Predicting IMDb Movies with KNN and Linear Regression

Luis Damy

Department of Artificial Intelligence
Universidade Federal de São Paulo
São José dos Campos, SP, Brasil
damy.luis@unifesp.br

Abstract— **A indústria de Filmes tem sido um dos centros mais importantes de entretenimento, além de um overflow na quantidade de dados sendo gerados a respeito de opiniões e reviews de pessoas no século XXI, com a chegada da internet. Dessa forma, a área de pesquisa de aprendizado de maquina tem sido cada vez mais popular para realizar estudos aplicados a previsões, tais como o de notas de filmes no banco de dados da IMDB. Foi utilizado nesse trabalho a data set da IMDB para predizer notas de filmes a partir dos algoritmos KNN que teve acurácia de 78%, além do algoritmo de regressão linear com 62%.**

Keywords— **IMDB, Machine Learning, movie prediction, movie rating.**

I. Introduction

A indústria de filmes tem expandido mundialmente, tendo oportunidade de assistir filmes no dia do lançamento e compartilhar a opinião sobre como foi em questão de segundos. Essa área trás desafios tais como a capacidade de produzir filmes de ótima qualidade que agradem o publico a fim de coletar o máximo de popularidade dentre os vários gêneros. Para que haja uma convergência de dados no publico mundial, uma plataforma de criticas e reviews incluindo notas, informações dos atores, diretores, resumos e custo de produção foi criada e é a mais famosa, chamada Internet Movie Database (IMDB) [1].

Há uma grande quantidade de estudos sobre previsão de filmes. Em alguns deles, a mineração de dados por meio de algoritmos é utilizada. Dessa forma é possível saber com eficiência quais modelos de previsão tais como KNN, Arvores de Decisões, Regressão Logica, MLP, Naive Bayes, SMO, Regressão Linear, são mais apropriados.

II. Related Work

Muitos outros pesquisadores tem utilizado várias técnicas de aprendizado de maquina para prever a taxa de sucesso. Os dados sobre os atributos de um filme ao invés de redes sociais estão sendo utilizada para analisar e foi descoberta que a regressão logística traz uma precisão de 84% [5]. Pramod, Abhisht e Geetha mostram que a logica Fuzzy contem uma acurácia alta para categorizar previsões [6]. O acesso a aprendizado de maquina estão sem do aplicado em datasets sintéticos para construir uma estrutura eficiente para predizer usando IMDB utiliza [7]. Outros algoritmos de aprendizado de maquina estão sendo usados para prever a taxa de sucesso de um filme [8]. Dependendo de vários atributos de filmes aplicados em modelos matemáticos estão sendo implementados para determinar o sucesso de filmes.

Outro estudo mostra as divisões no mercado domestico e internacional de filmes da Rússia, em que os filmes internacionais estão a frente do ano de 2002 até 2014 [1]. Esse trabalho também diferencia três fatores por trás de um filme bem sucedido. Eles são, custo, atrizes e atores populares, diretores e opiniões de espectadores. Por fim, o modelo conclui que essas sanções e o orçamento tem um efeito consideravelmente alto para o sucesso do filme.

III. Methodology

Neste estudo foi feito utilizado a classificação de dados, uma das formas mais comuns de técnicas de aprendizado de maquina conforme a Figura 1.

Os passos estão descritos abaixo:

- Extração de dados
- Pré processamento de dados
- Aplicação de técnicas de aprendizado de maquina
- Comparação dos resultados de diferentes algoritmos

O método de classificação supervisionada é separado em vários grupos. A data set é dividido em grupos de treinamento e teste.

O primeiro é utilizado para modelar, treinar o algoritmo por meio de uma porcentagem dos dados extraídos e moldados. Já o segundo será aplicado para realizar o teste e obter o resultado a partir da acurácia em relação ao de teste.

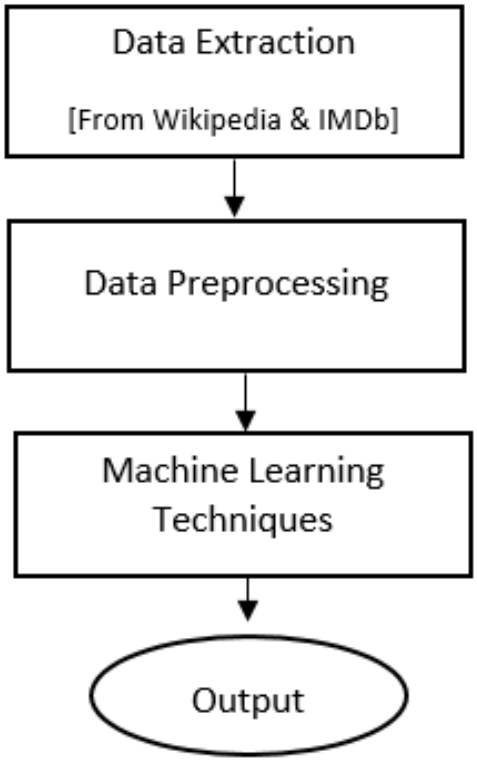

Fig.1 Workflow da metodologia do estudo

Para que seja feita a analise correta da data set, ela deve ser formatada e limpa. Essa fase é chamada de pré-processamento.

Nesse estudo, foi feita a remoção de campos desnecessários tais como nomes de diretores, atores, idioma, etc. Em seguida foi separado a tabela de itens relevantes para o estudo.

TABLE I

Índices	Colunas
0	Gênero
1	Duração
2	Rating
3	Review Critico
4	Review Usuários
5	Votos Usuários
6	Votos Reviews
7	Likes do filme
8	Likes do diretor
9	Likes dos atores
10	Classificação Indicativa

A. The Movie Dataset

A extração de dados foi realizada pela leitura do arquivo IMDB.csv, provenientes do site IMDB [1]. A data set utilizada contem cerca de 5000 linhas e 30 colunas de informações a principio.

B. Data Extraction

Por meio de funções de leitura de arquivos disponibilizados na biblioteca Pandas foi possível a extração e filtragem dessas informações. A partir disso, foi realizada a etapa de pré-processamento com a remoção de colunas como id.

Ademais, foi feito um despejamento das informações filtradas na leitura do primeiro arquivo para outro arquivo chamado IMDB_likes_review.csv. Por meio desse segundo arquivo, foi realizado uma distribuição das informações por meio histogramas representados a nas Figura 2.

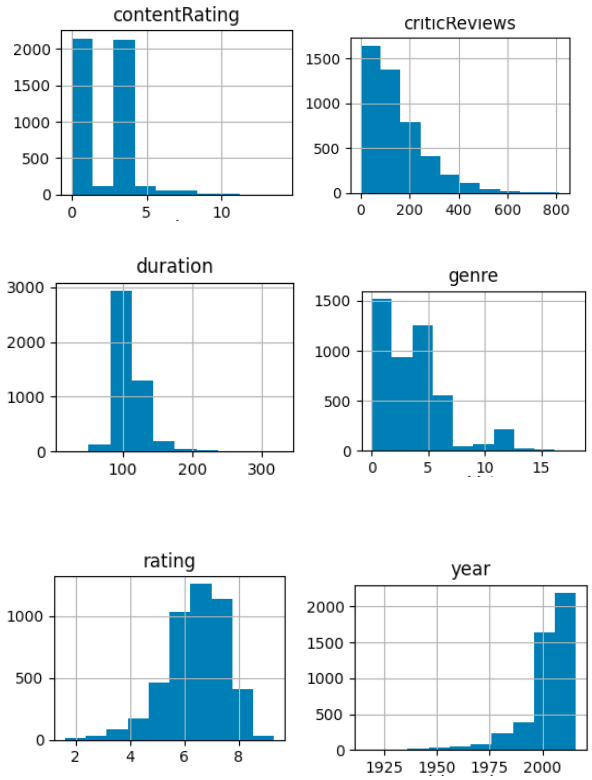

Fig.2 Estatística de dados pré-processamento

Dessa forma, foi utilizado no carregamento de dados para os modelos algumas informações relevantes para a utilização das técnicas de aprendizado de maquina. Com esse processo, o filtro de informações trouxe os dados a serem utilizados nos algoritmos cerca de 4500 filmes.

IV. Experimental Results

A. *Linear Regression*

O modelo de regressão linear em estatística é um modelo com uma variável explorada. Isto é, em um gráfico bidimensional, há pontos de exemplo e uma variável dependente e outra independente. Esse método funciona de forma que ele tenta buscar a função linear, uma reta, que com a maior precisão possível prevê os valores da variável dependente como uma função da variável independente.

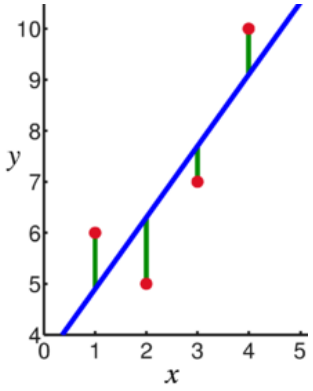

Fig.3 Explicação Linear Regression

O algoritmo de regressão linear utilizado no estudo foi o Ordinary Least Squares (OLS) por ser um dos métodos mais simples para realizar os testes. Esse método tem como foco a minimização da soma das diferenças de quadrados observados e de valores preditos.

A partir do treino dos dados com a função train_test_split disponível em sklearn model_selection foi possível a realização dos treinos dos dados de treino responsáveis por 40% do total.

Além disso, por meio da biblioteca statsmodels.api foi possível a utilização do modelo "sm" que contem a função OLS capaz de realizar o teste, passando como argumentos os valores de x e y de treino. Neles, sabe-se que o y contem o valor original dos votos.

Os resultados obtidos foram salvos por meio da função fit() do próprio sm. Em seguida, foi feito um resumo destas informações e escrita em um arquivo chamado "OLS_result.txt".

OLS Regression Results

Dep. Variable:	rating	R-squared (uncentered):	0.979
Model:	OLS	Adj. R-squared (uncentered):	0.979
Method:	Least Squares	F-statistic:	1.876e+04
Date:	Sun, 31 Jul 2022	Prob (F-statistic):	0.00
Time:	01:29:51	Log-Likelihood:	-3768.6
No. Observations:	2784	AIC:	7551.
Df Residuals:	2777	BIC:	7593.
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
fbPosters	-0.0420	0.009	-4.642	0.000	-0.060	-0.024
year	0.0022	4.82e-05	45.985	0.000	0.002	0.002
duration	0.0135	0.001	15.562	0.000	0.012	0.015
contentRating	0.1200	0.010	12.122	0.000	0.101	0.139
criticReviews	0.0013	0.000	6.710	0.000	0.001	0.002
userReviews	-0.0005	7.65e-05	-6.727	0.000	-0.001	-0.000
userVotes	3.248e-06	2.17e-07	14.940	0.000	2.82e-06	3.67e-06

Omnibus:	350.876	Durbin-Watson:	2.006
Prob(Omnibus):	0.000	Jarque-Bera (JB):	632.304
Skew:	-0.822	Prob(JB):	4.98e-138
Kurtosis:	4.658	Cond. No.	9.42e+04

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[3] The condition number is large, 9.42e+04. This might indicate that there are
strong multicollinearity or other numerical problems.']]

Fig.4 Arquivo OLS Resultado

Com base nos dados obtidos do Arquivo (Fig. 4) é possível mostrar por meios matemáticos houve uma acurácia de 62%. Esse valor foi calculado a partir dos resultados da predição dos testes tipo X e por meio da do calculo dele com a acurácia dos casos reais do tipo Y, a diferença entre eles é de 0.620286. Além disso, o tempo de execução do algoritmo de regressão foi de 0.0016 segundos.

B. K-Nearest Neighbours

O Algoritmo KNN é um método de aprendizado não paramétrico supervisionado utilizado para classificação e regressão. Ele utiliza uma técnica em que consegue designar pesos aos contribuidores dos seus "vizinhos", representando os valores pontos mais próximos ao seu redor. Dessa forma, os vizinhos mais próximos contribuem para uma media maior do que as distancias dos outros. (Figura 5) Uma vez que designado o valor de K, o vizinho que estiver contido dentro da distancia dessa atribuição será designado com a categoria do mais próximo.

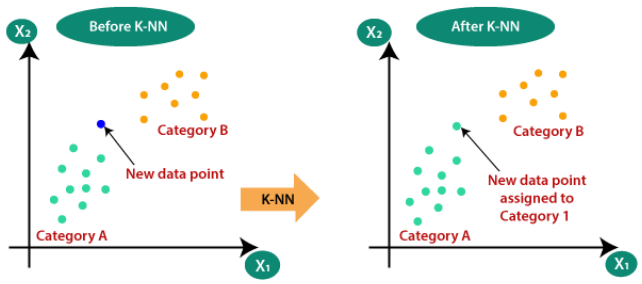

Fig.5 Explicação KNN

Neste algoritmo foi feita outra filtragem a cerca das notas da IMDB por meio de uma taxa, partindo de 0 até 10 separada em 2.5 cada trecho, correspondente com as notas reais. Além disso, foi feita uma grade de categorias para corresponder a qualidade do filme divida na Tabela 2.

TABLE II

Critério de notas		
IMDB Rating	**Categoria**	**Grade**
0.0 – 2.4	0.0	VeryBad
2.5 – 4.9	2.5	Bad
5.0 – 7.4	5.0	Good
7.5 – 9.9	7.5	Very Good
10.0	10.0	-

Além disso, foi feita uma avaliação com base no valor de K acerca de sua precisão, em um espaço de valores entre 10 a 100 (Figura 6).

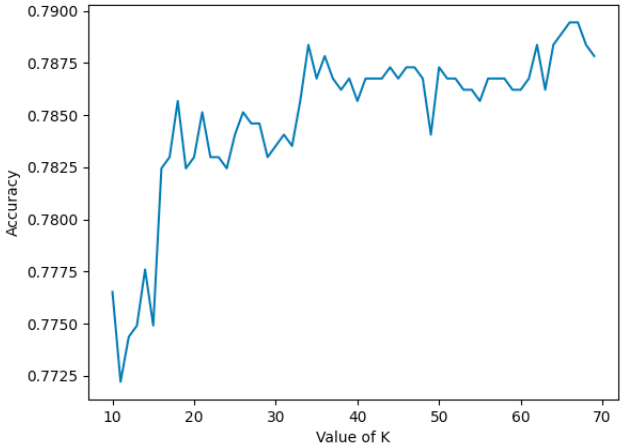

Fig.6 Comparação entre valores de K

Dessa forma, é possível visualizar que o ápice em que o valor de K correspondente a uma maior precisão seria em torno de 65, quando sua acurácia é de aproximadamente 79%. Porém o tempo de execução foi entorno de 4.741 segundos.

V. Conclusions

Os resultados obtidos no trabalho foram os contido na Tabela III. Embora houve uma performance significantemente maior em relação ao algoritmo KNN, ele teve um tempo de execução muito superior ao de Regressão Linear.

TABLE III

Algoritmo	% Precisão	Tempo Execução
Linear Regression	62.02	0.0016 sec
KNN	78.95	4.741 sec

VI. Discussion and FutureWork

Foi analisado a questão da influencia a cerca da mudança no tamanho da amostra em questão ambos os algoritmos testados. Após a analise, não houve mudança significativa na variação da variável test_size quando está sendo feita o treinamento dos dados na função train_test_split.

Ademais, a variação do valor dado para K, também não apresentou melhoras significativas além dos limites definidos na Figura 4. Correspondente ao aprendizado de maquina em respeito aos dados obtidos.

References

[1] Internet Movie Data Base. URL: https://www.imdb.com/

[2] Warda Ruheen Bristi, Zakia Zaman, Nishat Sultana, "Predicting IMDb Rating of Movies by Machine Learning techniques" URL:https://www.researchgate.net/publication/338369572_Predicting_IMDb_Rating_of_Movies_by_Machine_Learning_Techniques

[3] Wikipedia KNN Definition. URL: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[4] Wikipedia Linear Regression. URL: https://en.wikipedia.org/wiki/Linear_regression

[5] M. H. Latif and H. Afzal, "Prediction of movies popularity using machine learning techniques," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 16, no. 8, p. 127, 2016.

[6] S. Pramod, A. Joshi, and A. Mary, "Prediction of movie success for real world movie dataset," *Int. J. of Advance Res., Ideas and Innovations in Technol*, vol. 3, no. 3, 2017.

[7] Medium Web, "Predicting IMDb Ratings of New Movies" URL: https://medium.com/web-mining-is688-spring-2021/predicting-imdb-ratings-of-new-movies-2b39459fee9a

[8] TowardsDataScience Web, "Predicting IMDb Movie Ratings using Supervised Machine Learning" URL: https://towardsdatascience.com/predicting-imdb-movie-ratings-using-supervised-machine-learning-f3b126ab2ddb

[9] GitHub Web, "Predicting IMDb Ratings with Linear Regression" URL: https://github.com/josephpcowell/cowell_proj_2

[10] GitHub Web, "K-Nearest-Neighbor-IMDB-Project" URL: https://github.com/Alex-Schlee2/K-Nearest-Neighbors-IMDB-Project

[11] Analyticsvidhaya Web, "Predicting Movie Genres using NLP URL: https://www.analyticsvidhya.com/blog/2019/04/predicting-movie-genres-nlp-multi-label-classification/\

[12] Medium Web, "IMDB Movie Genre tag prediction" URL: https://www.analyticsvidhya.com/blog/2019/04/predicting-movie-genres-nlp-multi-label-classification/

[13] GitHub Web, "Python-IMDb-Ratings-Prediction" URL: https://github.com/akashagte/Python-IMDb-Ratings-Prediction

[14] Kaggle Web, "IMDB-Perform Sentiment Analysis with scikit-learn"