ממ"ן 11 – מבוא לראייה ממוחשבת, נדב מידן ת.ז. 200990240

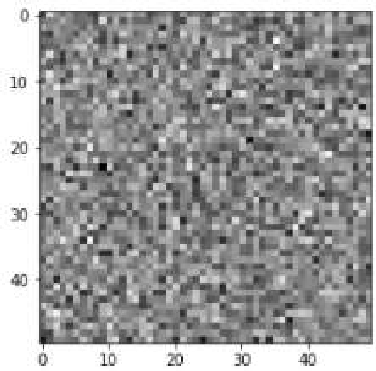
שאלה 1

א)

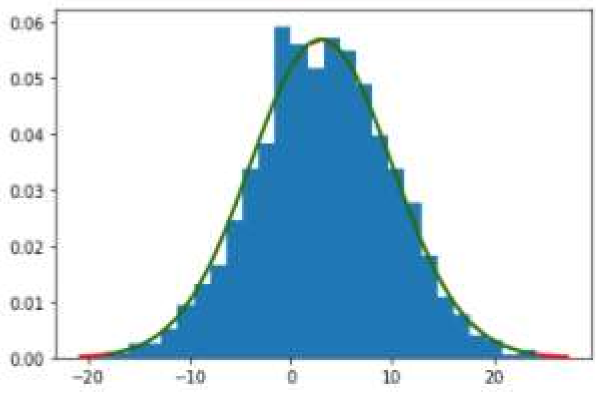
במטריצה מגודל 50 X 50 ישנן 2500 דגימות, לכן השתמשתי בפונקציית random.normal עם הפרמטרים הגאוסייניים (ההתפלגות הממוצעת וסטיית התקן) הנתונים בשאלה, על פני 2500 דגימות.

ע"מ להציג את המטריצה, השתמשתי בספרייה matplotlib, ובמתודהgray ע"מ להציגה ברמות אפור.

```
Out[23]: <matplotlib.image.AxesImage at 0x1818d515280>
```

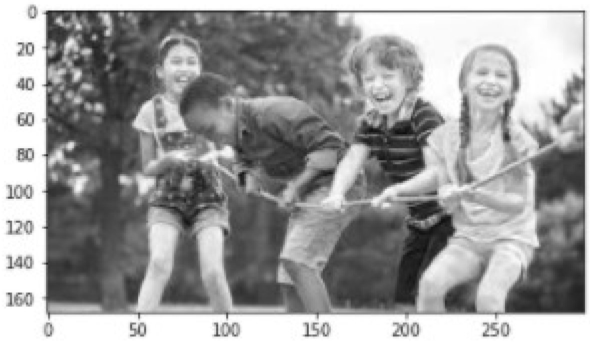


ב) אפשר לראות את הגאוסיאן באדום(יצרתי אותו בעזרת פונקציה מובנית מהאינטרנט) ואת ההתפלגות עם נתוני הממוצע וסטיית התקן שלנו בשאלה, ואנחנו רואים שההתפלגות הנורמלית מצליחה להתחקות די טוב אחרי הגאוסיאן אך לא בדיוק עליו.

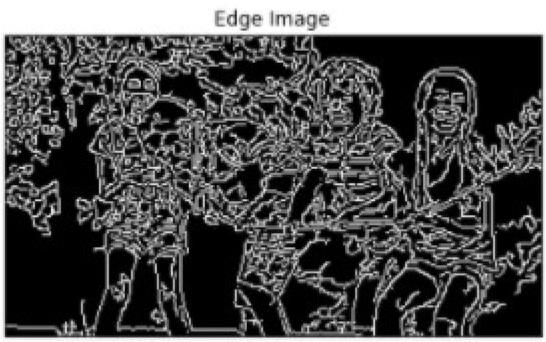


ג) קריאת התמונה 'kids' נעשית ע"י מתודת imshow של cv2 וע"מ שתוצג ראשית בצבע, עלי לפצל את אובייקט התמונה לגווני r, g, b ולכרוך אותם מחדש.

ובמצב אפור, של matplotlib של imshow ע"מ להציגה בגווני רמות אפור, כפי שראינו בהרצאה, מציגים עם `[:, :, 2]` במטריצת התמונה.

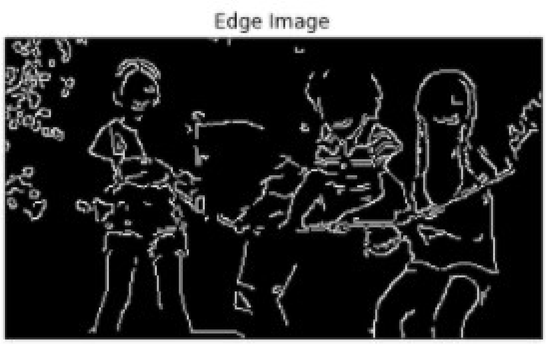


ד) את אלג' Canny edge detector ניתן כמובן להריץ על תמונה בגווני רמות אפור **בלבד,** ולכן נריץ אותו על התמונה השמאלית, האפורה.
הפרמטרים המשפיעים על טיב תוצאות האלגוריתם התמונה הם בעיקר שני ה thresholds של שלב ה Hysteresis Thresholding. פרמטרים אלה קובעים את סף הרגישות עבור הקצוות/ קווי המתאר אותם אנחנו רוצים למצוא ע"י האלג', וניתן לראות בתמונות הבאות את תוצאות ה"משחק" בפרמטרים אלו.



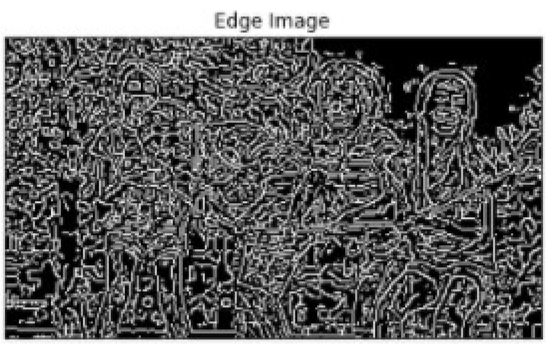
Threshold 1 – 50

Threshold 2 - 200



Threshold 1 – 300

Threshold 2 - 500



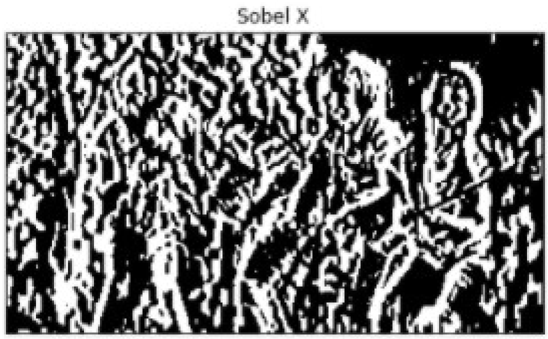
Threshold 1 – 20

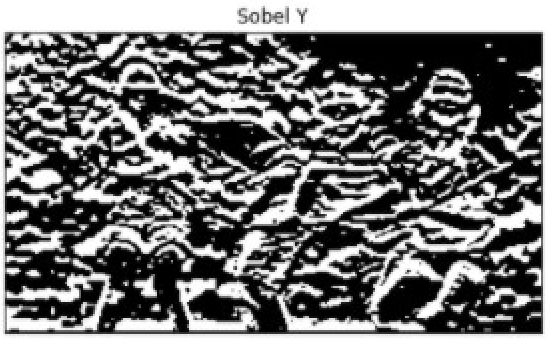
Threshold 2 - 30

כעת ברור כי, סף גבוה מאפשר לזהות פיקסלים חזקים ואילו סף נמוך מאפשר לזהות פיקסלים לא רלוונטיים. אם נגדיר שני ספים נמוכים, נקבך תמונה אשר מוצפת בקווי מתאר ולכן קשה מאוד לזהות את הפרטים בתמונה, תוצאה דומה נקבל גם עבור שני ספים גבוהים כאשר יותר מדי קווי מתאר יסווגו כלא רלוונטיים ויימחקו, וכך נקבל תמונה ובה קווי מתאר לא רציפים ומידע אובד.

לכן, ודאי שמתוך 3 התמונות עליהן הפעלנו את האלגוריתם עם פרמטרים שונים, עבור סף1 = 50 וסף שני = 200 , נתונים מאוזנים המאפשרים ריצה יעילה של האלג', נשיג את התוצאות הטובות ביותר.

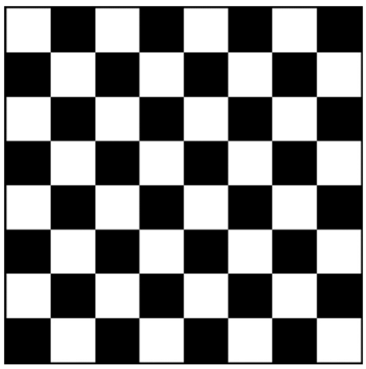
כעת נציג את התוצאות המתקבלות מאופרטור סובל:



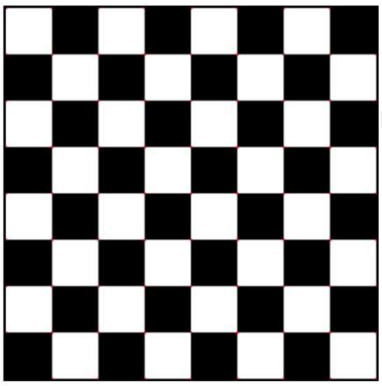


נראה שהתוצאות המתקבלות מאופרטור סובל הינן מאופיינות בקווי מתאר עבים וגסים יותר, כל פיקסל עובר פילטר אנכי ואופקי, ולכן החישוב אינו "יקר" ויעל. אולם, מסיבה זו ההערכה של הגרדיאנט הי די גסה, מה שמוביל לקווי מתאר מרוחים מעט.

ה) נשים לב לתמונה המקורית ולתוצאות עבור שני הסטים השונים של הפרמטרים.



```
cornerHarris(gray,2,3,0.9)
```



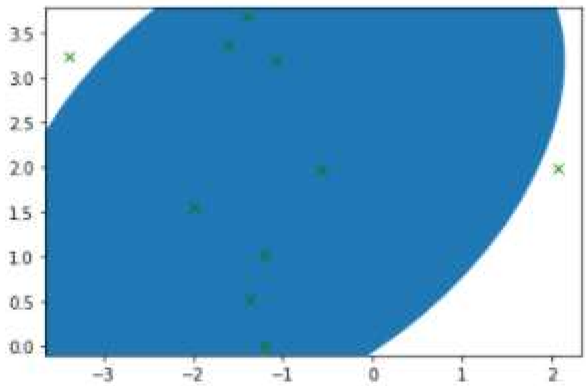
```
cornerHarris(gray,2,3,0.04)
```

עם threshold אופטימלי(ע"פ רוב, מאוד תלוי בתמונה) של [img[dst>0.01*dst.max()]

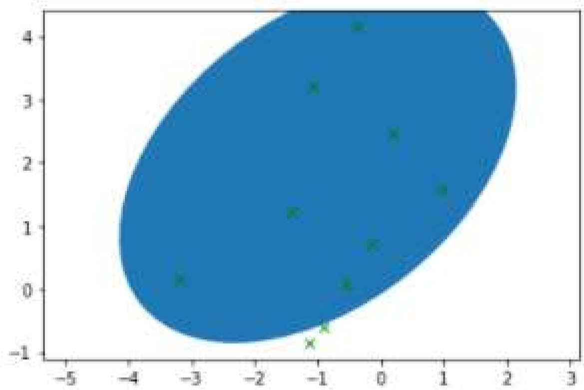
ולכן קל לראות שעבור הנתונים המיוצגים בתמונה השמאלית, מושגת התוצאה הטובה יותר- כאשר המשתנה החופשי של האלג' k הינו עדיין- קטן, מאפשר לזהות את הפינות ולסמן באדום.

שאלה 2

א) השתמשתי בפונק' הנתונה plot_cov_ellipse עם הנתונים בשאלה, ודגמתי 10 דגימות רנדומליות נורמליות ע"י multivariate_normal, עם המטריצה הקווריאנס והממוצע הנתונות.



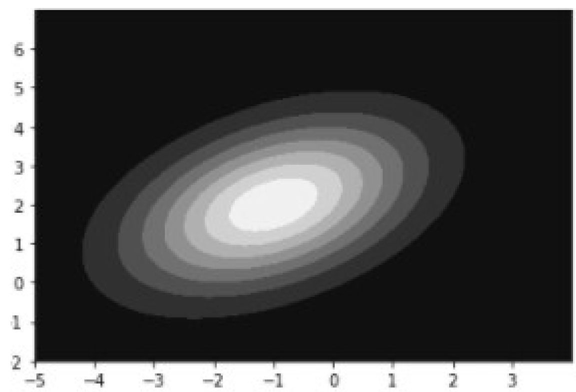
ב) את הדגימות שמרתי למערך a. כעת ע"פ הפונקציות המובנות של numpy עבור mean ו cov ניתן לשערך, לשחזר את נתוני ה mean והcov המקוריים(נשמרו תחת new_mean, new_cov).
משחק קצר עם מספר הדגימות מבהיר כי זהו הפרמטר בו תלויה רמת הדיוק של השיערוך.
כעת, נחשב את ההפרשים המנורמלים בין ה mean וה cov המקוריים והמשוערכים, ונציג את המרחק בין הערכים המשוערכים שסכום ההפרשים המרובעים.



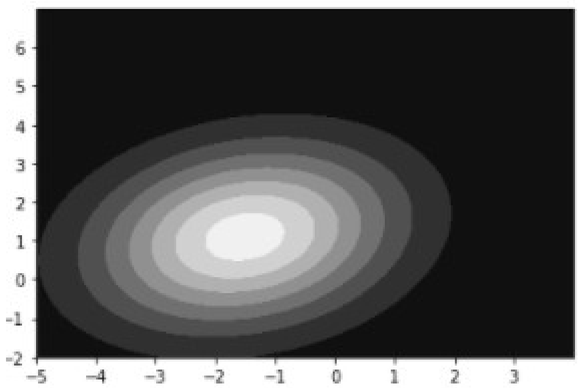
```
Estimated mean =
 [-0.76778452  1.21023291]
 Estimated cov =
 [[1.22721191 0.59969117]
 [0.59969117 2.73181863]]


cov diff = 1.5098712595146906
 mean dif = 0.823198689975034
Match rate is =
2.95736730348509
```

ג) הרעיון הוא ליצור גריד ע"פ ההמלצה בשאלה ובתוכו לדגום. נציג את הייצוג של כל תמונת התפלגות(מקורית ומשוערכת) וההפרש המנורמל של פונקציית ההתפלגות שלהם(pdf) בתמונה יהווה את ההבדל בין ההסתברויות.

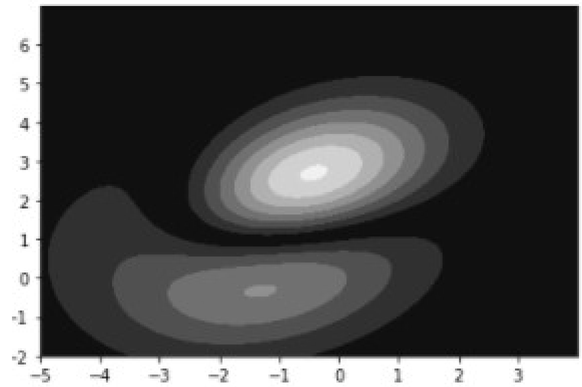


מקור



משוערך

הפרש



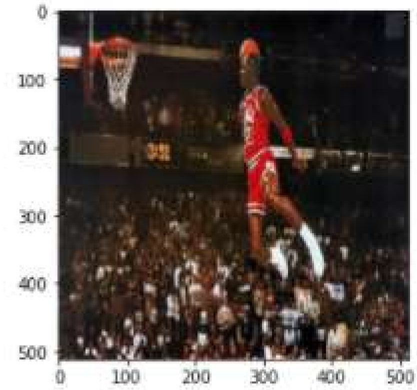
ד) כפי שהערתי בסעיף ב', הנתון המשפיע ביותר על מידת ההתאמה בין השיערוך לנתוני המקור הוא מס' הדגימות. נשים לב בגרף המתואר מטה, כבר החל מ1800 דגימות לערך ומעלה, רמת השיפור היא זניחה והתוצאה טובה מספיק. כפי שרצינו, בפלט המודפס ניתן לראות שככל שמסק הדגימות עולה, ההבדל יורד.
הנתונים מושגים ע"י ביצוע דגימות ברמה העולה בכל איטרציה, עד ל10000.

```
sampels: 10,    diff: 2.7087779144957373
sampels: 2000,  diff: 0.015441767841772765
sampels: 10000, diff: 0.0025343026586705734
```

<u>שאלה 3</u>

א) מימשתי פירמידה לפלסיאנית עבור התמונה של ג'ורדן, ב6 רמות. תחילה כמובן בונים פירמידה גאוסיינית, ובאמצעותה לאחר מכן בונים את הלפלסיאנית.

Original image



Below- Upper level Gaussian Pyramid

Level 5 of Laplacian Pyramid

Level 4 of Laplacian Pyramid

Level 3 of Laplacian Pyramid

Level 2 of Laplacian Pyramid

Level 1 of Laplacian Pyramid

ב) נבצע את התהליך מסעיף א עבור שתי התמונות, ובסופו נאחד בכל רמה את התמונות המתאימות מהפירמידות. בשלב האחרון, הreconstruct נמזג ביניהם באופן מותאם ומשתלב, לקבל תמונה ממוזגת, ולא משודכת.
משחק עם ערכי הרמות יראה שעבור 2/3 רמות מיזוג התמונות דומה יותר להדבקה/ שידוך של שני גזרי התמונות, וזו תוצאה לא מספקת. ואילו עבור 5/6 רמות ויותר, מתקיים מעין ניקוי רעשין מהאובייקטים המרכזיים עצמם, טשטוש והמיזוג גם כן לא מספק. עבור 4 רמות מושגת התוצאה האופטימלית, כפי שניתן לראות:

Image a

Image b

Mix

Mix reconstruct