh lệch giữa đầu ra dự đoán và đầu ra trên tập dữ liệu

****

*Hình 21: So sánh đầu ra dự đoán và đầu ra trên tập dữ liệu ( tập test ) (80 mẫu đầu tiên)*

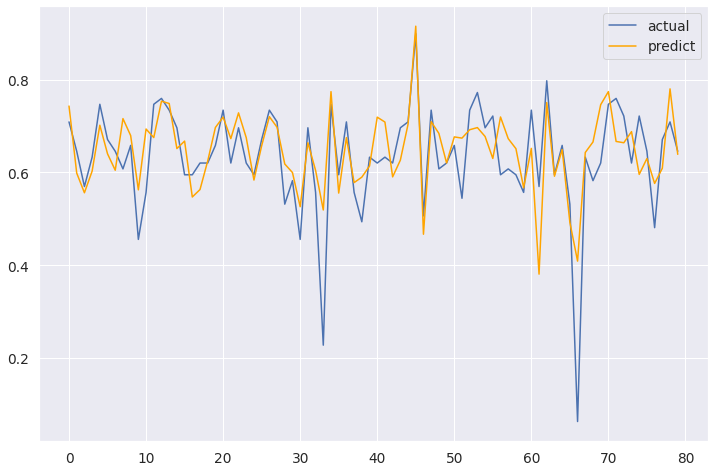
**4.3.2 Neural Network**

Đồ thị thể hiện hiệu suất trên tập huấn luyện và xác thực



*Hình 22: Training loss và Validation loss ( 100 epochs )*

Đồ thị thể hiện sự chênh lệch giữa đầu ra dự đoán và đầu ra trên tập dữ liệu



*Hình 23: So sánh đầu ra dự đoán và đầu ra trên tập dữ liệu ( tập test ) (80 mẫu đầu tiên)*

**4.4 Đánh giá mô hình**

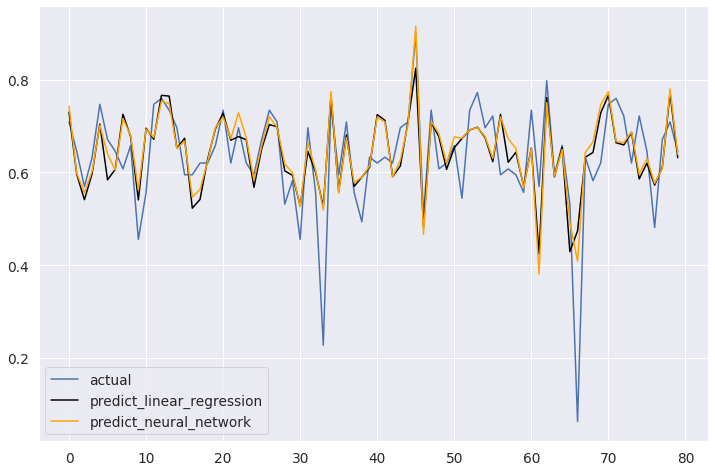
**4.4.1 Độ đo (Metrics)**

**BẢNG SO SÁNH CÁC ĐỘ ĐO METRICS**

|  |  |  |
| --- | --- | --- |
|  | **Linear Regression** | **Neural Network** |
| **MSE** | 0.0059979503995608915 | 0.005682486500773637 |
| **RMSE** | 0.07744643568015827 | 0.07538226914051896 |
| **R2\_score** | 0.5393099588876914 | 0.5635400819831391 |

*Bảng 2: So sánh các độ đo metrics*

**4.4.2 So sánh mô hình**



*Hình 24: So sánh đầu ra dự đoán và đầu ra trên tập dữ liệu ( tập test ) (80 mẫu đầu tiên)*

* + ***Nhận xét:*** *Quan sát bảng độ đo metrics ( bảng 1), ta có thể thấy rằng sự chênh lênh giữa hai mô hình là không quá lớn và mô hình mạng nơ ron cho kết quả tốt hơn. Điều này cũng có thể dễ phát hiện khi nhìn vào độ thị trên ( hình 21 ) bên cạnh đó, mô hình dự đoán có tính tổng quát, không bị quá khớp hoặc quá sai so với mô hình dữ liệu.*

# 5. Kết luận

Thông qua bài tiểu luận, chúng em đã học được cách thu thập dữ liệu từ một trang web bằng việc sử dụng thư viện *beautifulsoup4*, *request*; trực quan hóa dữ liệu bằng thư viện *seaborn*, *matplotlib*; xử lý dữ liệu bằng thư viện *numpy*, *pandas*. Việc trực quan hóa cho chúng em một cái nhìn tổng quan về mỗi đặc trưng và mối liên hệ giữa các đặc trưng, từ đó có thể trích xuất, chọn ra những đặc trưng phù hợp cho bài toán, bên cạnh đó còn có một số bước xử lý làm sạch dữ liệu như *Xử lý dữ liệu trống, Ngoại lệ, Chuẩn hóa dữ liệu.* Từ đó lấy làm cơ sở cho việc chọn lựa mô hình huấn luyện ( *Linear Regression và Neural Network* ). Kết quả đạt được khá tốt với các giá trị đánh giá mô hình (*metrics)* MSE: ~ 0.005, RMSE: ~0.07, R2\_score: ~0.5. Cuối cùng, để bài toán có kết quả tốt hơn, chúng em hy vọng đưa ra các giải pháp, định hướng của nhóm như: thu thập nhiều dữ liệu với nhiều đặc trưng hơn, đánh giá sự tương quan giữa các đặc trưng kĩ càng hơn, và tìm hiểu, chọn ra mô hình huấn luyện phù hợp hơn.

# 6. Tài liệu tham khảo

1. [*https://www.kaggle.com/pandula/a-simple-tutorial-on-beautifulsoup*](https://www.kaggle.com/pandula/a-simple-tutorial-on-beautifulsoup)
2. [*https://www.kaggle.com/parulpandey/a-guide-to-handling-missing-values-in-python*](https://www.kaggle.com/parulpandey/a-guide-to-handling-missing-values-in-python)
3. [*https://www.kaggle.com/pmarcelino/comprehensive-data-exploration-with-python/notebook*](https://www.kaggle.com/pmarcelino/comprehensive-data-exploration-with-python/notebook)
4. [*https://www.kaggle.com/rtatman/data-cleaning-challenge-scale-and-normalize-data*](https://www.kaggle.com/rtatman/data-cleaning-challenge-scale-and-normalize-data)
5. [*https://machinelearningcoban.com/2016/12/28/linearregression/*](https://machinelearningcoban.com/2016/12/28/linearregression/)
6. [*https://machinelearningmastery.com/check-point-deep-learning-models-keras/*](https://machinelearningmastery.com/check-point-deep-learning-models-keras/)
7. [*https://www.analyticsvidhya.com/blog/2021/06/linear-regression-using-neural-networks/*](https://www.analyticsvidhya.com/blog/2021/06/linear-regression-using-neural-networks/)
8. [*https://www.aionlinecourse.com/tutorial/machine-learning/evaluating-regression-models-performance*](https://www.aionlinecourse.com/tutorial/machine-learning/evaluating-regression-models-performance)