

# Norms, Angles, and Your Movie Choices

Rahma Okta Feriska (10121050),      Audie Quisha Jerome Tampubolon (10122060),      Davina Sybilla Bahrudin  
(10122068)  
Institut Teknologi Bandung

## CONTENTS

<b>I</b>	<b>Introduction</b>	2
<b>II</b>	<b>Mathematical Background</b>	2
II-A	Pearson Correlation Coefficient . . . . .	2
II-B	Jarak Korelasi: . . . . .	2
II-C	Jarak Euclidean: . . . . .	3
<b>III</b>	<b>Problem Definition</b>	3
<b>IV</b>	<b>Proposed Solutions</b>	3
IV-A	Penyortiran Data . . . . .	3
IV-B	Perbandingan Menggunakan Jarak Euclidean . . . . .	3
IV-C	Perbandingan Perhitungan Korelasi Pearson . . . . .	3
IV-D	Rekomendasi Film Hasil Perbandingan . . . . .	4
IV-E	Rekomendasi Film Rating Pribadi . . . . .	4
IV-F	Film yang Disukai . . . . .	4
<b>V</b>	<b>Analysis and Interpretation</b>	4
V-A	Penyortiran Data . . . . .	4
V-B	Perbandingan Menggunakan Jarak Euclidean . . . . .	4
V-B1	Perhitungan Nilai Jarak Euclidean . . . . .	4
V-B2	Perbandingan Nilai Jarak Euclidean . . . . .	4
V-C	Perbandingan Perhitungan Korelasi Pearson . . . . .	4
V-C1	Perhitungan Nilai Koefisien Korelasi Pearson . . . . .	4
V-C2	Perbandingan Nilai Korelasi Pearson . . . . .	4
V-D	Rekomendasi Film Hasil Perbandingan . . . . .	5
V-E	Rekomendasi Film Rating Pribadi . . . . .	5
V-E1	Jarak Euclidean . . . . .	5
V-E2	Korelasi Pearson . . . . .	5
V-E3	Hasil Rekomendasi Menurut Rating Pribadi . . . . .	5
V-E4	Daftar Film yang Disukai . . . . .	6
<b>VI</b>	<b>Conclusions and Recommendations</b>	6
VI-A	Kesimpulan . . . . .	6
VI-B	Rekomendasi . . . . .	6
<b>Appendix A: Variabel yang Digunakan</b>		7
<b>Appendix B: Kode MATLAB</b>		7
<b>Appendix C: Data</b>		9
<b>References</b>		9

## LIST OF FIGURES

1	Kode untuk pertanyaan nomor 1 . . . . .	7
2	Kode untuk pertanyaan nomor 2 . . . . .	7
3	Kode untuk pertanyaan nomor 3 . . . . .	7
4	Kode untuk pertanyaan nomor 4 . . . . .	7
5	Kode untuk pertanyaan nomor 5 . . . . .	7
6	Kode untuk pertanyaan nomor 6 . . . . .	8
7	Kode untuk pertanyaan nomor 7 . . . . .	8
8	Kode untuk pertanyaan nomor 8 . . . . .	8
9	Kode untuk pertanyaan nomor 9 . . . . .	8
10	Kode untuk pertanyaan nomor 10 . . . . .	8
11	Kode untuk pertanyaan nomor 11 . . . . .	8
12	Kode untuk pertanyaan nomor 12 . . . . .	8
13	Kode untuk pertanyaan nomor 13 . . . . .	9

## LIST OF TABLES

I	Variabel Syntax MATLAB . . . . .	7
---	----------------------------------	---

# Norms, Angles, and Your Movie Choices

**Abstract**—Salah satu aplikasi dari Aljabar Linear dalam kehidupan nyata adalah pengembangan sistem rekomendasi film untuk suatu pengguna pada suatu platform hiburan. Sistem tersebut menerapkan konsep jarak Euclidean dan koefisien korelasi Pearson. Jarak Euclidean mengukur kesamaan preferensi berdasarkan perbedaan rating antara pengguna, sementara koefisien korelasi Pearson menilai kekuatan hubungan antara pola rating pengguna. Dengan menghitung jarak Euclidean, akan diambil nilai terkecil sebagai asumsi similaritas selera antara dua pengguna. Sedangkan perhitungan menggunakan koefisien korelasi Pearson mengandalkan nilai perhitungan dengan angka mendekati 1 yang menandakan adanya hubungan positif antara dua objek perbandingan. Dengan melihat hasil perhitungan ini, akan diambil data film pengguna dengan similaritas selera, sehingga data pengguna tersebut dapat dijadikan rekomendator untuk pengguna lainnya.

**Kata kunci** : Rekomendasi film, jarak Euclidean, koefisien korelasi Pearson, aljabar linear.

## I. INTRODUCTION

Dalam era digital saat ini, analisis data telah menjadi landasan bagi berbagai layanan internet yang berfokus pada kepuasan pengguna. Di antara berbagai aplikasi analisis data, prediksi preferensi pengguna adalah aspek yang sangat vital, terutama bagi platform hiburan seperti Netflix, Pandora, dan Amazon. Melalui penggunaan operasi Aljabar Linear, platform-platform ini dapat memahami preferensi individu dan menyajikan rekomendasi yang sangat dipersonalisasi kepada pengguna mereka.

Salah satu contoh yang menonjol adalah Proyek Music Genome oleh Pandora, yang menggunakan pendekatan matematis untuk menganalisis preferensi musik pengguna dan merekomendasikan lagu-lagu yang cocok. Konsep ini mengilhami banyak penelitian lain dalam domain ini, termasuk penerapan Aljabar Linear dalam prediksi preferensi pengguna untuk film.

Dalam laporan ini, kami akan mengeksplorasi penggunaan Aljabar Linear untuk memprediksi preferensi pengguna dalam konteks penggunaan platform hiburan. Fokus utama kami adalah pada aplikasi Aljabar Linear dalam sistem rekomendasi film, dengan menggunakan data dari basis data MovieLens yang mencakup penilaian dari ribuan pengguna terhadap ribuan film.

Dengan menggabungkan teknik-teknik seperti jarak Euclidean, perkalian skalar standar, dan koefisien korelasi, kami berharap dapat menyajikan pandangan yang mendalam tentang bagaimana prinsip-prinsip matematis ini dapat membantu memahami preferensi pengguna dan menyediakan rekomendasi yang relevan.

Dengan demikian, laporan ini tidak hanya akan memberikan wawasan tentang penggunaan Aljabar Linear dalam analisis data pengguna, tetapi juga memberikan gambaran ten-

tang potensi aplikasi praktisnya dalam pengembangan sistem rekomendasi yang efektif dan tepat sasaran.

## II. MATHEMATICAL BACKGROUND

Untuk membantu menyelesaikan masalah tersebut, dapat dilakukan pendekatan secara matematis. Penggunaan Aljabar Linear merupakan alternatif untuk memberikan rekomendasi film yang mengacu pada konsep perbandingan data. Data yang dimiliki akan dihitung menggunakan konsep Aljabar Linear untuk nantinya dibandingkan sehingga dapat memberikan rekomendasi yang paling sesuai atau paling mendekati selera penggunaannya. Konsep ini menggunakan beberapa persamaan seperti Korelasi Pearson dan Jarak Euclidean.

### A. Pearson Correlation Coefficient

Koefisien korelasi Pearson digunakan untuk mengukur seberapa erat hubungan linier antara dua variabel. Untuk dua vektor data  $X$  dan  $Y$  dengan  $n$  elemen, koefisien korelasi Pearson  $r$  didefinisikan sebagai:

$$r = \frac{\sum_1^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_1^n (X_i - \bar{X})^2} \sqrt{\sum_1^n (Y_i - \bar{Y})^2}} \quad (1)$$

di mana  $\bar{X}$  dan  $\bar{Y}$  adalah rata-rata dari vektor  $X$  dan  $Y$ , secara berturut-turut. Koefisien korelasi Pearson memiliki rentang nilai antara -1 hingga 1. Jika koefisien korelasi bernilai positif maka kedua variabel memiliki hubungan yang linear positif. Jika koefisien korelasi bernilai nol maka tidak ada korelasi antara kedua variabel. Dan apabila koefisien korelasi bernilai negatif maka kedua variabel memiliki hubungan linear negatif.

Dalam perhitungan koefisien korelasi Pearson, terdapat asumsi yang harus dipenuhi, di antaranya:

- 1) Terdapat hubungan linier antara  $X$  dan  $Y$  (Asumsi linieritas)
- 2) Data berdistribusi normal (Asumsi normalitas)
- 3) Variabel  $X$  dan  $Y$  bersifat simetris, artinya tidak ada variabel yang berperan sebagai variabel bebas atau tergantung; keduanya dianggap setara dan saling mempengaruhi secara linier.
- 4) *Sampling representative*
- 5) Kedua variabel memiliki variansi yang sama

### B. Jarak Korelasi:

Jarak korelasi antara dua vektor  $X$  dan  $Y$  dengan  $n$  elemen dapat dihitung menggunakan koefisien korelasi Pearson. Dalam konteks proyek ini, kita menggunakan jarak korelasi sebagai ukuran kesamaan antara vektor rating yang diberikan oleh *trial user* dengan vektor rating yang diberikan oleh pengguna lain yang sudah ada dalam database.

### C. Jarak Euclidean:

Jarak Euclidean adalah matriks jarak yang paling umum digunakan dalam ruang Euclidean. Dalam konteks ruang Euclidean, jarak antara dua titik P dan Q dalam dimensi n dapat dihitung menggunakan rumus sebagai berikut

$$d(P, Q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (2)$$

dimana  $P = (p_1, p_2, \dots, p_n)$  dan  $Q = (q_1, q_2, \dots, q_n)$  adalah koordinat dari dua titik dalam ruang euclidean. Jarak Euclidean digunakan untuk mengukur seberapa mirip preferensi *trial user* dengan pengguna lain dalam database. Semakin kecil jarak Euclidean artinya semakin mirip preferensi film yang mereka suka. Hal ini yang digunakan dalam menghasilkan rekomendasi film yang dipersonalisasi.

### III. PROBLEM DEFINITION

Fokus utama dalam proyek ini adalah bagaimana mengaplikasikan prinsip-prinsip Aljabar Linear untuk memprediksi preferensi pengguna, sebuah area yang vital dalam bisnis bagi layanan internet seperti Netflix, Pandora, dan Amazon. Platform-platform tersebut secara cermat menganalisis preferensi konsumen untuk menyajikan produk yang cocok dengan keinginan mereka. Sebagai contoh, Proyek Music Genome oleh Pandora menggunakan pendekatan matematis dengan menggunakan vektor yang mengandung 450 komponen kategori lagu untuk merekomendasikan lagu-lagu yang cocok untuk diputar oleh pengguna selanjutnya.

Dengan memanfaatkan struktur data dalam bentuk vektor, kita dapat menggambarkan untuk menemukan kemiripan dan perbandingan antar data. Proyek ini akan mengadopsi konsep Aljabar Linear untuk mengevaluasi kemiripan selera dalam basis data MovieLens, yang terdiri dari sekitar 1 juta penilaian film dari 3.952 film yang dilakukan oleh 6.040 pengguna. Keluaran algoritma ini diharapkan dapat menghasilkan sistem rekomendasi yang sederhana dengan membandingkan preferensi pengguna. Teknik-teknik yang akan digunakan termasuk perhitungan jarak Euclidean, perkalian skalar standar, dan koefisien korelasi.

### IV. PROPOSED SOLUTIONS

Untuk memberikan rekomendasi yang tepat kepada *trial user*, perlu dilakukan perbandingan antara selera film *trial user* dengan rating film yang diberikan oleh pengguna lain. Similaritas dari kedua hal tersebut dapat diketahui dengan metode Aljabar Linier. Dalam hal ini, metode yang digunakan dengan menghitung jarak Euclidean dan koefisien korelasi Pearson.

#### A. Penyortiran Data

Data yang digunakan berisi penilaian 6.040 pengguna terhadap 3.952 film, sehingga perlu dilakukan penyortiran data untuk menentukan data yang akan digunakan sebagai pembandingan sebelum pada akhirnya memberikan kesimpulan berupa rekomendasi film kepada pengguna *trial*.

Penilaian yang diberikan pengguna dilakukan dengan memberikan angka pada range 0-5, di mana angka 0 berarti pengguna belum memberikan penilaian terhadap film tersebut, angka 1 menandakan pengguna tidak begitu menyukai film tersebut, hingga angka 5 yang menandakan bahwa pengguna amat menyukai film tersebut.

Penyortiran data ini secara spesifik dilakukan dengan menentukan 20 film terpopuler dan memilah pengguna yang memberikan rating kepada 20 film tersebut dengan nilai kisaran 1 hingga 5. Perlu diketahui bahwa pemisahan pengguna dengan penilaian angka 0 pada setidaknya salah satu dari ke-20 film menandakan bahwa pengguna belum memberikan penilaian, sehingga hasil perhitungan tidak akan relevan.

#### B. Perbandingan Menggunakan Jarak Euclidean

Penilaian pengguna yang dihasilkan setelah penyortiran data kemudian dibandingkan menggunakan metode jarak Euclidean dengan rumus sebagai berikut

$$d(P, Q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (3)$$

dimana  $P = (p_1, p_2, \dots, p_n)$  dan  $Q = (q_1, q_2, \dots, q_n)$  adalah koordinat dari dua titik dalam ruang Euclidean. P merupakan penilaian yang diberikan oleh *trial user* sedangkan Q merupakan penilaian yang diberikan oleh pengguna lainnya. Mengukur kemiripan berdasarkan jarak Euclidean memungkinkan kita untuk menilai vektor 20 dimensi yang berisi data pengguna dan rating yang diberikan. Hasil pengukuran ini dijadikan landasan untuk memilih pengguna dengan jarak Euclidean terkecil terhadap *trial user*.

#### C. Perbandingan Perhitungan Korelasi Pearson

Sebagai pembandingan dan penentu keefektifan metode yang dilakukan, perlu adanya pengukuran menggunakan konsep lain. Maka dari itulah diperlukan adanya perhitungan menggunakan koefisien korelasi Pearson yang didefinisikan sebagai berikut

$$r = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}} \quad (4)$$

Persamaan ini digunakan untuk mengukur keeratan hubungan linier antara dua variabel di mana X merupakan data penilaian dari *trial user* dan Y merupakan data penilaian dari pengguna lainnya, dan data berupa vektor dengan n elemen. Koefisien korelasi Pearson ini memiliki rentang nilai antara -1 hingga 1.

Untuk memenuhi asumsi normalitas yang harus dipenuhi dalam perhitungan korelasi Pearson, maka data yang digunakan akan disentralisasi terlebih dahulu sebelum dihitung menggunakan rumus yang diberikan. Sentralisasi ini dilakukan untuk menetralkan rating objektif yang diberikan oleh pengguna. Seperti misalnya, pengguna A sebenarnya menyukai film X, namun ia memberikan rating 3 karena standar penilaiannya yang cukup rumit. Sedangkan pengguna B juga memiliki rasa

suka yang sama seperti pengguna A terhadap film X, namun memberikan rating 5. Sentralisasi data ini akan meminimalisir pengaruh dari rating objektif tersebut sehingga menghasilkan data rekomendasi yang lebih akurat dibandingkan saat desentralisasi tidak dilakukan.

Melakukan perhitungan koefisien korelasi Pearson antara baris-baris matriks *ratings* dan vektor *trial user* ini menggunakan metode sintaks perulangan `for... end`. Hasil perhitungan dari tiap pengguna ini nantinya disimpan sebagai anggota vektor Pearson. Hasil pengukuran ini dijadikan landasan untuk memilih pengguna dengan nilai korelasi Pearson yang paling mendekati angka 1 terhadap *trial user*.

#### D. Rekomendasi Film Hasil Perbandingan

Setelah menghitung menggunakan jarak Euclidean dan korelasi Pearson, kemiripan hasil perhitungan menggunakan kedua metode ini kemudian akan dibandingkan. Hasil perbandingan ini dapat menjadi acuan sebelum memberikan rekomendasi film kepada *trial user* secara final.

#### E. Rekomendasi Film Rating Pribadi

Sebagai objek perbandingan, analisis rekomendasi film juga kami lakukan dengan membandingkan rating pribadi terhadap rating pengguna lain. Metode perhitungan yang digunakan tetap sama, yaitu menggunakan jarak Euclidean dan korelasi Pearson. Hasil akhir perbandingan ini juga sama, yaitu kedua metode menghasilkan hasil rekomendasi berbeda.

#### F. Film yang Disukai

Film yang disukai adalah film yang diberikan rating sempurna yakni 5.

### V. ANALYSIS AND INTERPRETATION

#### A. Penyortiran Data

Penyortiran data 20 film dengan penilaian terbaik yang akan dibandingkan yaitu:

- 1) E.T. the Extra-Terrestrial (1982)
- 2) Star Wars Episode IV - A New Hope (1977)
- 3) Star Wars Episode V - The Empire Strikes Back (1980)
- 4) Star Wars Episode VI - Return of the Jedi (1983)
- 5) Jurassic Park (1993)
- 6) Saving Private Ryan (1998)
- 7) Terminator 2
- 8) Matrix, The (1999)
- 9) Back to the Future (1985)
- 10) Silence of the Lambs, The (1991)
- 11) Star Wars Episode I - The Phantom Menace (1999)
- 12) Raiders of the Lost Ark (1981)
- 13) Fargo (1996)
- 14) Sixth Sense, The (1999)
- 15) Braveheart (1995)
- 16) Shakespeare in Love (1998)
- 17) Princess Bride, The (1987)
- 18) Schindler's List (1993)
- 19) Shawshank Redemption, The (1994)

#### 20) Groundhog Day (1993)

Setelah menentukan 20 film yang akan dibandingkan, kita menentukan data rating yang diberikan pengguna untuk dibandingkan dengan rating yang diberikan oleh *trial user*. Data ini ditentukan dengan menghilangkan data yang memiliki datum 0, yang memiliki arti bahwa terdapat setidaknya 1 dari 20 film yang belum diberikan rating oleh pengguna lain. Data yang telah disortir berupa rating yang diberikan oleh pengguna terhadap 20 film terpilih dengan range 1-5.

#### B. Perbandingan Menggunakan Jarak Euclidean

1) *Perhitungan Nilai Jarak Euclidean*: Jarak Euclidean antara rating yang diberikan oleh *trial user* dan pengguna lainnya dihitung menggunakan rumus jarak Euclidean. Melalui perhitungan tersebut, jarak Euclidean yang didapatkan berada pada kisaran:

$$5.65685424949238 \leq d(P, Q) \leq 10.3923048454133$$

Pengukuran ini memudahkan kita untuk melakukan perbandingan berdasarkan jarak Euclidean masing-masing data.

2) *Perbandingan Nilai Jarak Euclidean*: Hasil perhitungan yang beragam menunjukkan adanya perbedaan hubungan antara penilaian yang diberikan oleh masing-masing pengguna. Semakin kecil jarak Euclidean antara kedua data tersebut, maka selera yang dimiliki antara *trial user* dan pengguna lainnya semakin mirip, sehingga memungkinkan kita untuk menentukan film lain untuk direkomendasikan kepada *trial user*.

Melalui hasil perhitungan, diketahui bahwa pengguna dengan jarak Euclidean terkecil memiliki nilai 5.65685424949238. Di mana film lain dengan rating 5 yang diberikan oleh pengguna ini yang akan dijadikan rekomendasi kepada *trial user*. Secara spesifik, sistem rekomendasi film tersebut menggunakan asumsi bahwa *trial user* dan pengguna lain tersebut memiliki kesamaan selera film.

#### C. Perbandingan Perhitungan Korelasi Pearson

1) *Perhitungan Nilai Koefisien Korelasi Pearson*: Perhitungan menggunakan korelasi Pearson dihitung dengan menggunakan rumus korelasi Pearson. Melalui perhitungan tersebut, didapatkan range koefisien korelasi Pearson antara matriks rating pengguna yang telah disentralisasi dengan vektor rating *trial user* yang telah disentralisasi:

$$-0.369587438532043 \leq r \leq 0.628961556962687$$

Pengukuran ini mempermudah kita untuk membandingkan setiap data berdasarkan korelasi Pearson yang terkait.

2) *Perbandingan Nilai Korelasi Pearson*: Hasil perhitungan korelasi Pearson berkisar pada angka -1 hingga 1, di mana seperti yang sudah dijabarkan sebelumnya, angka -1 menandakan adanya relasi negatif, angka 0 menandakan tidak adanya relasi, sedangkan angka 1 menandakan adanya relasi positif antara dua objek. Oleh karena itu, nilai korelasi Pearson yang mendekati 1 merupakan hasil perhitungan yang kita perlukan. Karena angka tersebut menandakan adanya hubungan yang

positif, di mana dapat diartikan sebagai kedua pengguna memiliki selera film yang mirip.

Melalui hasil perhitungan, diketahui bahwa pengguna dengan nilai korelasi Pearson paling mendekati 1 yakni 0.628961556962687. Di mana film lain dengan rating 5 yang diberikan oleh pengguna ini yang akan dijadikan rekomendasi kepada *trial user*. Secara spesifik, sistem rekomendasi film tersebut menggunakan asumsi bahwa *trial user* dan pengguna lain tersebut memiliki kesamaan selera film.

#### D. Rekomendasi Film Hasil Perbandingan

Berdasarkan hasil perbandingan jarak Euclidean, film yang direkomendasikan untuk *trial user* adalah:

- Pulp Fiction (1994)
- Deer Hunter, The (1978)
- Red Violin, The (Le Violon rouge) (1998)
- Sixth Sense, The (1999)
- Children of Paradise (Les enfants du paradis) (1945)
- Being John Malkovich (1999)

Sedangkan berdasarkan hasil perbandingan korelasi Pearson, film yang direkomendasikan untuk *trial user* adalah:

- Taxi Driver (1976)
- Schindler's List (1993)
- Fargo (1996)
- Godfather, The (1972)
- North by Northwest (1959)
- Casablanca (1942)
- Citizen Kane (1941)
- Mr. Smith Goes to Washington (1939)
- Bonnie and Clyde (1967)
- Bob Roberts (1992)
- Paris Is Burning (1990)
- 12 Angry Men (1957)
- To Kill a Mockingbird (1962)
- Title not available
- Grand Day Out, A (1992)
- Raging Bull (1980)
- Annie Hall (1977)
- Stand by Me (1986)
- Killing Fields, The (1984)
- My Life as a Dog (Mitt liv som hund) (1985)
- Tickle in the Heart, A (1996)
- Boys, Les (1997)
- There's Something About Mary (1998)
- On the Waterfront (1954)
- Ordinary People (1980)
- Chariots of Fire (1981)
- Rain Man (1988)
- Saving Private Ryan (1998)
- Life Is Beautiful (La Vita è bella) (1997)
- Risky Business (1983)
- Brief Encounter (1946)
- Shower (Xizhao) (1999)

#### E. Rekomendasi Film Rating Pribadi

Kelompok kami memberikan rating pribadi berdasarkan 20 film terpopuler yang dipilih untuk dibandingkan. Melalui rating pribadi tersebut, perhitungan yang dilakukan memberikan hasil sebagai berikut:

1) *Jarak Euclidean*: Menggunakan perhitungan jarak Euclidean antara matriks rating pengguna dengan vektor rating pribadi, hasil perhitungan berada pada range:

$$8.60232526704263 \leq d(P, Q) \leq 12.9228479833201$$

Data pengguna yang kami jadikan sebagai objek rekomendasi merupakan pengguna dengan nilai jarak Euclidean terkecil, yaitu 8.60232526704263.

2) *Korelasi Pearson*: Menggunakan perhitungan korelasi Pearson antara matriks rating pengguna yang telah disentralisasi dengan vektor rating pribadi yang telah disentralisasi, hasil perhitungan berada pada range:

$$-0.424548847124009 \leq r \leq 0.5247480135004$$

Data pengguna yang kami jadikan sebagai objek rekomendasi merupakan pengguna dengan nilai korelasi Pearson yang paling mendekati angka 1, yaitu 0.5247480135004.

3) *Hasil Rekomendasi Menurut Rating Pribadi*: Berdasarkan hasil perbandingan jarak Euclidean, film yang direkomendasikan sesuai dengan rating pribadi kami adalah:

- Seven (Se7en) (1995)
- Usual Suspects, The (1995)
- Pulp Fiction (1994)
- Fargo (1996)
- Reservoir Dogs (1992)
- Third Man, The (1949)
- Star Maps (1997)
- Welcome To Sarajevo (1997)
- Out of Sight (1998)
- Run Lola Run (Lola rennt) (1998)
- American Beauty (1999)
- Gladiator (2000)
- Almost Famous (2000)
- Requiem for a Dream (2000)

Sedangkan berdasarkan hasil perbandingan korelasi Pearson, film yang direkomendasikan sesuai dengan rating pribadi kami adalah:

- Usual Suspects, The (1995)
- Rumble in the Bronx (1995)
- Shawshank Redemption, The (1994)
- Mask, The (1994)
- Naked Gun 33 1/3 (1996)
- Godfather, The (1972)
- Supercop (1992)
- Vertigo (1958)
- Maltese Falcon, The (1941)
- Monty Python's Life of Brian (1979)
- Monty Python and the Holy Grail (1974)
- Ran (1985)
- Unforgiven (1992)

- Big Sleep, The (1946)
- Zeus and Roxanne (1997)
- Truth or Consequences, N.M. (1997)
- Innocent Sleep, The (1995)
- Star Maps (1997)
- Mr. Nice Guy (1997)
- Rush Hour (1998)
- Ronin (1998)
- Lock, Stock, and Two Smoking Barrels (1998)
- Twin Dragons (Shuang long hui) (1992)
- Austin Powers
- Airplane! (1980)
- Operation Condor (Feiying gaiwak) (1990)
- Operation Condor 2 (Longxiong hudi) (1990)
- JFK (1991)
- Gladiator (2000)
- Shanghai Noon (2000)
- Naked Gun 2 1/2

4) *Daftar Film yang Disukai*: Film yang disukai oleh *trial user* adalah :

- Star Wars Episode IV - A New Hope (1977)
- Star Wars Episode V - The Empire Strikes Back (1980)
- Star Wars Episode VI - Return of the Jedi (1983)
- Matrix, The (1999)
- Silence of the Lambs, The (1991)
- Raiders of the Lost Ark (1981)
- Groundhog Day (1993)

Sedangkan film yang disukai oleh kami berdasarkan rating pribadi yang kami berikan adalah :

- Star Wars Episode IV - A New Hope (1977)
- Star Wars Episode I - The Phantom Menace (1999)
- Raiders of the Lost Ark (1981)
- Schindler's List (1993)

## VI. CONCLUSIONS AND RECOMMENDATIONS

Dari hasil perhitungan dan perbandingan yang dilakukan, terdapat kesimpulan yang dapat diambil sebagai luaran dari masalah Aljabar Linear yang diobservasi. Selain itu terdapat pula rekomendasi mengenai penggunaan masalah Aljabar Linear tersebut dalam konteks rekomendasi kepada pengguna.

### A. Kesimpulan

Projek ini menunjukkan bahwa penggunaan konsep Aljabar Linear yakni jarak Euclidean dan koefisien korelasi Pearson untuk memberikan rekomendasi film favorit kepada *trial user* berdasarkan penilaian pengguna lain memberikan hasil yang berbeda tergantung pada metrik yang digunakan. Selain itu, dapat disimpulkan bahwa:

- 1) Terjadinya perbedaan rekomendasi film dikarenakan metode jarak Euclidean dan koefisien korelasi Pearson memiliki pendekatan yang berbeda dalam mengukur kesamaan antara data pengguna. Hal ini mengakibatkan hasil rekomendasi yang dikeluarkan kedua metode berbeda. Metode Pearson lebih fokus pada korelasi linear antara dua variabel, sementara metode Euclidean

mengukur jarak geometris antara titik-titik data dalam ruang multidimensi.

- 2) Metode Euclidean mengukur jarak geometris antara preferensi pengguna. Nilai yang lebih kecil menunjukkan bahwa preferensi pengguna lebih mirip satu sama lain dalam ruang dimensi yang dianalisis. Dalam konteks sistem rekomendasi film, hal ini berarti bahwa pengguna dengan nilai Euclidean yang lebih kecil memiliki preferensi yang lebih serupa dengan pengguna lainnya. Oleh karena itu, film-film yang direkomendasikan kepada *trial user* adalah film-film yang mendapat rating tinggi dari pengguna dengan preferensi yang mirip, seperti yang diukur oleh metode Euclidean. Dengan demikian, metode Euclidean memungkinkan sistem rekomendasi untuk menyesuaikan rekomendasi film berdasarkan kemiripan preferensi pengguna, sehingga memastikan bahwa pengguna *trial* mendapatkan rekomendasi yang lebih relevan dan sesuai dengan selera mereka. Hal ini dapat meningkatkan kepuasan pengguna dan efektivitas sistem rekomendasi secara keseluruhan.
- 3) Koefisien korelasi Pearson, yang memiliki rentang nilai dari -1 hingga 1, memberikan indikasi seberapa erat hubungan antara preferensi pengguna *trial* dan pengguna lainnya dalam penilaian film. Ketika koefisien korelasi mendekati nilai 1, itu menandakan adanya kesesuaian yang semakin besar antara preferensi pengguna *trial* dan pengguna lainnya. Dengan kata lain, semakin tinggi nilai koefisien korelasi Pearson, semakin serupa preferensi antara pengguna *trial* dan pengguna lain, yang menyiratkan bahwa film-film yang dinikmati oleh pengguna lain dengan penilaian yang tinggi kemungkinan besar akan cocok dan relevan juga bagi pengguna *trial*.

### B. Rekomendasi

Berdasarkan hasil analisis dan kesimpulan penelitian, terdapat perbedaan hasil yang didapat menggunakan metode jarak Euclidean dan korelasi Pearson. Pada dasarnya, perbedaan hasil merupakan hal yang wajar terjadi karena dasar metode yang digunakan pun memiliki perbedaan yang cukup signifikan. Oleh sebab itu, kami merekomendasikan untuk menggunakan lebih dari satu metode dalam setiap langkah observasi yang dilakukan. Selain itu, kami juga merekomendasikan penggunaan kedua metode ini untuk memberikan rekomendasi film berdasar rating pengguna lainnya kepada pengguna yang baru.

Kedua metode ini memiliki langkah yang berbeda dan rasional. Metode Euclidean dapat memberikan rekomendasi berdasarkan kemiripan langsung antara preferensi pengguna, sementara metode Pearson dapat memberikan rekomendasi berdasarkan kesamaan pola atau tren dalam preferensi pengguna. Kombinasi dari kedua metode ini dapat meningkatkan kualitas rekomendasi film dan memperkaya pengalaman pengguna dalam menemukan konten yang sesuai dengan selera mereka.

## APPENDIX A VARIABEL YANG DIGUNAKAN

Dalam melakukan perhitungan rekomendator film menggunakan syntax MATLAB, terdapat beberapa variabel yang digunakan

TABLE I. VARIABEL SYNTAX MATLAB

Variabel	Definisi	Tipe Data
closestuserDist	jarak Euclidean terkecil	float
closestuserPearson	koefisien korelasi Pearson terbesar	float
comparison	vektor berisi 0 dan 1	float
DistIndex	indeks urutan eucl dari terkecil	float
eucl	array Jarak Euclidean	float
indexsmall	antara trialuser dengan ratings	float
liked	indeks 20 movies terpopuler	float
m	vektor movies rating 5 oleh trial user	float
m1	jumlah baris dari matriks usermovies	float
m2	jumlah baris dari matriks usersmoviessort	float
m3	jumlah baris dari matriks ratings	float
MaxPearson	jumlah baris dari matriks ratingscent	float
MinDist	korelasi Pearson terbesar	float
movies	Jarak Euclidean terpendek	float
mycent	array judul-judul movies	integer
myclosestuserDist	vektor myratings yang telah disentralisasi	float
myclosestuserPearson	indeks jarak Euclidean terkecil	float
myrecommendDist	indeks korelasi Pearson terbesar	float
myrecommendPearson	vektor indeks movies yang direkomendasikan kepada kelompok kami menurut Jarak Euclidean	float
mycomparison	indeks movies rekomendasi	float
myDistIndex	kelompok kami menurut korelasi Pearson	float
myeucl	vektor berisi 0 dan 1	float
myliked	indeks urutan myeucl terkecil	float
myMaxPearson	vektor Jarak Euclidean myratings dengan setiap baris matriks ratings	float
myMinDist	vektor indeks movies rating 5 yang kelompok kami suka	float
mypearson	urutan terbesar vektor mypearson	float
myPearsonIndex	urutan jarak euclidean terpendek	float
myratings	vektor koefisien korelasi pearson	float
n	antara ratingscent dengan mycent	float
n1	indeks urutan terbesar korelasi Pearson	float
n2	indeks urutan terbesar korelasi Pearson	float
pearson	vektor rating 20 movies terpopuler oleh kelompok kami	float
PearsonIndex	jumlah kolom matriks usermovies	float
ratings	indeks kolom matriks usersmoviessort	float
ratingscent	jumlah kolom matriks ratings	float
recommendDist	vektor korelasi pearson	float
recommendPearson	antara ratingscent dan trialusercent	float
rowi	indeks urutan korelasi Pearson terbesar	float
trialuser	matriks yang berisi rating yang diberikan oleh usermoviessort ke seluruh 20 movies terpopuler (tidak ada rating 0)	float
trialusercent	matriks ratings setelah disentralisasi	float
usermovies	vektor indeks movies rekomendasi	float
usersmoviessort	trial user menurut Euclidean	float
	vektor indeks movies rekomendasi	float
	trial user menurut korelasi Pearson	float
	vektor baris ke-i dari matriks ratingscent	float
	rating 20 movies terpopuler dari trial user	float
	vektor trialuser setelah disentralisasi	float
	matriks rating movies dari 0-5	float
	rating dari 20 film terpopuler (termasuk rating 0) oleh semua users	float

## APPENDIX B KODE MATLAB

Kode Matlab dapat diakses melalui <https://bit.ly/KodeMatlabProjek7Alin>

Berikut adalah kode Matlab yang digunakan untuk menyelesaikan permasalahan rekomendasi film untuk pengguna.

```
%% No 1 : Load the data
clear;
%Memuat data
load('users_movies.mat','movies','users_movies','users_movies_sort', ...
    'index_small','trial_user');
% m dan n adalah jumlah baris dan kolom dari users_movies
[m,n]=size(users_movies);
```

Fig. 1. Kode untuk pertanyaan nomor 1

```
%% No 2 : print out 20 most popular movies
% index_small : indeks dari 20 movies terpopuler
fprintf('20 movies terpopuler adalah sebagai berikut:\n')
for j=1:length(index_small)
    fprintf('%s \n',movies{index_small(j)})
end;
fprintf('\n')
```

Fig. 2. Kode untuk pertanyaan nomor 2

```
%% No 3 : Membuat matriks ratings yg berisi user yg rate keseluruhan 20
% film terpopuler
% m1 dan n1 adalah jumlah baris dan kolom dari matriks users_movies_sort
[m1,n1]=size(users_movies_sort);

% ratings adalah matriks yang berisi rating yang diberikan oleh
% user_movies_sort ke seluruh 20 movies terpopuler (tidak ada rating 0)
ratings=[];
for j=1:m1
    %Cek apakah user rate seluruh 20 movies terpopuler dengan cara mengalikan
    % semua rating yg diberikan. Jika hasil perkaliannya sama dengan 0, maka
    % terdapat movie yang tidak di-rate oleh user tersebut.
    % Matriks ratings hanya mengambil user yang rate keseluruhan movies
    % terpopuler yakni jika perkalian ratingnya tidak sama dengan 0
    if prod(users_movies_sort(j,:))~=0
        ratings=[ratings; users_movies_sort(j,:)];
    end;
end;
```

Fig. 3. Kode untuk pertanyaan nomor 3

```
%% No 4 : Find the Euclidean distances
% m2 dan n2 adalah jumlah baris dan kolom dari matriks ratings
[m2,n2]=size(ratings);
for i=1:m2
    %menghitung jarak Euclidean antara setiap baris ratings (setiap user
    % yg memberikan rating keseluruhan film terpopuler) dengan trial_user.
    % Jarak euclidean disimpan dalam array eucl
    eucl(i)=norm(ratings(i,:)-trial_user);
end
```

Fig. 4. Kode untuk pertanyaan nomor 4

```
%% No 5
% Mengurutkan jarak euclidean dari terkecil ke terbesar.
% Semakin dekat jaraknya semakin mirip taste movie nya.
% MinDist : eucl (Jarak Euclidean) diurutkan dari yg terpendek
% DistIndex : indeks yang berkaitan dengan eucl yg telah diurutkan dari
% terpendek.
% closest_user_Dist : indeks dari user dengan jarak Euclidean terpendek

[MinDist,DistIndex]=sort(eucl,'ascend');
closest_user_Dist=DistIndex(1);
```

Fig. 5. Kode untuk pertanyaan nomor 5



```
%% No 6
%sentralisasi matriks ratings dan trial_user
ratings_cent = ratings-mean(ratings,2)*ones(1,n2);
trial_user_cent=trial_user-mean(trial_user);
```

Fig. 6. Kode untuk pertanyaan nomor 6

```
%% No 7
% m3 adalah jumlah baris dari matriks ratings
m3 = size(ratings_cent, 1);

%inisialisasi vektor pearson yang nantinya untuk menyimpan koefisien
%korelasi Pearson
pearson = zeros(1, m3);

%Menghitung koefisien korelasi Pearson
for i = 1:m3
    % Mengambil baris ke-i ratings_cent
    row_i = ratings_cent(i, :);

    %Koefisien korelasi Pearson antara baris ke-i rating_cent dengan
    %trial_user_cent
    pearson(i) = corr(row_i', trial_user_cent', 'Type', 'Pearson');
end
```

Fig. 7. Kode untuk pertanyaan nomor 7

```
%% No 8
%MaxPearson : koefisien korelasi Pearson diurutkan dari yg paling besar
%PearsonIndex : indeks yg berkaitan dengan koefisien korelasi Pearson setelah
% diurutkan dari yang paling besar
[MaxPearson, PearsonIndex] = sort(pearson, 'descend');

% closest_user_Pearson : indeks dari koefisien korelasi Pearson terbesar
closest_user_Pearson = PearsonIndex(1);
```

Fig. 8. Kode untuk pertanyaan nomor 8

```
%% No 9
% Membandingkan hasil dari menggunakan jarak Euclidean dengan koefisien
% korelasi Pearson.
% comparison = 1 jika DistIndex = PearsonIndex dan comparison = 0 jika
% keduanya tidak sama
comparison = (DistIndex == PearsonIndex);

% Menampilakn indeks user dengan taste yang mirip dengan trial user

%berdasarkan koefisien korelasi Pearson
disp('Closest User Pearson:');
disp(closest_user_Pearson);

%berdasarkan jarak Euclidean
disp('Closest User Dist');
disp(closest_user_Dist);

%Diperoleh closest_user_Dist = 14 dan closest_user_Pearson = 88
% sehingga hasil yang diperoleh dengan menggunakan jarak Euclidean dan
% koefisien korelasi Pearson akan berbeda
```

Fig. 9. Kode untuk pertanyaan nomor 9

```
%% No 10 : Recommendations
%recommend_dist adalah array yg berisi indeks movies yang direkomendasikan kepada trial user menurut Jarak Euclidean
recommend_dist=[];

%menentukan movies yang akan direkomendasikan kepada trial user berdasarkan jarak Euclidean
% n = banyaknya kolom dari user_movies (banyaknya movies)
for k=1:n
    %jika closest_user_Dist memberikan rating 5 pada movie indeks ke-k, maka rekomendasikan movie tersebut kepada trial user
    if (users_movies(closest_user_Dist,k)==5)
        recommend_dist=[recommend_dist; k];
    end;
end;

%recommend_Pearson vektor yg berisi indeks movies yang direkomendasikan kepada trial user menurut koefisien korelasi Pearson
recommend_Pearson=[];
for k=1:n
    %jika rate film pada indeks k oleh closest_user_Pearson = 5, maka rekomendasikan movie tersebut kepada trial user
    if (users_movies(closest_user_Pearson,k)==5)
        recommend_Pearson=[recommend_Pearson; k];
    end;
end;

%liked adalah array yg berisi movies yg disukai trial user dgn kriteria trial user memberi rating 5
liked=[];
for k=1:120
    %jika trial user memberikan rating 5 maka trial user suka dengan movie tersebut
    if (trial_user(k)==5)
        %masukkan indeks movie yang disukai oleh trial user ke array liked
        liked=[liked; index_small(k)];
    end;
end;
```

Fig. 10. Kode untuk pertanyaan nomor 10

```
%% No 12
%Ratings yang diberikan oleh kelompok kami untuk ke seluruh 20 movies terpopuler
% yang disimpan di dalam array myratings
myratings= [4, 5, 0, 1, 3, 4, 4, 0, 3, 1, 5, 5, 0, 3, 1, 4, 4, 5, 3, 2];
```

```
%% No 12
%Ratings yang diberikan oleh kelompok kami untuk ke seluruh 20 movies terpopuler
% yang disimpan di dalam array myratings
myratings= [4, 5, 0, 1, 3, 4, 4, 0, 3, 1, 5, 5, 0, 3, 1, 4, 4, 5, 3, 2];
```

```
%% No 13
%m2 adalah jumlah baris dari matriks ratings
for i=1:m2
    %Menghitung jarak Euclidean antara setiap baris ratings (setiap user
    %yg memberikan rating keseluruhan movies terpopuler) dengan myratings.
    %Jarak euclidean disimpan dalam array myeucl
    myeucl(i)=norm(ratings(i,:)-myratings);
end

%Mengurutkan jarak euclidean dari terkecil ke terbesar
%Semakin dekat jaraknya semakin mirip taste movie nya
%myMinDist : myeucl (Jarak Euclidean) diurutkan dari yg terpendek
%myDistIndex : indeks yang berkaitan dengan myeucl yg telah diurutkan dari
%terpendek
%closest_user_Dist : indeks dari user dengan jarak Euclidean terpendek
[myMinDist,myDistIndex]=sort(myeucl,'ascend');
my_closest_user_Dist=myDistIndex(1)

%sentralisasi array myratings
my_cent=myratings-mean(myratings);

%inisialisasi array mypearson yang nantinya untuk menyimpan koefisien
%korelasi Pearson
mypearson = zeros(1, m3);

%Menghitung koefisien korelasi Pearson
% m3 adalah jumlah baris dari matriks ratings_cent
for i = 1:m3
    % Mengambil baris ke-i ratings_cent
    row_i = ratings_cent(i, :);

    %Koefisien korelasi Pearson antara baris ke-i rating_cent dengan my_cent
    mypearson(i) = corr(row_i', my_cent', 'Type', 'Pearson');
end

%myMaxPearson : koefisien korelasi Pearson diurutkan dari yg paling besar
%myPearsonIndex : indeks yg berkaitan dengan koefisien korelasi Pearson
% setelah diurutkan dari yang paling besar
[myMaxPearson, myPearsonIndex] = sort(mypearson, 'descend');

%my_closest_user_Pearson : indeks dari koefisien korelasi Pearson terbesar
my_closest_user_Pearson = myPearsonIndex(1)

% Membandingkan hasil dari menggunakan jarak Euclidean dengan koefisien
% korelasi Pearson
% mycomparison = 1 jika myDistIndex = myPearsonIndex dan mycomparison = 0 jika
% keduanya tidak sama
mycomparison = (myDistIndex == myPearsonIndex);
```

```

% Menampilkan indeks user dengan taste yang mirip dengan kelompok kami

%berdasarkan koefisien korelasi Pearson
disp('My Closest User Pearson:');
disp(my_closest_user_Pearson);

%berdasarkan jarak Euclidean
disp('My Closest User Dist:');
disp(my_closest_user_Dist);

%my_recommend_dist adalah array yg berisi indeks movies yang direkomendasikan
%kepada kelompok kami menurut Jarak Euclidean
my_recommend_dist=[];
%n = banyaknya kolom matriks user_movies (banyaknya movies)
for k=1:n
    %menentukan movies yang akan direkomendasikan kepada kelompok kami
    %berdasarkan jarak Euclidean
    if (users_movies(my_closest_user_Dist,k)==5)
        %jika my_closest_user_dist memberikan rating 5 pada movie indeks ke-k,
        % maka rekomendasikan movie tersebut kepada kelompok kami
        my_recommend_dist=[my_recommend_dist; k];
    end;
end;

%my_recommend_Pearson adalah array yg berisi indeks movies yang direkomendasikan
%kepada kelompok kami menurut koefisien korelasi Pearson
my_recommend_Pearson=[];
for k=1:n
    %jika rate movie pada indeks k oleh my_closest_user_Pearson = 5,
    % maka rekomendasikan movie tersebut kepada kelompok kami
    if (users_movies(my_closest_user_Pearson,k)==5)
        my_recommend_Pearson=[my_recommend_Pearson; k];
    end;
end;

%myliked adalah array yg berisi movies yg disukai kelompok kami dgn kriteria
% kami memberi rating 5
myliked=[];
for k=1:20
    %jika kami memberikan rating 5 maka kami suka dengan movie tersebut
    if (myratings(k)==5)
        myliked=[myliked; index_small(k)];
    end;
end;

%Menampilkan judul movies yang kami suka
disp('Film yg kami suka :')
for i = 1 :length(myliked)
    fprintf('%s \n',movies{myliked(i)});
end;

fprintf('\n');

%Menampilkan judul movies yang direkomendasikan kepada kelompok kami
%berdasarkan jarak Euclidean
disp('Film yg direkomendasikan untuk saya berdasarkan jarak Euclidean :')
for i = 1 :length(my_recommend_dist)
    fprintf('%s \n',movies{my_recommend_dist(i)});
end;
fprintf('\n');

%Menampilkan judul movies yang direkomendasikan kepada kelompok kami
%berdasarkan koefisien korelasi Pearson
disp('Film yg direkomendasikan berdasarkan koefisien Pearson :')
for i = 1 :length(my_recommend_Pearson)
    fprintf('%s \n',movies{my_recommend_Pearson(i)});
end;
fprintf('\n');

```

Fig. 13. Kode untuk pertanyaan nomor 13

## APPENDIX C DATA

Data yang digunakan dan dikeluarkan oleh kode Matlab dalam proyek ini dapat diakses melalui

<https://bit.ly/DataProjek7Alin>

## REFERENCES

- [1] C.M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer Science+Business Media, 2006. [E-book] Available: microsoft, <https://www.microsoft.com/en-us/research/uploads/prod/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf>. Accessed on: May. 23, 2024.
- [2] "Korelasi Pearson," Dec. 3, 2023, [Online]. Available: <https://ss.mipa.ub.ac.id/korelasi-pearson/>. [Accessed: Jun. 6, 2024].