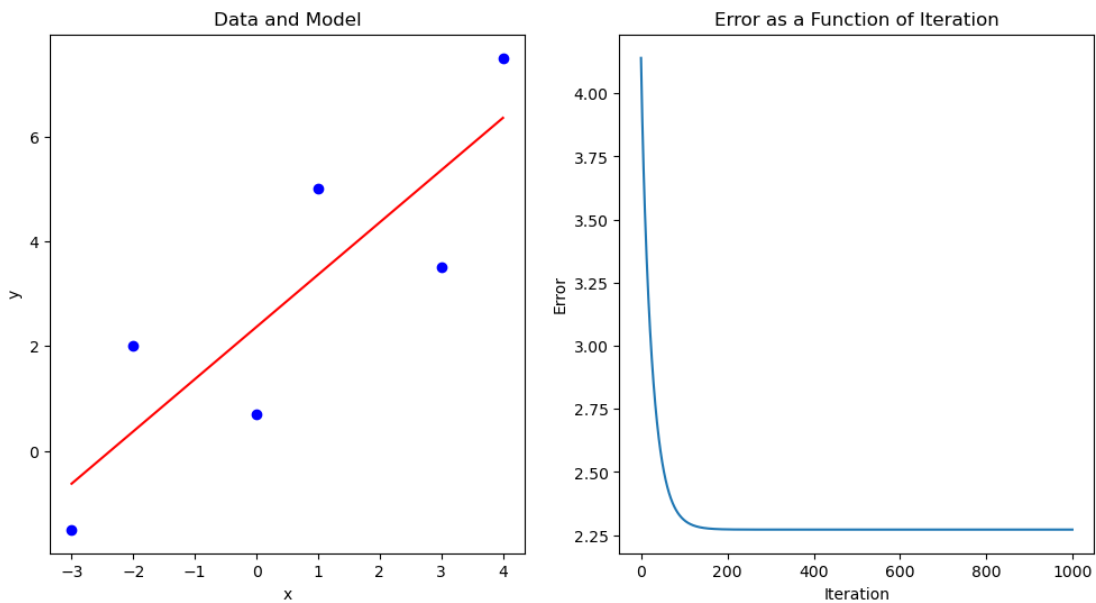


## פרויקט סופי

### שאלה 1

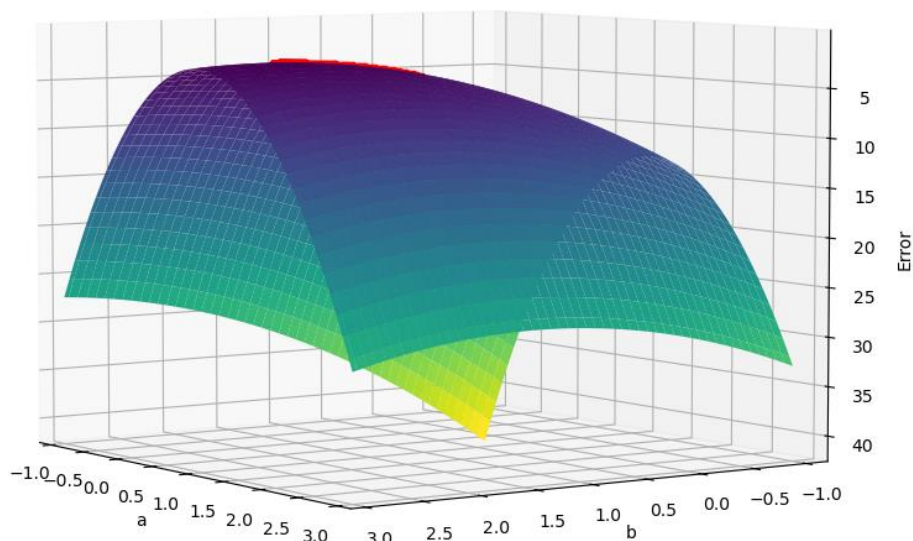
א.

Final parameters:  $a=0.997333333849981$ ,  $b=2.367999994270291$ .  
2. ו-3.

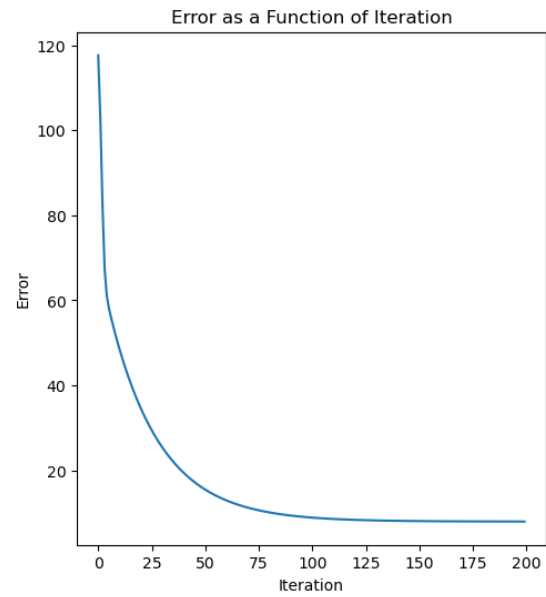
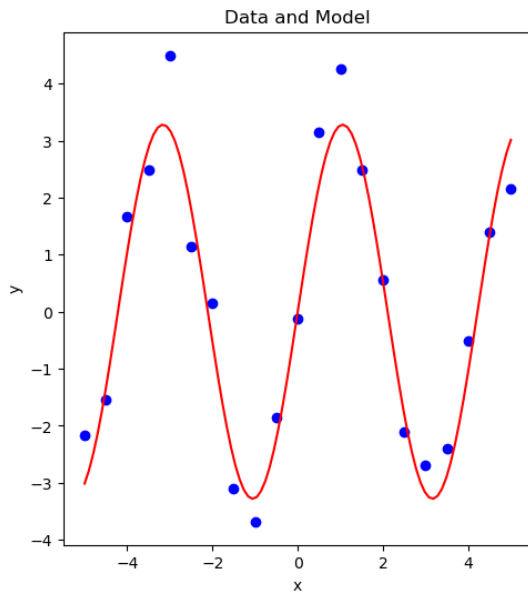


Error Surface Over Parameters

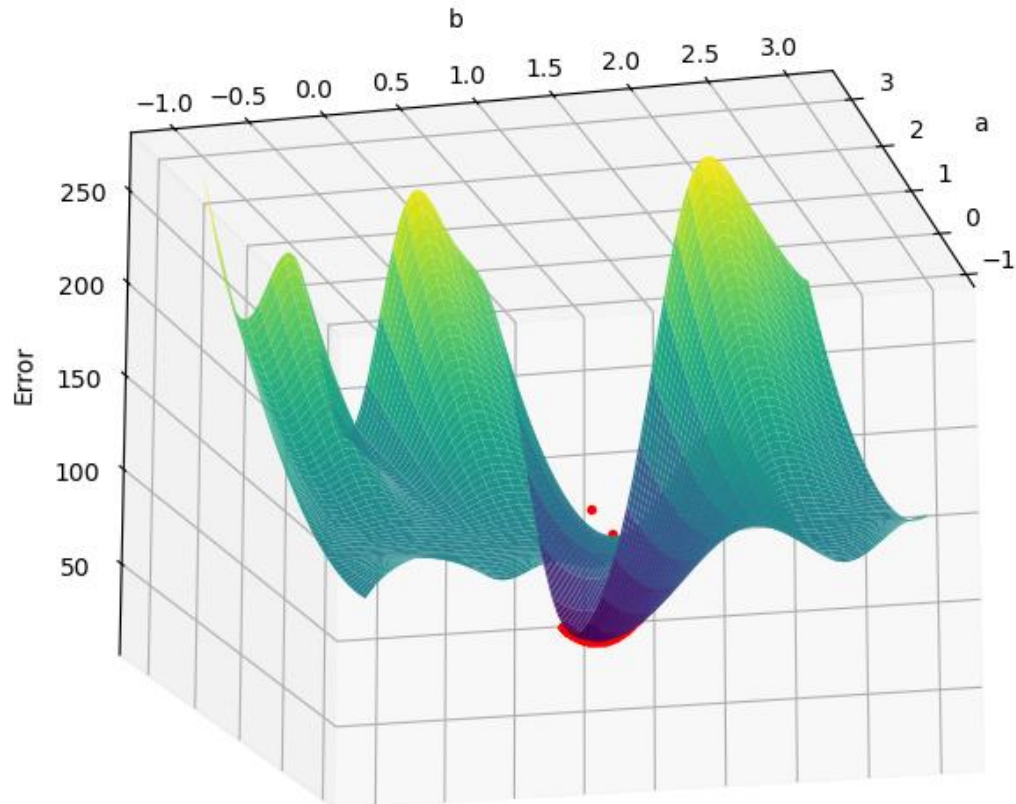
4.



ג.



Error Surface Over Parameters



מגישים:  
שחר אשר – 209305408  
הדר ליאל הרוש - 211721568

קצב למידה: 0.001

```
# Learning rate  
lr = 0.001
```

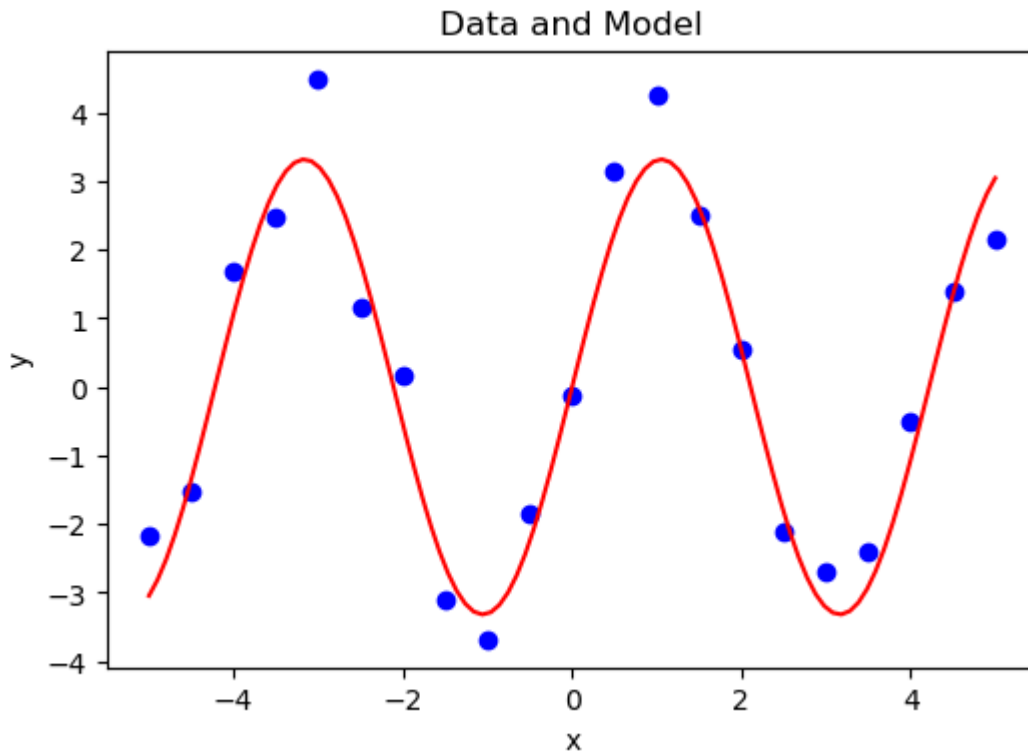
איטרציות: 200

```
# Number of iterations  
iterations = 200
```

הפרמטרים שמצאנו הם:

Final parameters:  $a=3.280345856906753$ ,  $b=1.4895161671221442$

.T



#### פרמטרים סופיים:

Final parameters:  $a=3.318145436025057$ ,  $b=1.488977981749276$

נראה שהמודל מתאים באופן טוב מאוד לנתונים. הקו האדום (המודל) עובר דרך רוב הנקודות הכחולות (הנתונים). זה מראה שפונקציית הסינוס מתאימה כמעט היטב למבנה של הנתונים. הפרמטרים הסופיים שהפונקציה `curve_fit` מצאה הם:

- $a = 3.318145436025057$

- $b = 1.488977981749276$

הפרמטר  $a$  הוא הערך שבו פונקציית הסינוס מתחילה (הגובה שלה), והפרמטר  $b$  הוא התדירות של הפונקציה. אם נשווה את הערכים האלה לערכים האמיתיים שהשתמשת בהם כדי ליצור את הנתונים, נראה שהפונקציה `curve_fit` מצאה פרמטרים שדי קרובים לערכים האמיתיים. זה מראה שהפונקציה `curve_fit` יכולה להיות כלי יעיל למציאת הפרמטרים של מודל שמתאים לנתונים.

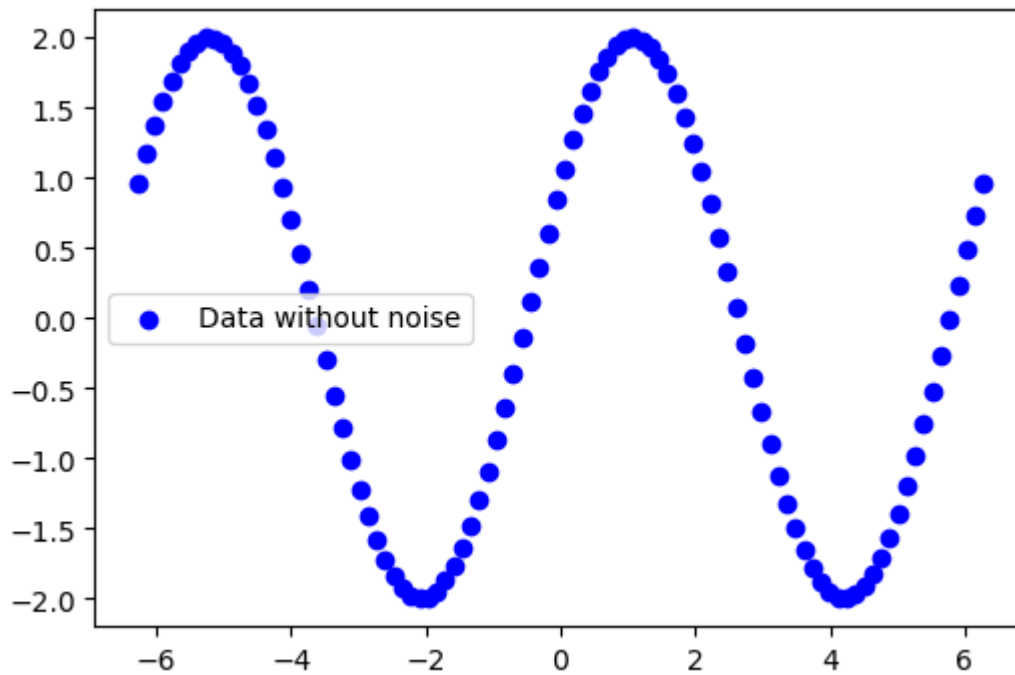
ה. המודל שבחרו הוא:  $f(x) = a \cdot \sin(b \cdot x + c)$ .

a, b ו- c הם הפרמטרים של המודל.

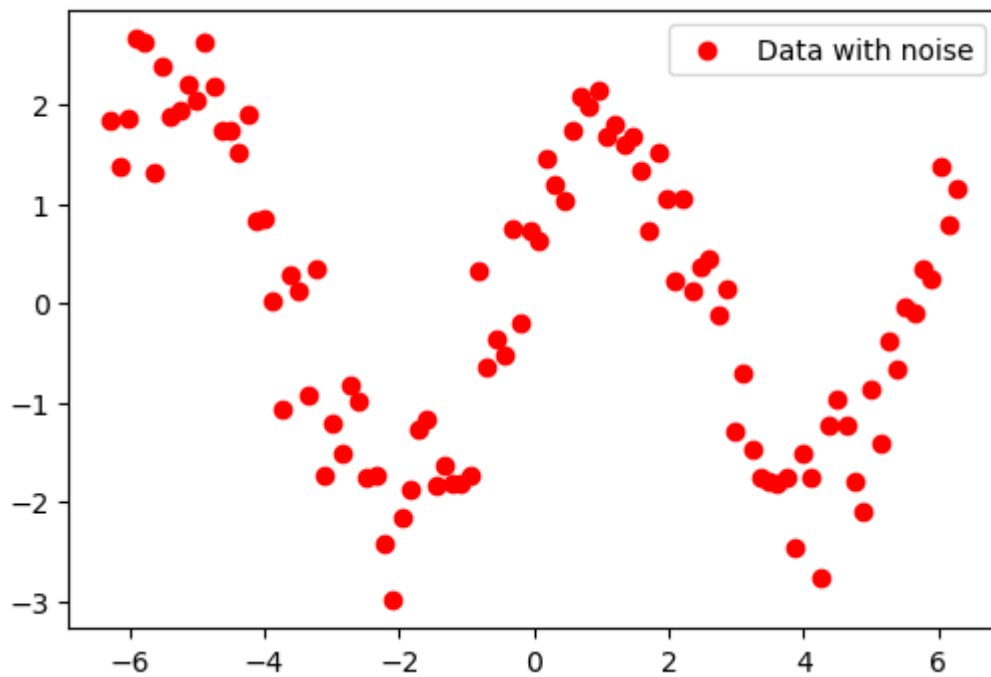
הפרמטרים ההתחלתיים שבחרנו הם:  $a\_val = 1, b\_val = 1, c\_val = 1$

```
# Initial parameters  
a_val, b_val, c_val = 1, 1, 1
```

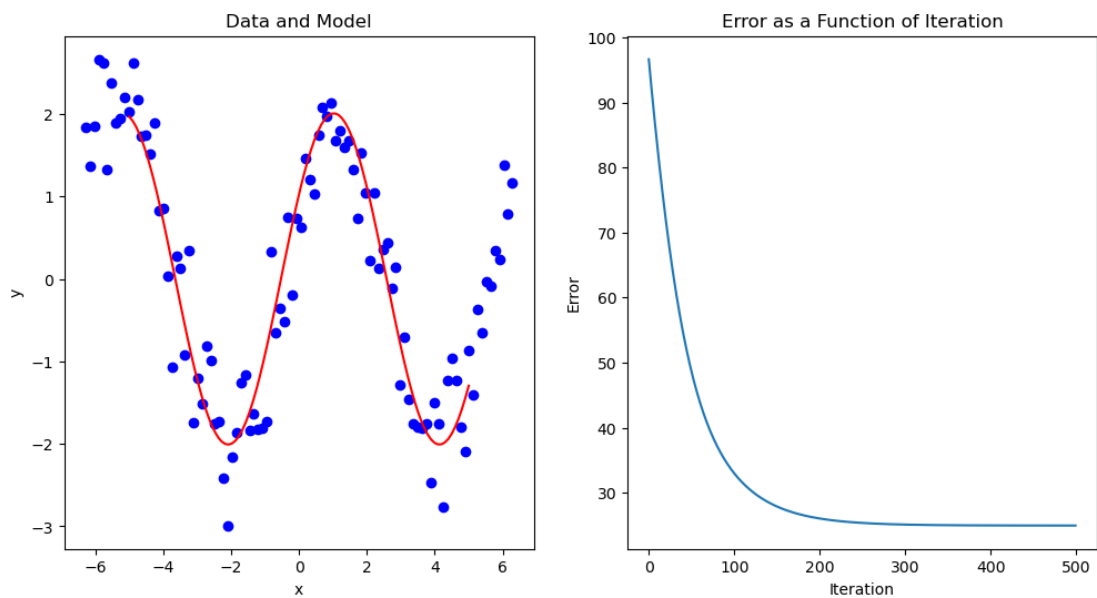
גרף לפני הוספת הרעש:



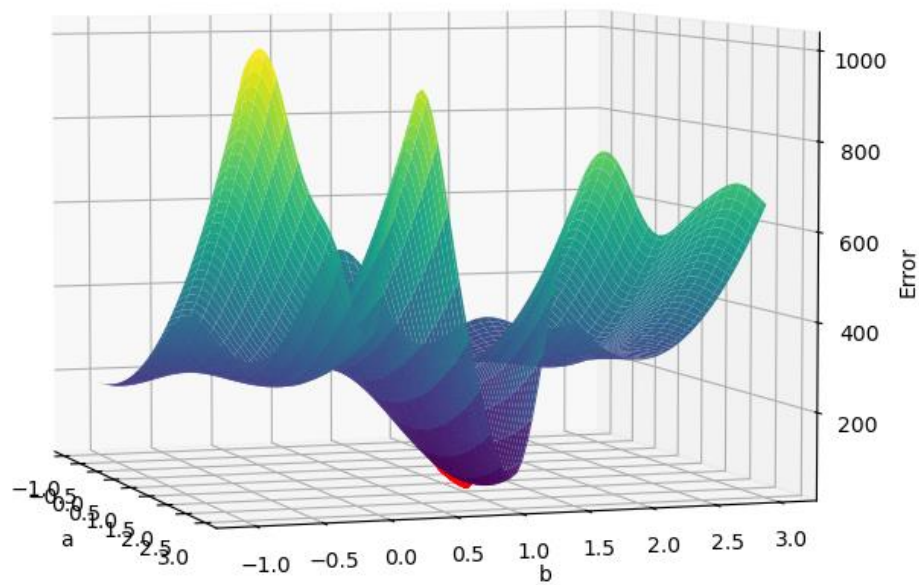
גרף אחרי הוספת הרעש:



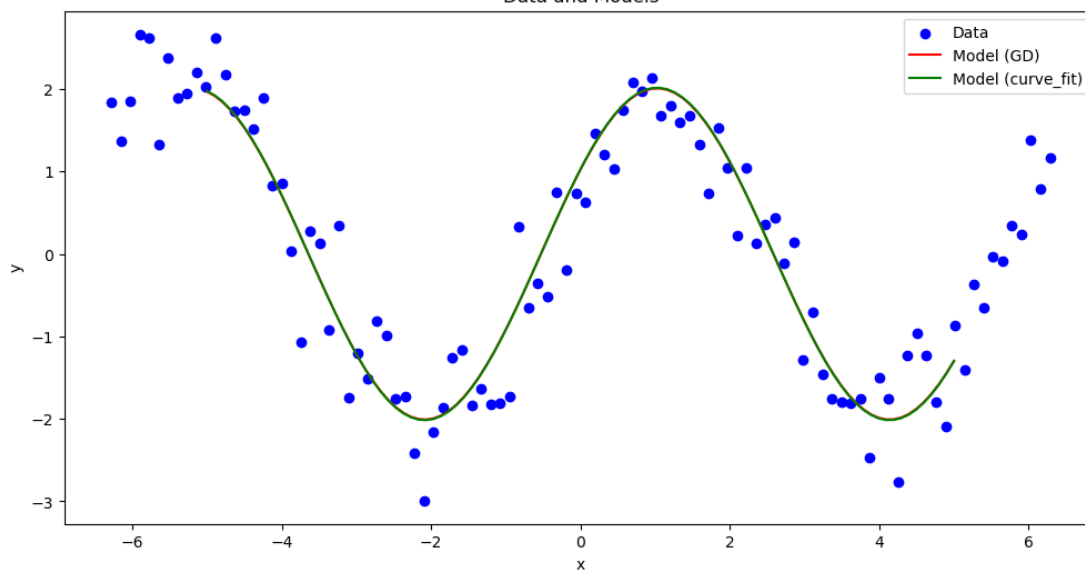
גרפים בדומה לסעיף א:



Error Surface Over Parameters



Data and Models



### פרמטרים שנמצאו על ידי Gradient descent:

Final parameters:  $a=2.0081125397799204$ ,  $b=1.0092632527092733$ ,  
 $c=0.536024564951171$

### פרמטרים שנמצאו על ידי curve\_fit:

Parameters found by curve\_fit:  $a=2.0156616351282866$ ,  
 $b=1.0093224363174194$ ,  $c=0.5359304117967073$

### הזמן שלוקח ל- Gradient descent:

Time elapsed for gradient descent: 2.832636833190918 seconds

### הזמן שלוקח ל- curve\_fit:

Time elapsed for curve\_fit: 0.0 seconds

ההבדלים בין שתי השיטות, Gradient descent ו- curve\_fit :

#### • זמן הריצה:

- Gradient descent: לקח כ-2.83 שניות למצוא את הפרמטרים האופטימליים.
- curve\_fit: שיטה זו הייתה מהירה באופן משמעותי, ולקחה רק כ-0.0 שניות.

ההבדלים נובעים מהתכונות הייחודיות של כל שיטה:

#### • **Gradient descent**: Gradient Descent מעדכן באופן איטרטיבי

את הפרמטרים. כל איטרציה כרוכה בחישוב הגרדיאנט ועדכון הפרמטרים בהתאם. תהליך זה נמשך עד שמגיעים להתכנסות.

- **curve\_fit**: curve\_fit משתמשת באלגוריתמי אופטימיזציה שתוכננו במיוחד כדי למצוא את הפרמטרים האופטימליים הממזערים את ההבדל בין הערכים הנצפים לערכים החזויים. אלגוריתמים אלו לרוב מתכנסים לפתרון בפחות איטרציות בהשוואה ל- Gradient Descent.

לסיכום, ההבדל בזמן הריצה נובע מהאופי האיטרטיבי של Gradient Descent, הדורש איטרציות רבות כדי להגיע להתכנס, במיוחד עבור פונקציות מורכבות או מערכי נתונים גדולים. וזאת לעומת curve\_fit, הנהנה מאלגוריתמי אופטימיזציה מיוחדים יותר שלעתים קרובות יעילים יותר במציאת הפרמטרים האופטימליים בפחות איטרציות.



### שאלה 3

א. TRUE:

1.  $(p1 \text{ and not } p2) \text{ or } (p1 \text{ and not } p4)$

2.  $(\text{not } p3 \text{ and not } p2)$

3.  $p1$

FALSE:

1.  $p5 \text{ or } p2$

2.  $p5 \text{ or } (p2 \text{ and } p8)$

3.  $(p2 \text{ and } p6) \text{ or } (p3 \text{ and } p7) \text{ or } (p9 \text{ and } p5)$

ב. נחשב את מספר הכללים שמסווגים נכונה את סט האימון:  
גודל המטריצה הוא – 3 על 3. ולכן, מספר המטריצות האפשריות הוא:  
 $9^2 = 512$ .

סט האימון שלנו מכיל 6 מטריצות מתוך כל המטריצות (מתוך 512  
מטריצות). לכן, סט המבחן יכיל 506 מטריצות ( $512 - 6 = 506$ ).

יש לנו שתי תשובות אפשריות – TRUE או FALSE. לכן, כל מטריצה  
יכולה להיות מסווגת לאחת מן התשובות – TRUE או FALSE.

↔ מספר הכללים שמסווגים את סט האימון הוא:  $2^{506}$ .