

## รายชื่อสมาชิก

1. กชกร สนิทกุล 62050126
2. เกษราภรณ์ โคเพื่อง 62050130
3. ฐิติรัตน์ ผสมทรัพย์ 62050147
4. ณัฐชริกา ทองแถม 62050154

## Code Study and Reference

- <https://www.kaggle.com/code/saurav9786/imdb-score-prediction-for-movies/notebook>

## Dataset

- <https://www.kaggle.com/datasets/carolzhangdc/imdb-5000-movie-dataset>

## Introduction

เลือก **IMDB 5000 Movie Dataset** มาในการ Train, Test โมเดล ซึ่งตัว Dataset ประกอบไปด้วย Attribute ทั้งหมดดังนี้

1. Color : ภาพยนตร์เรื่องนั้นเป็นแบบสีหรือขาวดำ
2. Director\_name: ชื่อผู้กำกับ
3. Duration : ความยาวของภาพยนตร์
4. Director\_facebook\_likes: ยอดกดถูกใจเพจ Facebook ผู้กำกับ
5. Actor\_1\_name: ชื่อนักแสดงนำคนที่ 1
6. Actor\_1\_facebook\_likes: ยอดกดถูกใจเพจ Facebook นักแสดงนำคนที่ 1
7. Actor2\_name: ชื่อนักแสดงนำคนที่ 2
8. Actor\_2\_facebook\_likes: ยอดกดถูกใจเพจ Facebook นักแสดงนำคนที่ 2
9. Actor\_3\_name: ชื่อนักแสดงนำคนที่ 3
10. Actor\_3\_facebook\_likes: ยอดกดถูกใจเพจ Facebook นักแสดงนำคนที่ 3
11. Cast\_total\_facebook\_likes: ยอดกดถูกใจเพจ Facebook รวมของนักแสดง
12. Movie\_facebook\_likes: ยอดกดถูกใจภาพยนตร์เรื่องนี้บน Facebook
13. Gross: รายได้
14. Genres: ประเภทของภาพยนตร์ เช่น 'Animation', 'Comedy', 'Romance', 'Horror', 'Sci-Fi', 'Action', 'Family'
15. Movie\_title: ชื่อภาพยนตร์
16. Num\_voted\_users: จำนวนคนโหวตให้ภาพยนตร์เรื่องนี้ (คะแนน)
17. Num\_user\_for\_reviews: จำนวนคนรีวิวให้ภาพยนตร์เรื่องนี้ (คำวิจารณ์)
18. Facenumber\_in\_poster: จำนวนนักแสดงในโปสเตอร์

19. Plot\_keywords: คีย์เวิร์ดพล็อตของภาพยนตร์
20. Movie\_imdb\_link : ลิงค์ IMDb ของภาพยนตร์
21. Language: ภาษาของภาพยนตร์
22. Country: ประเทศที่ภาพยนตร์ถูกผลิตขึ้นมา
23. Content\_rating: เรตติ้งภาพยนตร์
24. Budget: ต้นทุนภาพยนตร์
25. Title\_year: ปีที่ฉาย
26. Imdb\_score: คะแนน IMDb
27. Aspect\_ratio: อัตราส่วนภาพของภาพยนตร์

โดย Dataset ชุดนี้เก็บข้อมูลของภาพยนตร์และคะแนนของภาพยนตร์เรื่องนั้น ๆ โดยรวบรวมภาพยนตร์ทั้งหมด 5000 เรื่อง โดยมีภาพยนตร์ตั้งแต่ปีค.ศ. 1916 - 2016

และจุดประสงค์ของการสร้าง Model โดยนำข้อมูลชุดนี้มาใช้คือการสร้างโมเดลที่สามารถ Predict คะแนน IMDb (IMDb Rating) จาก Attribute ที่เหมาะสมได้

## Feature Selection

ทำการ Drop Attribute ที่เป็น Text จำพวก Name และ Link ทั้ง ได้แก่ Director\_name, Actor\_1\_name, Actor\_2\_name, Actor\_3\_name, Movie\_title และ Movie\_imdb\_link (ข้อมูลทั้งหมดที่ถูกตัดเป็น String) จากนั้นทำการ Drop Attribute ที่ไม่เกี่ยวข้องหรือไม่ต้องการใช้ในการคำนวณคะแนน ได้แก่ num\_critic\_for\_reviews, gross, num\_voted\_users, num\_user\_for\_reviews, cast\_total\_facebook\_likes, aspect\_ratio

จากนั้นทำการเช็คข้อมูลใน Dataset ว่าไม่มีข้อมูล NaN หรือ Column ไหนที่มีข้อมูลว่าง ละเมื่อตรวจพบข้อมูลที่มีข้อมูลว่าง ก็จะ Drop Row ที่ NaN ทั้งไป โดยหลังจาก Drop ข้อมูล Row นั้น ๆ ทิ้งไปแล้ว พบว่ามีข้อมูลเหลืออยู่ 84.39% หรือ 4256 Row จาก 5043 Row และเมื่อทำการตรวจสอบข้อมูลว่าไม่มีข้อมูลที่ว่างแล้ว ก็ทำการเช็คใน Dataset ว่ามีข้อมูลที่ซ้ำกันหรือไม่ต่อ และเมื่อพบว่ามี ก็ทำการ Drop ข้อมูลที่ซ้ำกันทิ้ง โดยหลังจาก Drop ทิ้งแล้ว พบว่าเหลือข้อมูลอยู่ 4158 จาก 5043 หรือ 82.45%

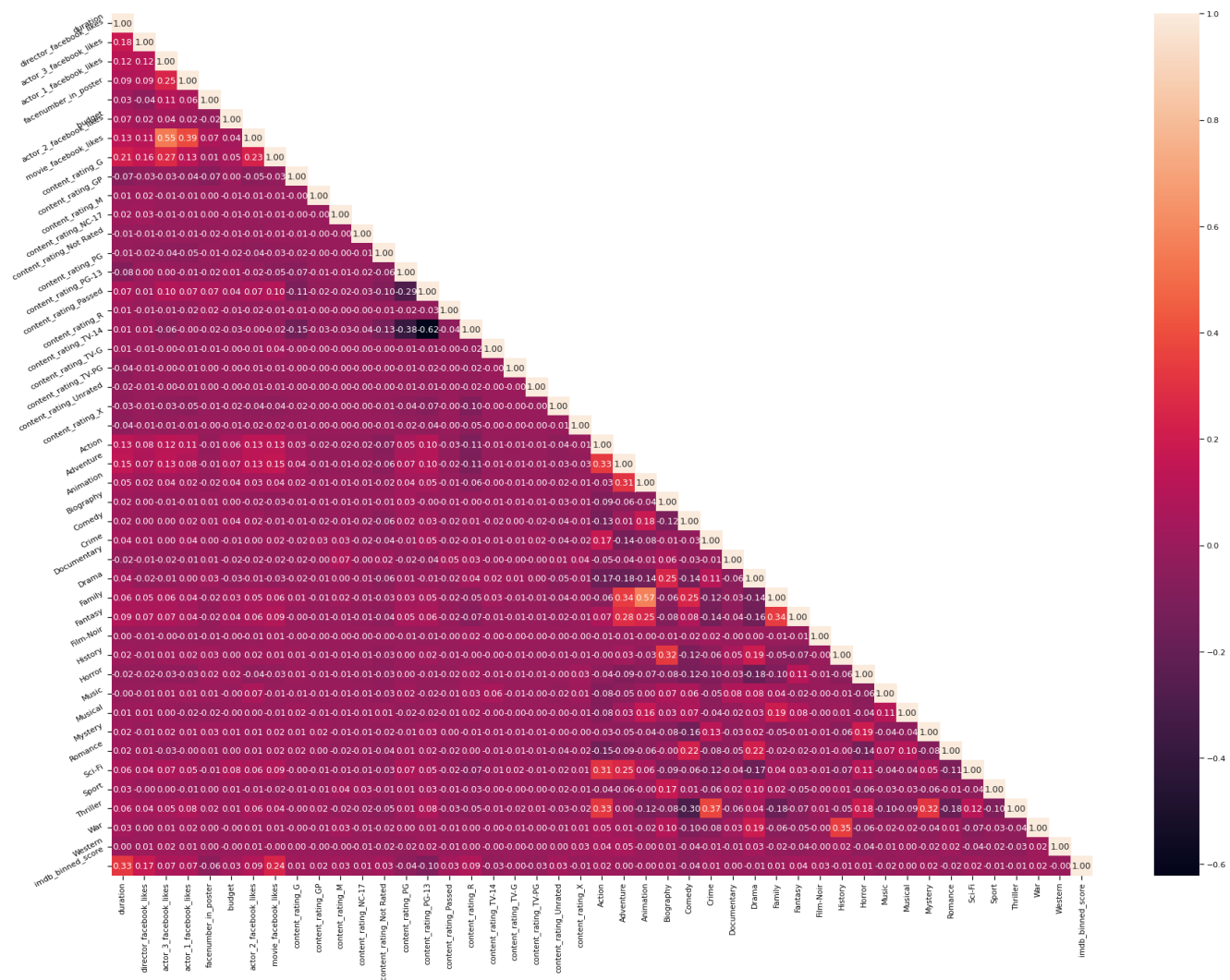
เมื่อได้ข้อมูลใน Row ที่ไม่ NaN และแต่ละ Row ไม่ซ้ำกันแล้ว ก็ทำการเช็คจำนวนความถี่ของข้อมูลในแต่ละ Column โดย Column ที่นำมาตรวจสอบ ได้แก่ Language, Country และ Color จากนั้นจะทำการ Drop Column Language, Country และ Color ทิ้ง เนื่องจากเป็นข้อมูลที่มีความถี่อยู่ใน Attribute เดียวเป็นจำนวนมาก เช่น Language พบว่าภาพยนตร์ถึง 3973 จาก 4158 ที่เป็นภาษาอังกฤษ, ภาพยนตร์ 3251 เรื่องจากทั้งหมดที่ถูกผลิตใน USA และภาพยนตร์ 3986 เรื่องจาก 4158 เรื่องที่เป็นภาพยนตร์สี โดยจะทำการ Drop Column Language, Country และ Color ทิ้ง

จากนั้นทำการ Drop Column plot\_keywords เนื่องจากเป็นชุดข้อมูลที่เฉพาะของแต่ละเรื่องและมีความแตกต่างกันมาก จึงไม่นำมาใช้

เมื่อได้ข้อมูลที่คาดว่าจะใช้แล้ว ก็จะนำมาปรับให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถคำนวณได้ เริ่มจาก Column Genres โดยใน Dataset นี้มีอยู่ทั้งหมด 22 ประเภท โดยจะใช้วิธีแบบ One-Hot Code คือ นำทั้ง 22 ประเภทนี้แยกออกเป็น Column ประเภทละ 1

### Column Content-Rating

เมื่อได้ข้อมูลทั้งหมดที่พร้อมคำนวณกับคอมพิวเตอร์ ก็นำมาเซต **Correlation** โดยได้ค่า **Correlation** ของแต่ละคอลัมน์ดังนี้



## Preprocessing

ทำการแยกคอลัมน์ข้อมูล X (Input) และ y (Output) โดย X จะมีคอลัมน์

1. Duration	23. Action
2. Director_facebook_likes	24. Adventure
3. actor_3_facebook_likes	25. Animation
4. Actor_1_facebook_likes	26. Biography
5. Facenumber_in_poster	27. Comedy
6. Budget	28. Crime
7. Actor_2_facebook_likes	29. Documentary
8. Movie_facebook_likes	30. Drama
9. content_rating_G	31. Family
10. content_rating_GP	32. Fantasy
11. content_rating_M	33. Film-Noir
12. content_rating_NC-17	34. History
13. content_rating_Not Rated	35. Horror
14. content_rating_PG	36. Music
15. content_rating_PG-13	37. Musical
16. content_rating_Passed	38. Mystery
17. content_rating_R	39. Romance
18. content_rating_TV-14	40. Sci-Fi
19. content_rating_TV-G	41. Sport
20. content_rating_TV-PG	42. Thriller
21. content_rating_Unrated	43. War
22. content_rating_X	44. Western

และ y จะมี Column imdb\_binned\_score ซึ่งเป็น Column Output

จากนั้นทำการ LabelEncoder กับข้อมูลในแกน y เพื่อให้พร้อมกับการทำงานในโมเดล

เมื่อข้อมูลทั้งหมดพร้อม ก็ทำการ Split ข้อมูลออกเป็น 2 ชุด คือสำหรับ Train และ Test

## Model

สร้างโมเดลสำหรับ Train และ Test โดยจะเลือกสร้างทั้งหมด 9 โมเดลและนำมาเปรียบเทียบค่า Accuracy กัน โดยโมเดลที่สร้างมีดังนี้

1. Logistic Regression
2. KNN
3. SVC
4. Naive Bayes
5. Decision Tree
6. Random Forest
7. Bagging
8. Gradient Boosting

โดยแต่ละโมเดลมีค่า Accuracy ดังนี้

Model	Accuracy (f1-score)
Logistic Regression	0.65
KNN	0.63
SVC	0.62
Naive Bayes	0.07
Decision Tree	0.57
Random Forest	0.67
Bagging	0.6859066859066859
Gradient Boosting	0.67

(ดูผลทั้งหมดได้ที่ <https://eithub.com/NuttharikaTht/MovieIMDbScoreModel/blob/main/model.ipynb> ในส่วน Model Comparison)

จากผลจะเห็นว่า Model ที่ใช้วิธี Bagging มีค่า Accuracy สูงที่สุด คือประมาณ 0.69 และ Model ที่ใช้วิธี Naive Bayes มีค่า Accuracy ต่ำที่สุดคือ 0.07