**נשתמש ב Pareto Weighted Sum Tuning על מנת לפתור:   
 Multiobjective Optimization Problems  
[מקור 1]**בהינתן אוסף של מניות,אנחנו נרצה להשיג 2 מטרות:  
1.למקסם את *תוחלת הרווח המרבי*  
2. *למזער את תוחלת ההפסד המרבי*לשם כך,נשתמש בגישת Sum Weighted:

נתחיל בכך שנגדיר את *תוחלת ה לרכישת מנייה כלשהי.  
לשם כך נגדיר:  
 תוחלת הרווח המרבי ממנייה*   *תוחלת ההפסד המרבי ממנייה*   *המשקל שנותן הלקוח ל ,המשקל שנותן הלקוח לתוחלת הרווח המרבי.  
 המחיר של מנייה .  
 מקסימום התקציב שיש ללקוח  
 כמות המנייה שנקנה (1 או 0 )*

*לכן, תוחלת ה לרכישת מנייה :*

*כך בעצם אנחנו נותנים "משקל" גם לרווח וגם להפסד ומאחדים את זה למשתנה אחד .*

*וכעת בעיית האופיטיזציה שלנו:*

*כאשר*

*כלומר, נרצה למקסם את ה של כל המניות שנקנה, בגבולות התקציב שניתן לנו.  
לשם פתירת בעיה זו, נרצה למצוא את .*

*בקוד שלנו נמצא את הערך של :  
נלמד כל פעם בהתאם לנתונים מהו עד אשר נקבל את ערך ה הטוב ביותר בהתחשב בכמות האיטרציות שנגדיר.*

***הסבר על הקוד:***

ניצור אוסף של מניות בשם .כל מנייה מוגדרת כ .

למנייה מוגדר :

ניתן ל"משתמש" מדגם מתוך המניות אלו, והוא ידרגם בכל איטרציה.  
בתחילה הדירוג נעשה על ידי בחירת רנדומי.  
הדירוג הוא בעצם חישוב של :

ככל ש גבוהה יותר, המנייה טובה יותר ומשתלם לקנות אותה.  
ככל ש נמוכה יותר, המנייה גרועה יותר ופחות משתלם לקנות אותה.

נהפוך את דירוג זה למספרים בין 0 לגודל המדגם.אם גודל המדגם הינו אז:  
מנייה בדירוג 0 זו המנייה הגרועה ביותר ,והמנייה בדירוג זו המנייה הטובה ביותר. בעזרת מידע זה ו נלמד את .[מידע זה ישמש אותנו באיטרציה הבאה]  
בסוף האיטרציה,נעשה ממוצע על כל ה שקיבלנו עד כה ונקבל את . את ערך זה נשמור ונמשיך לאיטרציה הבאה.

*לאחר שסיימנו את כל האיטרציות,נחזיר את כל ערכי ששמרנו.*

נוכל לראות כי ככל שנשפר את נקבל רווח גדול יותר בצורה משמעותית, במחיר של הפסד קצת יותר גדול ,אך לא בצורה משמעותית:

Chart, scatter chart

Description automatically generated

**הפסדו קוד:**-נגדיר עם ערך שרירותי.   
-נגדיר שונות , tolerance עם ערך שרירותי  
- ניצור פרופיל למשתמש בעזרת ו tolerance  
-   
- ניצור את המניות - כפי שהוסבר מקודם.  
-ניצור מערך המכיל גדלים שונים ל //נוכל לבחור ערכים שרירותיים.  
-נגדיר מספר איטרציות מירבי=   
-נגדיר alpha\_vectors\_learned מערך ריק

ניצור מה בגודל

1. -לכל ב :
   1. לכל אשר קטן ממספר :
      1. נחזיר alpha\_learned בעזרת ה שיצרנו והפרופיל משתמש /// בעזרת SVM RANKING
      2. ניצור חדש בגודל מבין כל הזוגות שלא היו ב הקודם.
      3. נוסיף ל alpha\_vectors\_learned את ה alpha\_learned הנוכחי.
      4. נחשב את : ממוצע על כל ה alpha\_learned שלמדנו עד כה.
      5. נוסיף את ל result\_mean\_alpha
      6. ניצור פרופיל משתמש חדש בעזרת ה result\_mean\_alpha שלמדנו.
   2. נחזיר result\_mean\_alpha

**מקורות:**

1.השתמשנו בשיטה שלHarry Wang לפתירת Multiobjective Optimization Problems:  
https://github.com/harryw1248/Pareto-Weighted-Sum-Tuning