Exploring the relationship between user's characteristics and movie recommendations using a KNN-based recommender system

Muhao Hu[1,4,7], **Wufan Xiao**[2,5], **Yuzhang Li**[3,6]

[1]College of Sci and Engineering, University of Minnesota, Twin Cities, Minneapolis, MN, 55455, USA
[2]College of Arts and Sciences, University of Washington, Seattle, WA, 98195, USA
[3]College of Electronic Information, Tianjin University of Science and Technology No. 1038 Dagu South Road, Hexi District, Tianjin 300222 China

[4]muhaohu1023@gmail.com
[5]wufanxiao888888@gmail.com
[6]lyell0710@gmail.com
[7]corresponding author

Abstract. The central aim of this research is to elucidate the degree to which demographic variables, including but not limited to age and gender, bear on the performance of movie recommendation algorithms within film recommendation systems. Further, we endeavor to uncover any existent correlations between these user characteristics and the resultant outputs of such systems. Leveraging the expansive dataset available via MovieLens, we employ a linear regression model to ascertain the four critical variables (age, gender, occupation, and the average user rating for films previously watched) that have the most profound influence on movie recommendation algorithms. Once these salient factors have been determined, we assign their respective weights and incorporate these into a KNN algorithm. We then subject the resultant model to rigorous testing to verify the accuracy of our results and to ascertain whether the integration of these weighted elements enhances the overall precision of the movie recommendation system. While extant literature predominantly focuses on the amalgamation of KNN with other algorithms, our study charts a novel course by using linear regression. This methodology allows us to intuitively illustrate the relationship between user demographics and the movie recommendation system and enables us to evaluate whether emphasizing certain characteristics can augment the system's effectiveness. Our findings suggest that of all the user characteristics examined, the mean of users' ratings for movies previously watched exerts the greatest influence on the outputs of the movie recommendation system. Moreover, incorporating weights reflective of the average user ratings across all movie features within the KNN algorithm can significantly bolster the accuracy of the resultant movie recommendations.

Keywords: Movie Recommendation System, KNN algorithm, linear regression, Weight Based KNN Recommender System

1. Introduction

In the contemporary era of data proliferation, vast volumes of information are generated daily, rendering immediate judgment a daunting task. Movie recommendation systems emerged as a solution to this complexity, distilling raw data and presenting users with curated, processed information. Despite their utility, these systems must continuously evolve to keep pace with the dynamic nature of the information landscape. Consequently, this paper investigates the extent to which human demographic variables such as age and gender influence the outcomes of movie recommendation systems and, further, whether the integration of these weighted characteristics into a KNN recommendation algorithm can enhance the accuracy of the resulting recommendations.

Numerous studies have embraced the strategy of augmenting movie recommendation algorithms by combining two distinct methods, such as the instance of a weight-based similarity algorithm named IR-IUF++ being combined with the KNN algorithm [1]. Contrarily, our approach seeks to refine the KNN algorithm more intuitively by incorporating weights within the KNN algorithm using linear algebra techniques.

It is important to note that our research relies on a dataset sourced from several films completed in 1998 [2], which may introduce certain limitations to our findings. However, we are confident that future advancements in data collection and processing will rectify these constraints and further bolster the accuracy of our weighted KNN recommendation algorithm.

2. Literature Review

Predominantly, current research endeavors to optimize film recommendation algorithms subscribe to a paradigm of amalgamating two distinct algorithms to enhance the accuracy of recommendations. This approach is echoed in the literature titled "Weight Based KNN Recommender System" [1], where the research shares a parallel objective to our own – improving the accuracy of movie recommendations by deploying a weighted KNN algorithm. However, their approach integrates the IR-IUF++ algorithm with KNN, emphasizing similarity computation and consequently modifying the KNN algorithm considerably [3,4].

Conversely, in our study, we utilize a linear regression methodology initially to ascertain whether a linear relationship exists between user demographics and the outcomes of the movie recommendation system. Subsequently, we integrate the weighted demographic variables that demand attention into the KNN algorithm. We verify our approach by embedding these weights into the KNN algorithm to determine whether the result yields a more precise movie recommendation.

The rationale behind the adoption of linear algebra is to illustrate more intuitively the nature of the demographic features that are integrated into the KNN recommendation algorithm. Thus far, extant literature has not adequately addressed how these user characteristics influence the KNN film recommendation algorithm. Although we anticipate that future research might yield more sophisticated algorithms to encapsulate the impact of demographic features on movie recommendation outcomes, our current endeavor employs linear regression to explore the relationship between these demographic characteristics and film recommendation systems, subsequently incorporating this relationship into the KNN algorithm.

3. Methods

The data repository for our research is MovieLens, an extensive collection of viewer movie ratings, detailed film metadata, and comprehensive user demographic information curated by the University of Minnesota. The MovieLens databases have enjoyed a long-standing existence and have been leveraged in numerous academic studies, testifying to the credibility and reliability of the data source.

Firstly, data pre-processing, including cleaning and extraction, will be implemented. Unlike some movie websites, our research does not require data related to movie years. After the extraction process, linear regression will be employed to elucidate the correlation between user characteristics and movie ratings and to identify the most significant human features affecting user ratings. To pinpoint a linear relationship, we selectively focus on the top 20 highest-rated movies. The rationale behind this selection is that films rated by a limited number of users could potentially bias the linear regression. In contrast, a larger sample provides more accurate data, allowing for a better fit of the linear relationship and thereby increasing the persuasiveness of the results. For these top 20 films, linear regression will be conducted utilizing R programming software. The independent variables will comprise the user's age, occupation, gender, and the average of the user's ratings across all viewed movies. Conversely, the dependent variable will be the user's rating for each film. Through this process, a linear regression equation for a specific movie rating in relation to each user's characteristics can be derived. The coefficients from this equation will subsequently allow us to identify the user characteristics that most profoundly impact ratings. For the initial 20 films, we will establish 20 linear regression equations and document the frequency of occurrence of each significant feature separately. Ultimately, based on the frequency of these user characteristics, we can identify the user characteristics that most significantly influence user ratings in the top 20 films. Overall, linear regression can offer us an objective metric that needs to be weighted in the KNN recommendation algorithm, thereby enhancing the accuracy and efficiency of movie recommendations.

Secondly, to substantiate our conjecture, we apply the results obtained from linear regression to weight the KNN recommendation algorithm. Ultimately, the verification method integrated with R programming software is employed. The verification process involves extracting one user's information and executing the KNN algorithm using the remaining users' data. Following this, the removed user's data is inputted to obtain the predicted movie rating outcome from the KNN algorithm. This predicted rating is then juxtaposed with the actual movie rating provided by the extracted user. This procedure is conducted to verify whether the KNN algorithm, when weighted with the user characteristics, enhances the precision of the movie recommendation system. By demonstrating a closer approximation between predicted and actual ratings, the validity of our approach can be empirically established.

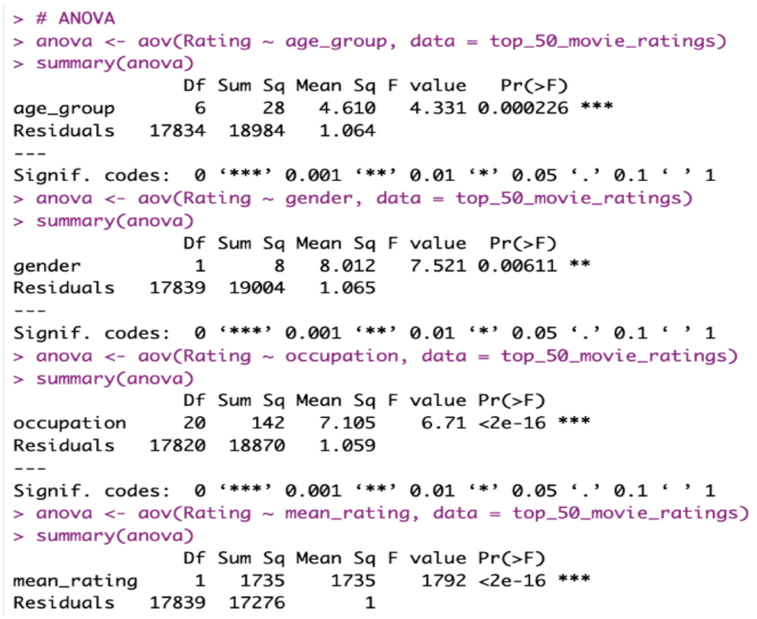
In conclusion, our study elucidates an evident influence of viewer ratings for previously watched movies on the outcomes of movie recommendation systems. The rationale behind this conclusion is that a user who frequently rates movies highly may also extend high ratings to other films. This correlation appears intuitively plausible and thus serves as the principal line of inquiry in this research. Our findings reaffirm the hypothesis that specific demographic characteristics indeed impact the precision of movie recommendation systems.

4. Results

4.1. Linearly Regression

For linear regression, we start by cleaning the data and extracting the required information. We need to quantify certain information, the basic idea being, for example, to assign a value of 1 to males and a value of 0 to females to arrive at the result that males tend to score a certain coefficient higher than females. This is what quantitative data means. After cleaning up all the data, we need to set up the linear regression equation in the R program and check the results for validity. At the same time, the R program will return a linear relationship between the user's rating of the movie and the user's characteristics. Next, we will compare the coefficients of the features, and the features with higher coefficients will be identified as the user features that have more influence on the ratings. Finally, we will use the R program to run a linear regression on the top 20 most-rated movies, counting which types of user features appear most frequently in these 20 linear relationships. Finally, we conclude that user ratings for movies they have seen are the most influential factor in movie ratings.

In the beginning, we need to identify all those user features that need to be added to the linear regression model. Logically influencing the user's final rating should be due to a number of factors. All types of user characteristics should be accounted for in the linear regression model. Still, we need to identify those user characteristics that should be accounted for in the linear regression model. For this purpose, we tested using R program. This study tested whether the four variables of user age, user gender, user occupation, and user average rating should be added to the linear regression model.



```
> # ANOVA
> anova <- aov(Rating ~ age_group, data = top_50_movie_ratings)
> summary(anova)
               Df Sum Sq Mean Sq F value   Pr(>F)    
age_group       6     28   4.610   4.331 0.000226 ***
Residuals   17834  18984   1.064                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> anova <- aov(Rating ~ gender, data = top_50_movie_ratings)
> summary(anova)
               Df Sum Sq Mean Sq F value  Pr(>F)   
gender          1      8   8.012   7.521 0.00611 **
Residuals   17839  19004   1.065                   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> anova <- aov(Rating ~ occupation, data = top_50_movie_ratings)
> summary(anova)
               Df Sum Sq Mean Sq F value Pr(>F)    
occupation     20    142   7.105    6.71 <2e-16 ***
Residuals   17820  18870   1.059                   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> anova <- aov(Rating ~ mean_rating, data = top_50_movie_ratings)
> summary(anova)
               Df Sum Sq Mean Sq F value Pr(>F)    
mean_rating     1   1735    1735    1792 <2e-16 ***
Residuals   17839  17276       1                   
```

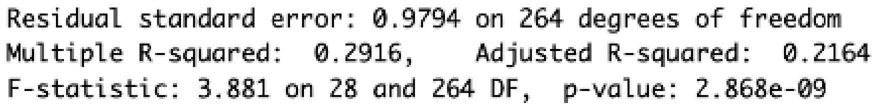
Figure 1. ANOVA summary: ANOVA summary on multivariate linear regression

From Figure 1 ANOVA summary on multivariate linear regression, we can determine that for this multivariate linear model, the p-values of all four variables are significant, so we will consider all four variables to be added to the linear regression. The main idea of linear regression is to determine if there is a linear relationship between user ratings and user characteristics. Eventually, a linear model will be determined for user characteristics and user movie ratings.

$$user's\ rating = \beta_1 \times user's\ age + \beta_2 \times user's\ gender + \beta_3 \times user's\ occupation + \beta_3 \times user's\ mean\ movie\ rating \quad (1)$$

In this model, the information of various types of users can be subdivided into many categories, so this is a multivariate linear regression model.

We will demonstrate the research process by performing linear algebra on a Movie118. First, we extract information about the movie Movie 118 and all the users who rated Movie 118. Finally, we perform data entry in the R program and view the linear regression summary about Movie 118.



```
Residual standard error: 0.9794 on 264 degrees of freedom
Multiple R-squared:  0.2916,    Adjusted R-squared:  0.2164 
F-statistic: 3.881 on 28 and 264 DF,  p-value: 2.868e-09
```

Figure 2. Result Summary: Summary of Linear Regression Results for Movie118

Figure 2 shows the summary of Linear Regression Results for Movie118. Aggregation reveals that the p-value is 2.868e-9, which is much less than 0.05, and the null hypothesis can be rejected, determining that for Movie 118, there is a certain linear relationship between user ratings and certain user characteristics. Having established that there is a linear relationship, we need to compare which user characteristics have the greatest impact. We need to compare the coefficient components to see which user characteristics have the greatest impact.

Table 1. Coefficients Summary: Coefficients of each user's characteristics for Movie118

Coefficients: Variable	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.588665	0.597887	-0.985	0.32573
age_groupYoung Adults	-0.034744	0.188339	-0.184	0.85378
age_groupEarly Thirties	-0.324917	0.234095	-1.388	0.16632
age_groupMid to Late Thirties	0.048087	0.237352	0.203	0.83961
age_groupEarly Forties	0.136131	0.232012	0.587	0.55788
age_groupLate Forties to Early Sixties	0.041664	0.295152	0.141	0.88785
age_groupSenior	0.506949	1.237004	0.41	0.68227
genderM	0.061377	0.142452	0.431	0.66692
occupationartist	-1.265508	0.453285	-2.792	0.00562**
occupationdoctor	-0.307843	0.607077	-0.507	0.61251
occupationeducator	0.44947	0.282828	1.589	0.11321
occupationengineer	0.054432	0.282599	0.193	0.84741
occupationentertainment	-0.148254	0.459036	-0.323	0.74698
occupationexecutive	0.722294	0.345896	2.088	0.03774*
occupationhealthcare	0.332979	0.449632	0.741	0.45962
occupationhomemaker	1.489892	0.607632	2.452	0.01486*
occupationlawyer	-0.125258	1.027025	-0.122	0.90302
occupationlibrarian	-0.247722	0.343868	-0.72	0.47192
occupationmarketing	1.223195	0.602119	2.031	0.04321*
occupationnone	-0.034858	0.535006	-0.065	0.9481
occupationother	0.306208	0.267068	1.147	0.2526
occupationprogrammer	0.459539	0.293437	1.566	0.11853
occupationretired	-0.35969	1.016593	-0.354	0.72376
occupationsalesman	0.598583	0.609283	0.982	0.32678
occupationscientist	0.16904	0.535814	0.315	0.75264
occupationstudent	0.008436	0.262906	0.032	0.97443
occupationtechnician	0.001834	0.389714	0.005	0.99625
occupationwriter	0.091526	0.325476	0.281	0.77877
mean_rating	1.023183	0.140269	7.294	3.51E-12***
Signif. codes: 0'***' 0.001'**' 0.01 '*' 0.05 '·' 0.1' ' 1				

By comparing the coefficients of various user characteristics, it is possible to determine which characteristics are most influential. According to fingure2, we can only have user occupation and user's average rating as more significant influences. We can analyze and find that occupation has more influencing factors and the coefficients are larger, so we can draw a preliminary conclusion that the occupation of the user is a more important influencing factor for Movie118. Perhaps we should weigh the factor of user occupation in the KNN recommendation algorithm to improve the accuracy of the movie recommendation system.

After analyzing the Movie118 example, we get the results for a movie. We repeated the above and analyzed the linear regression results for the first 20 movies and made statistics. The more dominant influences in the linear regression results for each movie were recorded.

Table 2. Coefficients Record: Record of significant coefficients of linear regression for the first 20 movies

MovieID	Distinctive User Characteristics
423	only mean
118	artist(4),mean
15	artist(2), gender, mean
151	age(3)
216	only mean
210	age(4), writer(1), mean
9	doctor(3)
176	age(1), healthcare(1), mean
204	age(1), healthcare(1), mean
1	only mean
168	writer(1), mean
117	artist(1), mean
121	artist(3), mean
748	age(1), mean
405	only mean
64	artist(1), age(2), mean
202	salesman(2), mean
288	age(3),none(2), mean
172	age(3), mean
318	entertainment(1), mean

What we can see is that for the top 20 movies, the user's average movie rating is the user characteristic that appears the most. So, we conclude that the user's average movie rating is the most important influencing factor for the movie recommender system. We will weight this user feature in the KNN recommendation algorithm.

4.2. KNN system

The k-Nearest Neighbors (KNN) algorithm is one of the simplest yet most widely used algorithms in the realm of machine learning, particularly in scenarios that require decision-making based on the entire dataset. KNN operates by finding a pre-defined number of samples closest in the distance to a new point and predicting the label from them. It is like the basic demonstration of the KNN algorithm shown in figure3. The number of samples can be a user-defined constant (k), or it can vary based on the local density of points. The distance can be any metric measure: standard Euclidean distance is the most common choice [5].

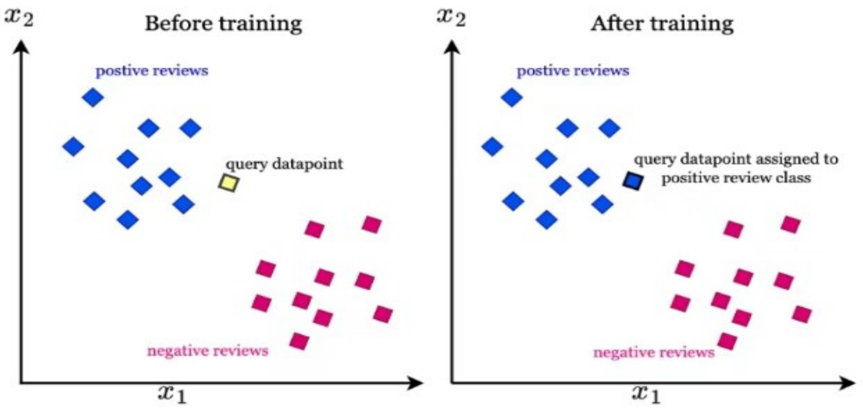


Figure 3. Basic idea for KNN algorithm: Prediction demonstration of the KNN algorithm [6]

Euclidean distance represents the shortest distance between two points in Euclidean space, which is in the following form:

$$D(P,Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2} \tag{2}$$

Where $P(p_1, p_2, \ldots, p_n)$ and $Q(q_1, q_2, \ldots, q_n)$are two points in an n-dimensional space [7]. In this paper, P and Q represent two users, and p_i ,$q_i \forall i \in \{1,2,\ldots,n\}$ represent the ratings they give for each movie. For our research, we want to add user's features, p_{n+1}, q_{n+1} to calculate a new Euclidean distance. Based on the outcomes derived from the multiple linear regression analysis, we observed that the mean rating attributed by users stands out as the most pivotal predictor for movie ratings. Consequently, we incorporated this particular feature into the KNN data frame to ascertain whether it can enhance the model's predictive accuracy [8].

Within the context of recommendation systems, KNN is often employed through a process called Collaborative Filtering (CF). There are two primary methods of CF: user-based and item-based. User-based Collaborative Filtering (UCF): UCF operates on the premise that similar users have similar preferences. In this method, for a target user, KNN identifies k users that have a similar taste to the target. Predictions are made by averaging the ratings of these k users. Item-based Collaborative Filtering (ICF): ICF, on the other hand, focuses on finding similar items based on the feedback from users. Given an item to be recommended, KNN finds k similar items that have been liked by the user in the past and makes predictions based on their ratings.

For the scope of our research, we have chosen to employ the User-based Collaborative Filtering method. This decision was predicated on the belief that users with similar behaviors and preferences in the past will likely continue to exhibit similarities in their future behaviors and preferences. UCF tends to provide more personalized recommendations as it specifically tailors the suggestions based on the behavior of similar users. Furthermore, our preliminary analysis indicated that our dataset has denser user-based similarities compared to item-based, making UCF a more suitable choice in achieving higher prediction accuracy for our specific case.

In our comprehensive dataset, we have aggregated 100,000 ratings, sourced from a distinct group of 943 users and spread across 1,682 movies. To strike a balance between the breadth of user behavior and computational feasibility, we judiciously selected our research parameters, setting n=100, k=50.

These are our following results:

Table 3. Difference Record: Record of Difference between Original KNN and Improved KNN

User ID	Movie ID	Accuracy_0	Accuracy_3	Difference
23	62	100	100	0
23	195	2	1	-1
23	32	3	8	5
23	203	74	75	1
23	82	14	18	4
23	747	90	92	2
23	381	38	48	10
23	258	0	0	0
23	131	100	100	0
23	151	12	15	3

In our experimental design, we strategically selected a user, denoted by the identifier "ID 23". Subsequently, a subset of 10 films previously rated by this user was chosen at random for analysis. To assess the predictive prowess of both the original and enhanced KNN algorithms, we conducted 100 predictive iterations for each film. The number of accurate predictions made by the original algorithm was documented as 'Accuracy_0', while those made by the improved algorithm were cataloged as 'Accuracy_3'. The differential measure, termed as 'difference', represents the disparity in the accurate predictions made by the two algorithms over the 100 iterations.

5. Discussion

The results presented through R programming, the first multivariate linear regression, helped us understand which of the user characteristics we studied were worthy of being considered in the final linear model. For the analysis of ANOVA, it was concluded that the user's gender, age, occupation, and average movie rating should be considered in the final multivariate linear model. After understanding the general linear model, we obtained a linear regression model for each movie through the R program. In Table 1, we can learn the linear regression model for the first 20 movies and record the number of occurrences of dominant features. By recording, we can get that the feature of the average user rating for the movie appeared 18 times, the user's age feature appeared 8 times, the user's occupation feature appeared 13 times, and the user's gender feature appeared 1 time. From the results, the user’s average movie rating is the most significant influencing factor. Logically, we can also explain the result that gender is not significant, which means that with the progress of time, gender equality has become more and more popular, and gender is no longer the main factor influencing the result. So, to a certain extent, such results are also in line with the thinking of today's society. For the results of linear regression, we get the conclusion that the average movie rating of users has a greater impact on the results of the movie recommendation system. We should weight this user characteristic in the KNN recommendation algorithm to verify whether it improves the accuracy of the movie recommendation system.

There are some limitations to this experiment, starting with the incompleteness of the data. MovieLens' data is from 1998, and more complete data is needed for modern predictions. Second, for the linear regression model, the data is still not perfect, and even for the top 20 most-rated movies, there are cases where some users did not rate them. This also leads to some errors in the linear regression model.

In the realm of collaborative filtering and recommendation systems, user attributes play a pivotal role in refining and personalizing suggestions. When deploying the k-Nearest Neighbors algorithm for our user-based collaborative filtering approach, our methodology revolved around the fundamental idea: richer user features can potentially augment the accuracy of the KNN predictions.

In our study, we specifically focused on user ID 23 as a test case. After randomly selecting 10 movies from this user's viewing history, we ran predictions 100 times for each movie, comparing the outcomes of the original KNN algorithm with our enhanced KNN method. Over the course of these trials, according to Table 2, we observed that while the enhanced KNN often provided more accurate predictions, there were instances where no noticeable difference was discerned, or in some rare cases, the enhanced model even lagged behind slightly. Despite these occasional discrepancies, the overall trend supported our initial hypothesis that integrating an enhanced KNN could potentially elevate the prediction accuracy. Nevertheless, we acknowledge some limitations in our experimental design. The 'k' and 'n' values, integral to the KNN algorithm, were arbitrarily chosen without systematic optimization, which could impact the reliability of our results. Additionally, while 100 prediction repetitions provided us with a substantial dataset, it might not be exhaustive enough to cover all possible prediction variances. A comprehensive evaluation would necessitate a broader study, optimally tuning parameters, and incorporating more users to ensure that our findings are consistent and widely applicable.

Our results substantiate this hypothesis. Introducing the `meanRating`—an aggregate measure of each user's typical rating behavior—into the KNN's input dataframe emerged as a salient improvement. The logic here is straightforward yet profound: while two users might share preferences for similar movies, their inherent rating patterns (i.e., their propensities to rate movies higher or lower on average) can differ. Capturing this user-specific 'bias' through the mean rating provides our KNN model with an additional dimension of personalization, enhancing prediction accuracy. However, KNN is sensitive to the curse of dimensionality. As we incorporate more features, the distance calculations between data points in multi-dimensional space become more taxing. This leads to elongated run times and potentially introduces noise, making it imperative to ensure that any added feature truly contributes to enhancing predictive accuracy. Our decision to incorporate `meanRating` was backed by tangible accuracy improvements, but one should always tread cautiously when enlarging the feature space.

Looking forward, there's vast untapped potential in exploring the item-based paradigm of KNN, like our approach. Just as user attributes—like `meanRating`—enhanced our user-based collaborative filtering, movie attributes can be harnessed to refine item-based collaborative filtering. Features such as genre, director, actor, release year, and even metadata from movie reviews could be leveraged. The challenge, much like what we confronted, will be in discerning which attributes meaningfully enhance predictions and which merely amplify noise and computational demands. Moreover, as computational efficiency emerges as a concern with growing dimensions, researchers could consider dimensionality reduction techniques or opt for more scalable algorithms as alternatives to traditional KNN. In summary, while our research paves the way for a more nuanced use of user-based KNN in recommendation systems, the journey has only begun. The intersection of feature engineering, algorithmic efficiency, and predictive accuracy remains a fertile ground for future exploration.

Maybe in the future, there will be more perfect data that can help us to make a better linear regression model or a more perfect KNN weighting method.

6. Conclusion

Our research has enriched our understanding of the influence of user characteristics on the accuracy of movie recommendation systems. We have identified that the average movie rating, a critical manifestation of user characteristics, plays a crucial role in enhancing the accuracy of the k-Nearest Neighbors (KNN) predictions. Our results demonstrate that while two users may prefer similar movies, their rating patterns can vary. Capturing this user-specific 'bias' via average rating provides an additional dimension of personalization to our KNN model, thereby improving prediction accuracy.

However, our research is not without limitations. Firstly, there is an inadequacy of data completeness. The MovieLens data we utilized dates back to 1998, which is not sufficient for modern predictions. Secondly, even among the top 20 most-rated movies, there are instances where some users have not rated them, leading to some inaccuracies in the linear regression model.

Importantly, we had to grapple with a significant trade-off as we expanded our feature set: computational complexity. KNN is inherently sensitive to the curse of dimensionality. With more features incorporated, distance calculations between data points in multi-dimensional space become more computationally intensive, leading to prolonged run times and the potential introduction of noise. Therefore, caution must be exercised when enlarging the feature space.

Looking ahead, there is significant untapped potential in exploring the item-based paradigm of KNN. Just as user attributes like “meanRating” enhanced our user-based collaborative filtering, movie attributes can be leveraged to refine item-based collaborative filtering. Such attributes could include genre, director, actors, release year, and even metadata from movie reviews. The challenge, however, lies in discerning which attributes meaningfully enhance predictions and which merely amplify noise and computational demands. Additionally, as computational efficiency becomes a concern with growing dimensions, researchers may consider dimensionality reduction techniques or opt for more scalable algorithms as alternatives to traditional KNN.

In summary, while our research lays a foundation for the more nuanced application of user-centric KNN in recommendation systems, it is merely the start. The intersection of feature engineering, algorithmic efficiency, and predictive accuracy remains a fertile ground for future exploration.

Acknowledgement

Muhao Hu, Wufan Xiao, and Yuzhang Li contributed equally to this work and should be considered co-first authors.

References

[1] Wang, B., Liao, Q., & Zhang, C. (2013, August). Weight based KNN recommender system. In 2013 5th International Conference on Intelligent Human-Machine Systems and cybernetics (Vol. 2, pp. 449-452). IEEE.

[2] GroupLens. (1998). MovieLens 100K Dataset [Data set] https://grouplens.org/datasets/movielens/100k/

[3] Subramaniyaswamy, V., & Logesh, R. (2017). Adaptive KNN based recommender system through mining of user preferences. Wireless Personal Communications, 97, 2229-2247.

[4] Jahrer, M., Töscher, A., & Legenstein, R. (2010, July). Combining predictions for accurate recommender systems. In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 693-702).

[5] Zhang, Z. (2016). Introduction to machine learning: k-nearest neighbors. Annals of translational medicine, 4(11).

[6] H, R. S. (2023a, April 5). K-nearest neighbors algorithm. Intuitive Tutorials. https://intuitivetutorial.com/2023/04/07/k-nearest-neighbors-algorithm/

[7] Bahrani, P., Minaei-Bidgoli, B., Parvin, H., Mirzarezaee, M., & Keshavarz, A. (2023). A new improved KNN-based recommender system. Theournal of Supercomputing, 1-35.

[8] Jingwen, Z. (2017). R algorithm for MovieLens [Source code]. GitHub.https://github.com/jingwen-z/R/blob/master/algorithm/MovieLens.R