

# דירוג רכבים באמצעות אלגוריתם TOPSIS



## אלגוריתמים היוריסטיים ומקורבים ויישומים

מגיש: נריה יקותיאל

תאריך: 16.09.2019

## תוכן עניינים

3.....	רקע
4.....	הקריטריונים שנבדקו
5.....	קבוצות בדיקה
Error! Bookmark not defined. ....	דירוג קבוצות בדיקה
7.....	אלגוריתם TOPSIS
7.....	תהליך האלגוריתם
Error! Bookmark not defined. ....	שימוש האלגוריתם בבחירת רכב
Error! Bookmark not defined. ....	דוגמה לנתונים אמיתיים
Error! Bookmark not defined. ....	נתונים
11.....	נרמול נתונים
12.....	מימוש אלגוריתם TOPSIS בפייתון
13.....	סימולציות
13.....	סימולציה Golden Age
14.....	סימולציה Family
14.....	סימולציה Teenagers
15.....	אלגוריתם TOPSIS – קוד פייתון
17.....	מסקנות – אלגוריתם TOPSIS

כיום יש מבחר ענק של רכבים מגוונים מאוד שמתאימים להמון צרכים. כמו כן, יש המון אופציות רכישה כגון, רכישה באופן פרטי, קניה מליסינג וכמובן קנית רכב חדש מסוכנות רכבים.

כאשר אדם מגיע לסוכנות ומעוניין לקנות רכב, סוכן המכירות שואל את אותו אדם מספר שאלות על סוג הרכב שאותו מעוניין לקנות. לאחר מכן בהתבסס על תשובותיו של אותו אדם, סוכן הנסיעות מציע לו כמה רכבים שמתאימים, פחות או יותר, לדרישותיו של הלקוח.

אך המון אנשים בימינו אינם מודעים להמון פרמטרים שיש לקחת בחשבון כאשר הם קונים רכב. פרמטרים כגון:

- **בסיס הגלגלים** – המרחק בין מרכז הגלגל האחורי למרכז הגלגל הקדמי, פרמטר זה הוא ד"י חשוב ללקוח שמעוניין בנוחות פנים הרכב. ככל שבסיס הגלגלים גדול יותר כך פנים הרכב אמור להיות גדול יותר.
- **רוחב הרכב** – כמו הפרמטר הקודם, גם פרמטר זה מעיד על מרחב בפנים הרכב, ככל שהרוחב גדול יותר, קח לשלושה אנשים לדוגמה, יהיה יותר נוח לשבת במושב האחורי.
- **מבחן בטיחות** – כיום, כל רכב, שיוצא מהמפעל, עובר מבחן בטיחות. מבחן שבו בודקים מה מצב הרכב והנוסעים שיושבים בו לאחר פגיעה חזיתית. כמו כן, כל שנה המבחן הנ"ל משתכלל ונהייה יותר נוקשה למען בטיחות הקהל הרחב. (לרוב כאשר מפרסמים את רמת הבטיחות של רכב, מפרסמים את רמת אבזור הבטיחות שלו, חיישני רוורס, עצירה אוטונומית בחירום וכו'. יש לציין שהפרמטר הזה שונה מפרמטר מבחן הבטיחות).
- **דירוג** – הפרמטר הזה אינו ידוע לציבור הרחב בצורה משמעותית (אני אישית לא הכרתי אותו). אתרים ומגזינים של רכבים בודקים כל רכב חדש שהושק בעזרת נסיעת מבחן. חלק מהדברים שנבדקים בנסיעה זו היא נוחות הישיבה ברכב, תגובתיות הרכב בנסיעה, נוחות האבזורים ברכב, וכמובן הביצועים בזמן הנסיעה. כאשר אותם מומחים מסיימים את נסיעת המבחן, הם מדרגים על פי דעתם את הרכב, ובסופו של דבר משקללים דירוג זה ע"י כמה מומחים ומפרסמים באתר (לדוגמה, באתר שממנו לקחתי את הנתונים לפרוייקט).

כמובן שגם הפרמטרים המוכרים נלקחו בחשבון, כמו המחיר הזול ביותר, צריכת דלק אופטימלית, גודל הרכב, גודל מיכל הדלק וגודל תא המטען.

## **הקריטריונים שנבדקו**

בחרתי 18 קריטריונים. את הקריטריונים חילקתי לארבע קבוצות שייכות:  
נוחות, כלכלי, ביצועים ושונות.

### נוחות

- מספר הדלתות – Number of Doors
- בסיס הגלגלים – Largest Wheels Base
- רוחב הרכב – Car Width
- אורך הרכב – Shortest Car Length
- גודל תא המטען – Boot Size
- גודל מיכל הדלק – Fuel Tank Liters

### כלכלי

- מחיר – Lowest Price
- צריכת דלק – ק"מ/לליטר – Fuel Consumption – Km/Liter

### ביצועים

- האצה מ0 עד 100 – Shortest 0 to 100
- נפח מנוע – Engine Volume
- מספר צילינדרים – Cylinders
- מספר שסתומים – Valves
- משקל/הספק – Power-to-weight Ratio
- מהירות מקסימלית – Top Speed
- טורבו – Turbo

### שונות

- משקל – Weight
- דירוג מומחה – Specialist Rank
- מבחן בטיחות – Safety Test

## קבוצות בדיקה

למען הניסוי, בחרתי שלוש קבוצות בדיקה, משפחה, צעירים ופנסיונרים.  
את הפרמטרים דירגתי מ 1 עד 10, כאשר 1 מציין שהחשיבות לפרמטר נמוכה,  
ו 10 מציין חשיבות מקסימלית (את ערכים אלו נירמלתי ועל כך אפרט בהמשך).

	Number of Doors	Largest Wheels Base	Car Width	Shortest Car Length
criteriaVector				
Family Weights Vector	10	8	10	5
Teenager Weights Vector	1	1	1	1
Golden Age Weights Vector	10	3	7	10

	Boot Size	Fuel Tank Liters	Lowest Price	Fuel Consumption - Km/Liter
	9	8	6	10
	1	4	10	3
	1	8	2	9

	Shortest 0 To 100	Engine Volume	Cylinders	Valves	Power-to-weight ratio
	2	4	4	2	1
	10	10	1	1	10
	6	7	7	7	1

	Top Speed	Turbo	Weight	Specialist Rank	Safety Test
	1	1	10	10	10
	10	10	3	5	7
	1	7	9	10	10

## דירוג קבוצות הבחירה

בקבוצת ה**נוחות**, דירגתי את המשפחה והפנסיונרים הערכים בחשיבויות גבוהות ואת הצעירים בחשיבויות נמוכות.

בקבוצה ה**ביצועים**, דירגתי את הצעירים עם חשיבויות גבוהות במיוחד, את הפנסיונרים בחשיבויות פחות גבוהות ואת המשפחה בחשיבויות הכי נמוכות.

בקבוצה ה**כלכלית**, דירגתי את הצעירים בחשיבות הגבוהה ביותר מכיוון שלהם חשוב שהמחיר יהיה נמוך, את המשפחה בחשיבות בינונית מכיוון שהם מוכנים להתפשר על מחיר קצת גבוהה יותר ולקבל איכות טובה יותר, ואת הפנסיונרים דירגתי בחשיבות נמוכה מכיוון שיש אנשים מבוגרים שמוכנים להשקיע מחיר לא נמוך על רכב.

בקבוצה ה**שונות**, המשפחה והפנסיונרים קיבלו את החשיבויות הגבוהות ביותר במבחן הבטיחות ודירוג המומחה ולצעירים חשיבויות נמוכות יותר.

אלגוריתם TOPSIS

## תיאור האלגוריתם

האלגוריתם מבצע בחירה על כמות האלטרנטיבות הקיימות באמצעות אנליזה של מולטי-קריטריונים.

הקונספט הבסיסי של האלגוריתם אומר, שלאטרנטיבה הטובה ביותר יהיה המרחק הקצר ביותר מהפתרון האידיאלי החיובי והמרחק הארוך ביותר מהפתרון האידיאלי השלילי. כמובן שהפתרונות האלה הם לא ריאליים, ואין באמת אלטרנטיבה שהיא בעצמה הפתרון האידיאלי החיובי.

בנוסף, האלגוריתם משתמש במשקלים שנקבעים לכל קריטריון, משקלים שמהווים אחוזים.

## תהליך האלגוריתם

### Step 1

Create an evaluation matrix consisting of  $m$  alternatives and  $n$  criteria, with the intersection of each alternative and criteria given as  $x_{ij}$ , we therefore have a matrix  $(x_{ij})_{m \times n}$ .

### Step 2

The matrix  $(x_{ij})_{m \times n}$  is then normalised to form the matrix

$R = (r_{ij})_{m \times n}$ , using the normalisation method

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{k=1}^m x_{kj}^2}}, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n$$

### Step 3

Calculate the weighted normalised decision matrix

$$t_{ij} = r_{ij} \cdot w_j, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n$$

where  $w_j = W_j / \sum_{k=1}^n W_k$ ,  $j = 1, 2, \dots, n$  so that  $\sum_{i=1}^n w_i = 1$ , and  $W_j$  is the original weight given to the indicator  $v_j$ ,  $j = 1, 2, \dots, n$ .

### Step 4

Determine the worst alternative ( $A_w$ ) and the best alternative ( $A_b$ ):

$$A_w = \{ \langle \max(t_{ij} \mid i = 1, 2, \dots, m) \mid j \in J_- \rangle, \langle \min(t_{ij} \mid i = 1, 2, \dots, m) \mid j \in J_+ \rangle \} \equiv \{t_{wj} \mid j = 1, 2, \dots, n\},$$

$$A_b = \{ \langle \min(t_{ij} \mid i = 1, 2, \dots, m) \mid j \in J_- \rangle, \langle \max(t_{ij} \mid i = 1, 2, \dots, m) \mid j \in J_+ \rangle \} \equiv \{t_{bj} \mid j = 1, 2, \dots, n\},$$

where.

### Step 5

Calculate the  $L^2$ -distance between the target alternative  $i$  and the worst condition  $A_w$

$$d_{iw} = \sqrt{\sum_{j=1}^n (t_{ij} - t_{wj})^2}, \quad i = 1, 2, \dots, m,$$

and the distance between the alternative  $i$  and the best condition  $A_b$

$$d_{ib} = \sqrt{\sum_{j=1}^n (t_{ij} - t_{bj})^2}, \quad i = 1, 2, \dots, m$$

**Step 6**

Calculate the similarity to the worst condition:

$$s_{iw} = d_{iw} / (d_{iw} + d_{ib}), \quad 0 \leq s_{iw} \leq 1, \quad i = 1, 2, \dots, m.$$

$s_{iw} = 1$  if and only if the alternative solution has the best condition; and

$s_{iw} = 0$  if and only if the alternative solution has the worst condition.

**Step 7**

Rank the alternatives according to  $s_{iw}$  ( $i = 1, 2, \dots, m$ ).

שימוש האלגוריתם בבחירת רכב






בכדי שנוכל לממש את האלגוריתם על בחירת רכב, הוצאתי נתונים אמיתיים על רכבים מהאתר <http://www.netcar.co.il/guide/guide.php>

בחרתי 18 קריטריונים ו-49 אלטרנטיבות (רכבים).

מאתר זה בחרתי את כל הפרמטרים שלי ואת הרכבים עצמם.

### דוגמה לנתונים אמיתיים:

#### הונדה CR-V

1.5Prestige	1.5Prestige	
כבישטח מיקרוואן	כבישטח מיקרוואן	קבוצה
***	***	דירוג
213,000	233,000	מחיר
5	5	דלתות
שטח	שטח	מרכב
18	18	שנת השקה
5	5+2	מושבים
CVT	CVT	גיר אוטו
1	1	מספר הילוכים
✓	✓	מגדש
כ	כ	הנעה
-	-	דלק
✓	✓	בקרת אקלים
		כריות אוויר
5	5	ניסוי ריסוק
3	3	שנות אחריות
1498	1498	נפח מנוע
4	4	צילינדרים
16	16	שסתומים
193/5600	193/5600	הספק
24.8/2000	24.8/2000	מומנט
225/60X18	225/60X18	צמיגים
4.60	4.60	אורך
1.86	1.86	רוחב
1.69	1.69	גובה
2.66	2.66	בסיס גלגלים
561	472	תא מטען
57	57	מיכל דלק
1523	1590	משקל
7.9	8.2	משקל/הספק
0.93	0.93	מקדם דינמי
10.0	10.0	אפס למאה
200	200	קמ"ש
11.3	11.3	ק"מ/ליטר

כתובים

	criteraVector										Number of Doors										Largest Wheels Base										Car Width Shortest Car Length										Fuel Tank Liters										Lowest Price										Fuel Consumption - Km/Liter										Shortest 0 To 100										Engine Volume										Cylinders										Values										Power-to-weight Ratio										Top Speed										Turbo										Weight										Specialist Rank										Safety Test																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
	Family Weights Vector										10										8										5										9										6										2										4										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1										1									

כפי שניתן לראות, לכל פרמטר יש נתונים שונים, ביחידות מידה שונות, לכן על מנת שהאלגוריתם יעבוד כמו שצריך, נירמלתי כל עמודה לפי הנוסחה הבאה

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

כאשר  $X$  מייצג ערך אמיתי (למשל מחיר), לכל מחיר לכל רכב, נחסיר ממנו את הרכב הזול ביותר ונחלק בהפרש בין הרכב היקר ביותר לרכב הזול ביותר, כך נקבל  $Z$  ערך מנורמל לכל מחיר של כל רכב.

כמו כן, יש לנרמל את המשקלים שנתנו לכל פרמטר. בנרמול זה, נסכם את כל הערכים מכל פרמטר ונחלק כל ערך בסכום שקיבלנו, כך נקבל ערכים מ 0 עד 1 המייצגים באחוזים את משקל כל פרמטר.

### מימוש אלגוריתם TOPSIS בפייטון

את האלגוריתם מימשתי בפייטון, ממשק משתמש באמצעות Tkinter.

Car Selection - TOPSIS Implementation

Car Selection - TOPSIS Implementation - Rank Criteria from 1 to 10\*

1 = Least Important  
10 = Most Important

**Comfort**

Number of Doors

1

Largest Wheels Base

1

Car Width

1

Shortest Car Length

1

Boot Size

1

Fuel Tank Liters

1

**Economics**

Lowest Price

1

Fuel Consumption - Km/Liter

1

**Performance**

Shortest 0 To 100

1

Engine Volume

1

Cylinders

1

Valves

1

Power-to-weight Ratio

1

Top Speed

1

Turbo

1

**Miscellaneous**

Weight

1

Specialist Rank

1

Safety Test

1

\*Rank Wisely :)

\*Best Choice

\*Worst Choice

Calculate

Reset

Family

Teenagers

Golden Age

סימולציות

בדוגמה, ניתן לראות את המשקלים שמופיעים כאשר לוחצים על Golden Age לאחר מכן לוחצים על Calculate ונקבל את הרכב המתאים ביותר, זאת אומרת הרכב שהכי קרוב לפתרון האידיאלי החיובי והכי רחוק מהפתרון האידיאלי השלילי.

כמו כן נקבל את הרכב הכי פחות מתאים, הרכב שהכי רחוק מהפתרון האידיאלי החיובי והכי קרוב לפתרון האידיאלי השלילי.

### דוגמה כאשר לוחצים על Golden Age

Car Selection - TOPSIS Implementation

Car Selection - TOPSIS Implementation - Rank Criteria from 1 to 10\*

1 = Least Important  
10 = Most Important

<b>Comfort</b>		<b>Performance</b>	
Number of Doors	10	Shortest 0 To 100	6
Largest Wheels Base	3	Engine Volume	7
Car Width	7	Cylinders	7
Shortest Car Length	10	Valves	7
Boot Size	1	Power-to-weight Ratio	1
Fuel Tank Liters	8	Top Speed	1
		Turbo	7
<b>Economics</b>		<b>Miscellaneous</b>	
Lowest Price	2	Weight	9
Fuel Consumption - Km/Liter	9	Specialist Rank	10
		Safety Test	10

\*Rank Wisely :)

Run Time 0.000378200000001062

\*Best Choice ב-10-5

\*Worst Choice מחזקה דמיון

Calculate Reset

Family Teenagers Golden Age

### דוגמה כאשר לוחצים על Family

Car Selection - TOPSIS Implementation - Rank Criteria from 1 to 10\*

1 = Least Important  
10 = Most Important

<b>Comfort</b> Number of Doors: 10 Largest Wheels Base: 8 Car Width: 10 Shortest Car Length: 5 Boot Size: 9 Fuel Tank Liters: 8  <b>Economics</b> Lowest Price: 6 Fuel Consumption - Km/Liter: 10  *Rank Wisely :)	<b>Performance</b> Shortest 0 To 100: 2 Engine Volume: 4 Cylinders: 4 Valves: 2 Power-to-weight Ratio: 1 Top Speed: 1 Turbo: 1  <b>Miscellaneous</b> Weight: 10 Specialist Rank: 10 Safety Test: 10
--	---

Run Time: 0.0004155999999966296

\*Best Choice: 3-ח-10  
\*Worst Choice: יגואר F-Type

Family  
Teenagers  
Golden Age

Calculate Reset

## דוגמה כאשר לוחצים על Teenagers

Car Selection - TOPSIS Implementation - Rank Criteria from 1 to 10\*

1 = Least Important  
10 = Most Important

<b>Comfort</b> Number of Doors: 10 Largest Wheels Base: 3 Car Width: 7 Shortest Car Length: 10 Boot Size: 1 Fuel Tank Liters: 8  <b>Economics</b> Lowest Price: 2 Fuel Consumption - Km/Liter: 9  *Rank Wisely :)	<b>Performance</b> Shortest 0 To 100: 6 Engine Volume: 7 Cylinders: 7 Valves: 7 Power-to-weight Ratio: 1 Top Speed: 1 Turbo: 7  <b>Miscellaneous</b> Weight: 9 Specialist Rank: 10 Safety Test: 10
---	--

Run Time: 0.0003525999999993701

\*Best Choice: אלפא ג'וליה  
\*Worst Choice: סוויסה אייגו

Family  
Teenagers  
Golden Age

Calculate Reset

כפי שניתן לראות בדוגמאות הנ"ל, לכל קבוצה נקבל פתרון אחר בהתאם למשקלים הנבחרים.

## אלגוריתם TOPSIS – קוד פייתון



## מסקנות – אלגוריתם TOPSIS



המטרה שלנו הייתה לבחור את הרכב הטוב ביותר המתאים לקבוצה או אדם על פי חשיבויות שהם מגדירים לכל פרמטר המוצג בפניהם.

כפי שניתן, על פי משקלים שונים, קיבלנו לכל קבוצה רכב שונה לגמרי. בנוסף, ראינו שהרכב הגרוע ביותר המתאים לכל קבוצה גם משתנה על פי המשקלים השונים.

בנתונים שהשתמשתי בהם, מופיעים 49 רכבים ו-18 קריטריונים, זאת אומרת שהחישובים מתבצעים על 882 ערכים בסה"כ.

לפי זמני הריצה ניתן לראות שזמן הריצה הוא נע בין 0.000378 שניות ל 0.00045 שניות, אלה זמני ריצה מאוד קטנים.

כך שלדעתי, אלגוריתם TOPSIS הוא אלגוריתם די מהיר בזמן הריצה שלו.

זאת אומרת, שכאשר נכניס לאלגוריתם מספר גדול פי כמה וכמה של נתונים, זמני הריצה ישארו סביב הכמה שניות.

המסקנה העיקרית לגבי האלגוריתם הנ"ל היא שהוא מחשב באופן די מהיר את האלטרנטיבה האולטימטיבית ביחס למשקלים הנתונים.

מה שהופך אותו לאלגוריתם די טוב כאשר רוצים לגלות מה האלטרנטיבה הטובה ביותר עם משקלים נתונים.