יובל דלוס 305211880

מבוא למערכות לומדות – תרגיל 1

Warm-up - Algebra Recap

1. Calculate the projection of $v = (1, 2, 3, 4)$ on the vector $w = (0, -1, 1, 2)$.

1. נרצה להשתמש בנוסחה לחישוב הטלת וקטורים, כאשר v מוטל על w נקבל כי הווקטור המוטל יהיה:

$$\frac{< v, w >}{\| w \|^2} \cdot w = \frac{< (1,2,3,4), (0,-1,1,2) >}{\left(\sqrt{< (0,-1,1,2), (0,-1,1,2) >}\right)^2} \cdot (0,-1,1,2) = \frac{1 \cdot 0 + 2 \cdot (-1) + 3 \cdot 1 + 4 \cdot 2}{\left(\sqrt{1+1+4}\right)^2} \cdot (0,-1,1,2) =$$

$$= \frac{9}{6} \cdot (0,-1,1,2) = 1\frac{1}{2} \cdot (0,-1,1,2) = \left(0, -\frac{3}{2}, \frac{3}{2}, 3\right)$$

2. Calculate the projection of $v = (1, 2, 3, 4)$ on the vector $w = (1, 0, 1, -1)$.

2. פעם נוספת נשתמש בנוסחה:

$$\frac{< v, w >}{\| w \|^2} \cdot w = \frac{< (1,2,3,4), (1,0,1,-1) >}{\| (1,0,1,-1) \|^2} \cdot (1,0,1,-1) = \frac{1+3-4}{3} \cdot (1,0,1,-1) = 0 \cdot (1,0,1,-1) = (0,0,0,0)$$

3. Prove the angle between two non-zero vectors $v, w \in \mathbb{R}^m$ is ± 90 iff $\langle v, w \rangle = 0$.

3. נשתמש בזהות של $cos\theta$ ובפיתוח המלא של חישוב ווקטור הטלה:

$$\| v \| \, cos\theta \cdot \frac{u}{\| u \|} = \| v \| \cdot \frac{< v, u >}{\| v \| \cdot \| u \|} \cdot \frac{u}{\| u \|} = \frac{< v, u >}{\| u \|^2} \cdot u$$

$$cos\theta = \frac{< v, u >}{\| v \| \cdot \| u \|}$$

נתון לנו כי הזווית בין הווקטורים היא $90°$, לכן ניעזר בזהות של קוסינוס ונקבל:

$$\cos(\pm 90) = 0 = \frac{< v, u >}{\| v \| \cdot \| u \|} \Longrightarrow < v, u > = 0$$

מצד שני, נתון לנו כי $< v, u > = 0$, ושני הווקטורים שונים מ-0, לכן נקבל שווקטור ההטלה יהיה

$$\frac{< v, u >}{\| u \|^2} \cdot u = 0 \cdot u = 0 \in \mathbb{R}^n$$

וזה קורה רק כשהווקטורים אנכים זה לזה.

4. Prove that Orthonormal matrices are isometric transformations. That is let $T : V \mapsto W$ be some linear transformation and A the corresponding matrix. Then if A is orthonormal then $\forall x \in V \quad ||Ax||_2 = ||x||_2$.

.4

$$\|x\|_2 \overset{\text{def}}{\triangleq} \left(\sum_{i=1}^{m} |x_i|^2\right)^{\frac{1}{2}} \overset{\text{def}}{\triangleq} \left(\sum_{i=1}^{m} < x_i|x_i >\right)^{\frac{1}{2}} = \left(\sum_{i=1}^{m} < x_i \mid I_M x_i >\right)^{\frac{1}{2}} \overset{\text{def}}{\triangleq} \left(\sum_{i=1}^{m} < x_i|A^T A x_i >\right)^{\frac{1}{2}} \overset{\text{Orthonormal}}{\triangleq}$$

$$= \left(\sum_{i=1}^{m} < Ax_i|Ax_i >\right)^{\frac{1}{2}} = \left(\sum_{i=1}^{m} |Ax_i|\right)^{\frac{1}{2}} = \|Ax\|_2$$

SVD

5. Assume A is invertible. Write a formula for the inverse of A using only the matrices U, D, V where UDV^T is an SVD decomposition of A. Many learning algorithm implementations require calculating the inverse of a matrix. Explain why knowing the SVD decomposition of matrix is usefull in this context.

.5 נבחן ראשית מה עושה פירוק ה-SVD. כאשר אנו מפרקים מטריצה לשלוש מטריצות SVD אנו בעצם מפרקים אותה ל-3 פעולות שונות, סיבוב, מתיחה, וסיבוב חזרה לכיוון המקורי. לשם כך, בכדי לחשב את ההופכי של מטריצת A נרצה לבצע את הפעולות בסדר הפוך, אזי לסובב, לכווץ ולסובב חזרה, לשם כך נגדיר

$$A^{-1} = VD^{-1}U^T$$

כאשר

$$D^{-1} = \begin{bmatrix} \frac{1}{d_{11}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{d_{nn}} \end{bmatrix}$$

נוכיח את הטענה הזו:

$$AA^{-1} = (UDV^T)(VD^{-1}U^T) = UD(V^TV)D^{-1}U^T = U(DD^{-1})U^T = UU^T = I_n$$

כנדרש, כאשר נעזרנו בעובדה ש U ו- V הן מטריצות אורתוגונליות ולכן מקיימות $U^TU = I_n$.

לדעת את ה-SVD יכול מאוד להאיץ חישובים של מטריצות, בעוד חישוב של מטריצה הופכית כולל פעולות רבות ויכול לעלות $O(n^3)$, חישוב של כל אחת ממטריצות המשוחלפת וההופכית-מספרית של מטריצות ה-SVD יעלו $O(n)$ בלבד, כאשר n הוא אורך הווקטור הארוך ביותר במטריצה.

יובל דלוס 305211880

6. Find an SVD of

$$C = UDV^T = \begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix}$$

I.e., find matrices U, D, V^T where U, V are orthogonal matrices and D is diagonal.

Do the following steps:

- Calculate C^TC.
- Deduce V and D (hint: use the *Eigenvalues Decomposition*; You can either use the function *numpy.linalg.eig* in python or refresh your memory and do it manually).
- Find U using the equality $CV = UD$.

.6

- $C^T = \begin{pmatrix} 5 & -1 \\ 5 & 7 \end{pmatrix}$

$$C^TC = \begin{pmatrix} 5 & -1 \\ 5 & 7 \end{pmatrix}\begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix} = \begin{pmatrix} 26 & 18 \\ 18 & 74 \end{pmatrix}$$

אבל בנוסף נקבל כי

$$C^TC = VD^TU^TUDV^T = VD^TDV^T$$

כאשר המעבר השני התאפשר מכיוון ש-U מטריצה אורתוגונלית.

כעת, יש לנו מטריצה ריבועית בין מטריצה אורתוגונלית והמשוחלפת שלה, נמצא ו״ע וע״ע.

$$\det(C^TC - \lambda \mathrm{I_n}) = det\begin{pmatrix} 26-\lambda & 18 \\ 18 & 74-\lambda \end{pmatrix} = \lambda^2 - 100\lambda + 1{,}600 = (\lambda - 20)(\lambda - 80)$$

$$V_{\lambda=20} = \begin{pmatrix} 6 & 18 \\ 18 & 54 \end{pmatrix} = \begin{pmatrix} 6 & 18 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 3 \\ 0 & 0 \end{pmatrix} \Rightarrow v_1 = \frac{1}{\sqrt{10}} \cdot \begin{pmatrix} 1 \\ 3 \end{pmatrix}$$

$$V_{\lambda=80} = \begin{pmatrix} -54 & 18 \\ 18 & -6 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 18 & -6 \end{pmatrix} = \begin{pmatrix} 3 & -1 \\ 0 & 0 \end{pmatrix} \Rightarrow v_2 = \frac{1}{\sqrt{10}} \cdot \begin{pmatrix} 3 \\ -1 \end{pmatrix}$$

כאשר המקדם הוא לצרכי נרמול
לכן:

$$V^T = \frac{1}{\sqrt{10}}\begin{pmatrix} 1 & 3 \\ 3 & -1 \end{pmatrix}$$

$$D = \begin{pmatrix} \sqrt{80} & 0 \\ 0 & \sqrt{20} \end{pmatrix}$$

כעת בכדי למצוא את U ניעזר בעובדה כי $CV = UD$:

$$CV = \frac{1}{\sqrt{10}}\begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix}\begin{pmatrix} 1 & 3 \\ 3 & -1 \end{pmatrix} = \sqrt{10}\begin{pmatrix} 2 & 1 \\ 2 & -1 \end{pmatrix} = UD = U \cdot \sqrt{10}\begin{pmatrix} \sqrt{8} & 0 \\ 0 & \sqrt{2} \end{pmatrix} \Rightarrow$$

$$\Rightarrow U = \begin{pmatrix} \sqrt{1/2} & \sqrt{1/2} \\ \sqrt{1/2} & -\sqrt{1/2} \end{pmatrix}$$

יובל דלוס 305211880

7. (Power Iteration) In this section we will implement an algorithm for SVD decomposition, we will use the relation between SVD of A to EVD of A^TA that we saw in recitation. For some $A \in M_{m\times n}(\mathbb{R})$, define $C_0 = A^TA$.

Let $\lambda_1, \lambda_2, ...\lambda_n$ be the eigenvalues of C_0, with the corresponding eigenvectors $v_1, v_2, ...v_n$, ordered such that $\lambda_1 \geqslant \lambda_2 \geqslant ... \geqslant \lambda_n$.

Assume $\lambda_1 > \lambda_2$, where λ_1 is the largest eigenvalue and λ_2 is the second-largest one.

Define $b_{k+1} = \frac{C_0 b_k}{\|C_0 b_k\|}$, and initialize b_0 randomly.

Show that: $\lim_{k\to\infty} b_k = v_1$

Hint: use EVD decomposition of C_0 and represent b_0 accordingly. You can assume that $b_0 = \sum_{i=0}^{n} a_i v_i$, where $a_1 \neq 0$. As b_0 is initialized randomly, the probability of $a_1 = 0$ is zero.

7. ראשית נגדיר את המטריצה שלנו

$$A = UDV^T$$
$$C_0 = A^TA = VD^TDV^T$$

לכן נוכל להתייחס לו״ע של D^TD.

נביט על הווקטור ההתחלתי שלנו, נוכל לרשום אותו כצירוף לינארי של כל השאר:

$$b_0 = a_1v_1 + a_2v_2 + \cdots + a_nv_n$$

ונסתכל על התהליך הנדרש מאיתנו:

$$b_1 = \frac{C_0b_0}{\|c_0b_0\|} = \frac{1}{\|c_0b_0\|}\cdot C_0b_0 = \frac{1}{\|c_0b_0\|}\cdot C_0(a_1v_1 + a_2v_2 + \cdots + a_nv_n) \overset{(*)}{=} \frac{1}{\|c_0b_0\|}\cdot(a_1\lambda_1v_1 + a_2\lambda_2v_2 + \cdots + a_n\lambda_nv_n)$$

כאשר (*) נובע מכך שכל v_i הוא ו״ע, ולכן הוא אינו משתנה בפרט בהכפלה בע״ע המתאים לו.

באותו אופן:

$$b_2 = \frac{C_0b_1}{\|c_0b_1\|} = \frac{1}{\|c_0b_1\|}\cdot C_0b_1 = \frac{1}{\|c_0b_0\|}\cdot\frac{1}{\|c_0b_1\|}\cdot C_0(a_1\lambda_1v_1 + a_2\lambda_2v_2 + \cdots + a_n\lambda_nv_n) =$$
$$= \frac{1}{\|c_0b_0\|\cdot\|c_0b_1\|}\cdot(a_1\lambda_1^2v_1 + a_2\lambda_2^2v_2 + \cdots + a_n\lambda_n^2v_n)$$

ובאופן כללי:

$$b_k = \frac{1}{\prod_{i=0}^{k-1}\|c_0b_i\|}\cdot\left(a_1\lambda_1^kv_1 + a_2\lambda_2^kv_2 + \cdots + a_n\lambda_n^kv_n\right) =$$

$$= \frac{1}{\prod_{i=0}^{k-1}\|c_0b_i\|}\cdot\lambda_1^k\cdot\left(a_1v_1 + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^kv_2 + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^kv_n\right)$$

ומכיוון שהנחנו כי λ_1 הוא הגדול ביותר, נותר לנו להסיק כי $\frac{\lambda_i}{\lambda_1} < 0$ לכל i, ולכן

$$li\,m_{k\to\infty}\left(\frac{\lambda_i}{\lambda_1}\right)^k = 0$$

לכן

$$\lim_{k\to\infty} b_k = lim_{k\to\infty} \frac{1}{\prod_{i=0}^{k-1}\|c_0 b_i\|} \cdot \lambda_1^k \cdot \left(a_1 v_1 + a_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k v_2 + \cdots + a_n \left(\frac{\lambda_n}{\lambda_1}\right)^k v_n\right) = \frac{1}{\prod_{i=0}^{k-1}\|c_0 b_i\|} \cdot \lambda_1^k \cdot a_1 \cdot v_1$$

וקיבלנו את הווקטור המבוקש עד כדי ניפוח בסקלר.

Multivariate Calculus

8. Let $x \in \mathbb{R}^n$ be a fixed vector and $U \in \mathbb{R}^{n\times n}$ a fixed orthogonal matrix. Calculate the Jacobian of the function $f : \mathbb{R}^n \to \mathbb{R}^n$:

$$f(\sigma) = U diag(\sigma) U^T x$$

Here $diag(\sigma)$ is an $n \times n$ matrix where $diag(\sigma)_{ij} = \begin{cases} \sigma_i & i = j \\ 0 & i \neq j \end{cases}$

8. $f(\sigma)$ הנתונה לנו היא בעצם פירוק EVD של מטריצה אחרת A כלשהי, מכיוון שהיא מורכבת משתי מטריצות אורתוגונליות ובניהן מטריצה ריבועית אלכסונית מלאה בע״ע, לכן נוכל לכתוב:

$$f(\sigma) = Ax$$

ובתרגול למדנו כי עבור מטריצות כאלה:

$$J_\sigma(f) = A = Udiag(U^T x)$$

9. Use the chain rule to calculate the gradient of h:

$$h(\sigma) = \frac{1}{2}\|f(\sigma) - y\|^2$$

9. ניעזר בכלל השרשרת שראינו בתרגול.

נגדיר:

$$Z(x) = \frac{1}{2}\|x\|^2$$

$$g(\sigma) = f(\sigma) - y = Ax - y$$

וכעת יש לנו מצב של $h(\sigma) = Z\big(g(\sigma)\big)$ ונוכל להפעיל את כלל השרשרת:

$$J_{g(\sigma)}(Z) = \frac{1}{2} \cdot 2(f(\sigma) - y)^T = (f(\sigma) - y)^T$$

$$J_\sigma(g) = A = Udiag(U^T x)$$

$$J_{h(\sigma)} = (f(\sigma) - y)^T Udiag(U^T x)$$

10. Calculate the Jacobian of the softmax function (initial steps can be found in recitation file):

$$g(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

10. ראינו בתרגול כי עבור

$$h = \sum_{k=1}^{N} e^{a_k}, \qquad g_i = e^{a_k}$$

$$\frac{\partial S_i}{\partial a_j} = \frac{\partial}{\partial a_j} \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}} = \frac{\partial}{\partial a_j} \frac{g_i}{h}$$

ועבור $i = j$

$$\frac{\partial}{\partial a_j} \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}} = \frac{e^{a_i}\left(\sum_{k=1}^{N} e^{a_k}\right) - e^{a_i} e^{a_j}}{\left(\sum_{k=1}^{N} e^{a_k}\right)^2} = \frac{e^{a_i}}{\left(\sum_{k=1}^{N} e^{a_k}\right)} \cdot \frac{\left(\sum_{k=1}^{N} e^{a_k}\right) - e^{a_j}}{\left(\sum_{k=1}^{N} e^{a_k}\right)} = S_i(1 - S_j)$$

כעת, עבור $i \neq j$

$$\frac{\partial}{\partial a_j} \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}} = \frac{0 \cdot \left(\sum_{k=1}^{N} e^{a_k}\right) - e^{a_i} e^{a_j}}{\left(\sum_{k=1}^{N} e^{a_k}\right)^2} = \frac{-e^{a_i}}{\left(\sum_{k=1}^{N} e^{a_k}\right)} \cdot \frac{e^{a_j}}{\left(\sum_{k=1}^{N} e^{a_k}\right)} = -S_i \cdot S_j$$

וסה"כ:

$$J_{S(a)_{i,j}} = \begin{cases} S_i(1 - S_j), & i = j \\ -S_i \cdot S_j, & i \neq j \end{cases}$$

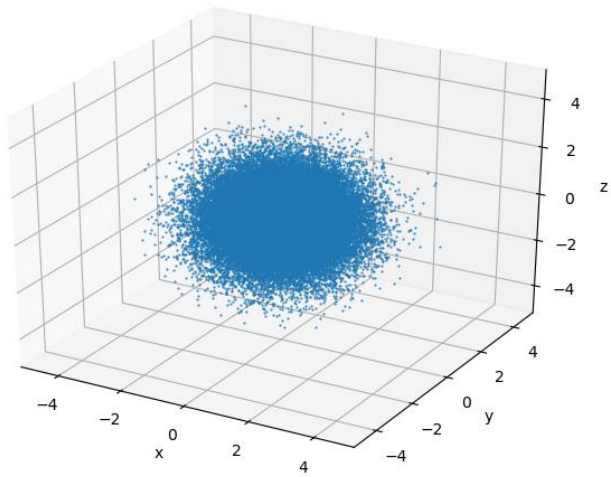
305211880 יובל דלוס

Multivariate Gaussian- practical question

In this question we will examine the three-dimensional case. We want to show how linear transformations affect the data set and in result the covariance matrix. First we will generate random points with mean values at the origin and unit variance $\sigma(x) = \sigma(y) = \sigma(z) = 1$.

11. Download the file 3d_gaussian.py. Use the identity matrix as the covariance matrix to generate random points and than plot them (with the given function).

.11

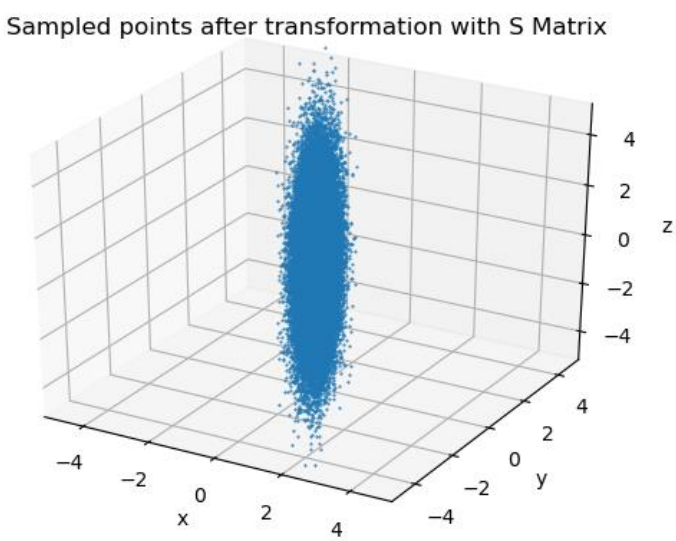


12. Transform the data with the following scaling matrix:

$$S = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

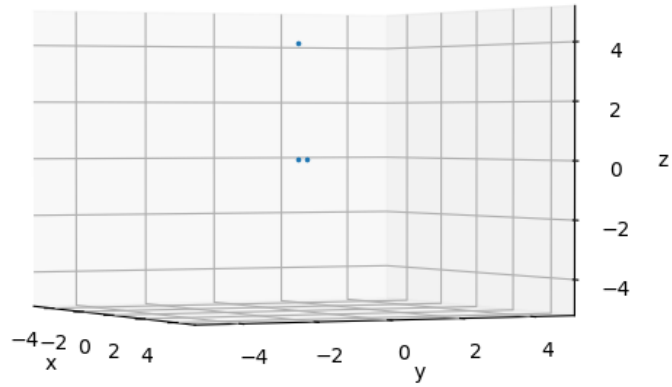
Plot the new points. What does the covariance matrix look like now (both analytically and numerically)?

.12



יובל דלוס 305211880

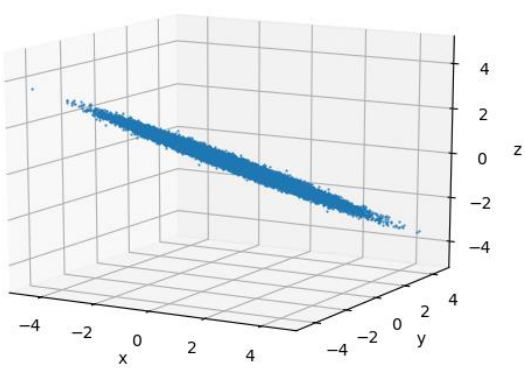
Covariance Matrix after transformation with S Matrix



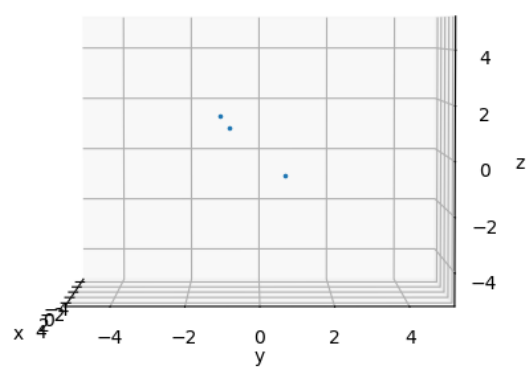

```
[[ 9.97494222e-03  3.67582784e-04  5.48624144e-05]
 [ 3.67582784e-04  2.49428528e-01 -1.34318899e-05]
 [ 5.48624144e-05 -1.34318899e-05  4.01919452e+00]]
```

13. Multiply the scaled data by random orthogonal matrix. Plot the new points. What does the covariance matrix look like now?

Transformed Matrix after multiplying in orthogonal matrix



Covariance Matrix after multiplying in orthogonal matrixx

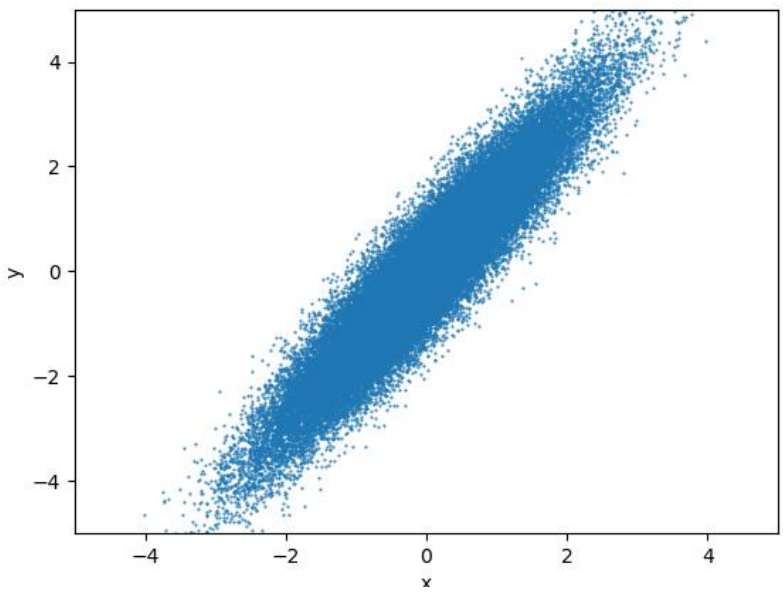


-

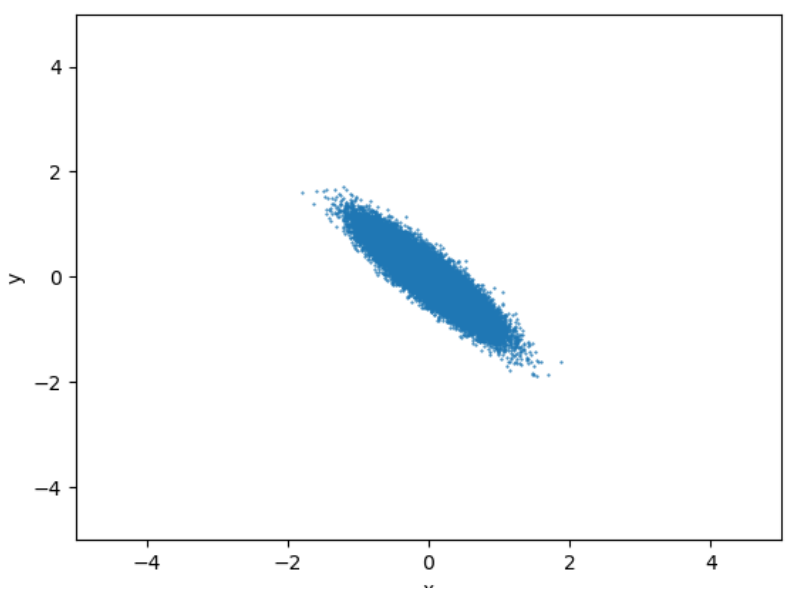
```
[[ 2.46100185 -1.06926436  1.5543838 ]
 [-1.06926436  0.74342985 -0.83012346]
 [ 1.5543838  -0.83012346  1.08241762]]
```

305211880 יובל דלוס

14. In recitation we claimed that the marginal distribution of a gaussian is still gaussian. Plot the projection of the data to the x, y axes. What does it look like? Add the plot to the submission.



15. In recitation we claimed that the conditional distribution of a gaussian is still gaussian. Only for points where $0.1 > z > -0.4$: Plot the projection of the points to the x, y axes. What does it look like? Add the plot to the submission.



יובל דלוס 305211880

16. You are given 100000 sequences of 1000 coin tosses, arranged in a matrix, "data", of 100000 rows and 1000 columns. To generate the data use the commands:

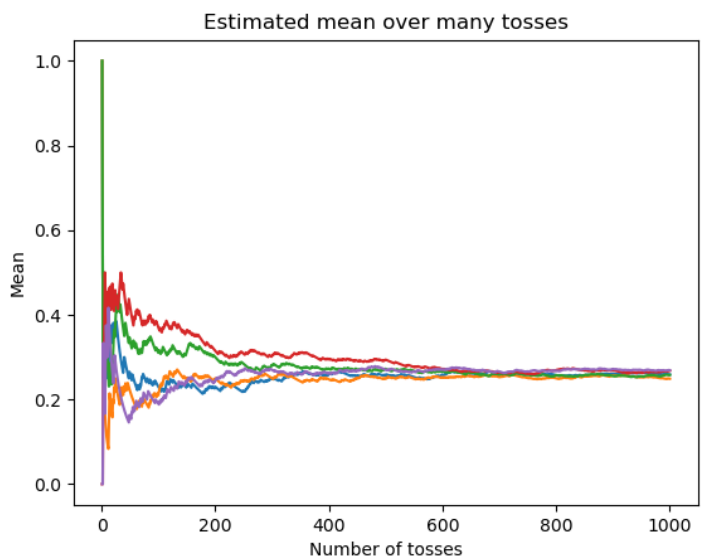
- import numpy
- data = numpy.random.binomial(1, 0.25, (100000,1000))

Define a variable "epsilon" which gets the values $[0.5, 0.25, 0.1, 0.01, 0.001]$.

(a) For the first 5 sequences of 1000 tosses (the first 5 rows in "data"), plot the estimate $\overline{X}_m = \frac{1}{m}\sum_{i=1}^{m} x_i$ as a function of m (i.e the mean of all tosses up to m). 1 figure with 5 plots (each row in a different color). What do you expect to see in this plot as m grows?

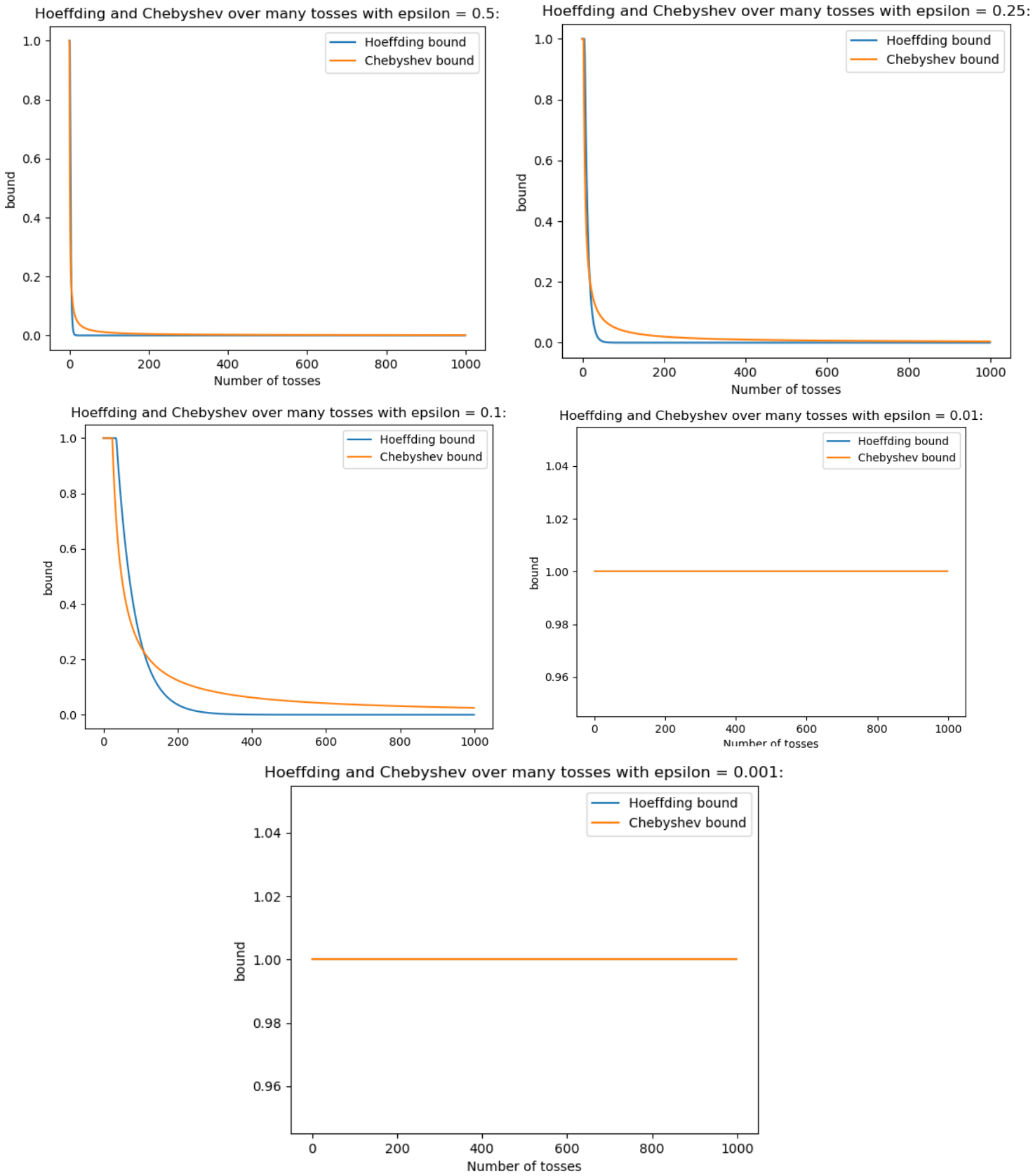
.16

.a בהתאם לציפייה, ניתן לראות בגרף שככל שנבדוק יותר דגימות כך התוחלת תלך ותתכנס לערך האמיתי שלה, במקרה זה 0.25.



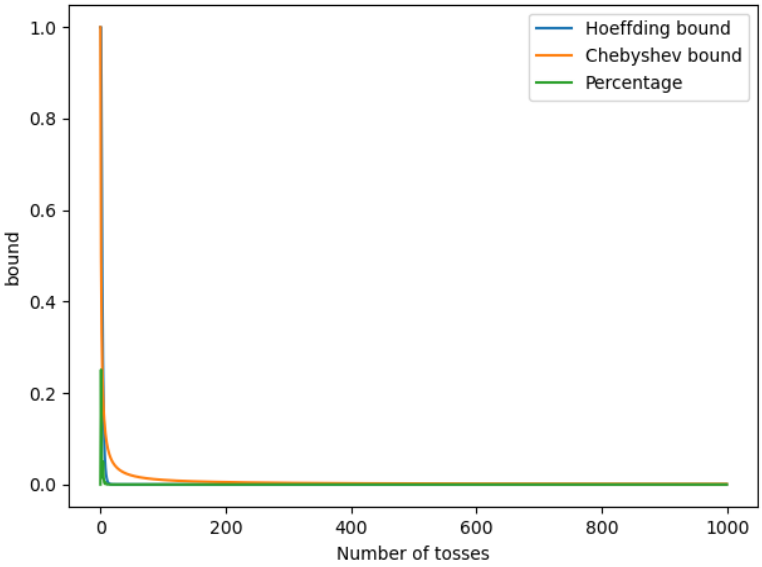
יובל דלוס 305211880

(b) For each bound (Chebyshev and Hoeffding seen in class) and for each ϵ, plot the upper bound on $P_{X_1,\ldots,X_m}\left(\left|\overline{X}_m - \mathbb{E}[X]\right| \geqslant \epsilon\right)$ (derived in class) as a function of m (where m ranges from 1 to 1000). 5 figures with 2 plots each (mention in the title of each plot what is ϵ and use a different color for each bound)[1].

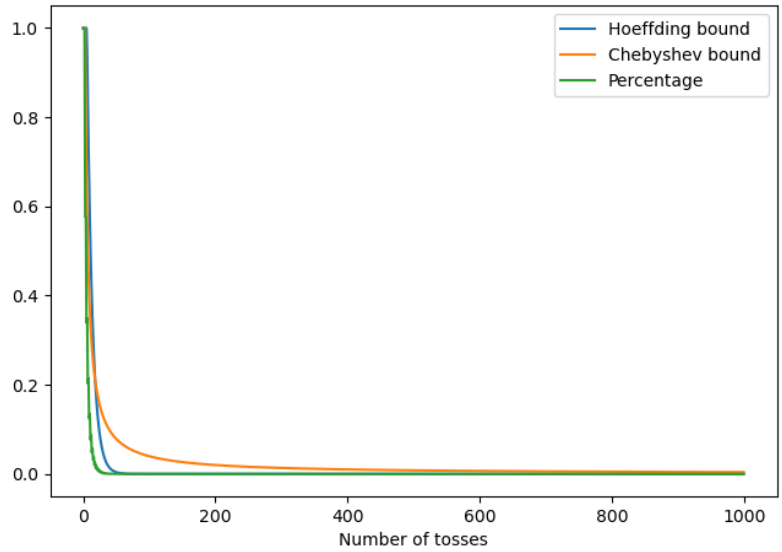


(c) You are now told that $p = 0.25$. On top of the figures from the previous question, plot the percentage of sequences that satisfy $\left|\overline{X}_m - \mathbb{E}[X]\right| \geqslant \epsilon$ as a function of m (now you know $\mathbb{E}[X] = p = 0.25$). What are you expecting to see in these plots? Explain.

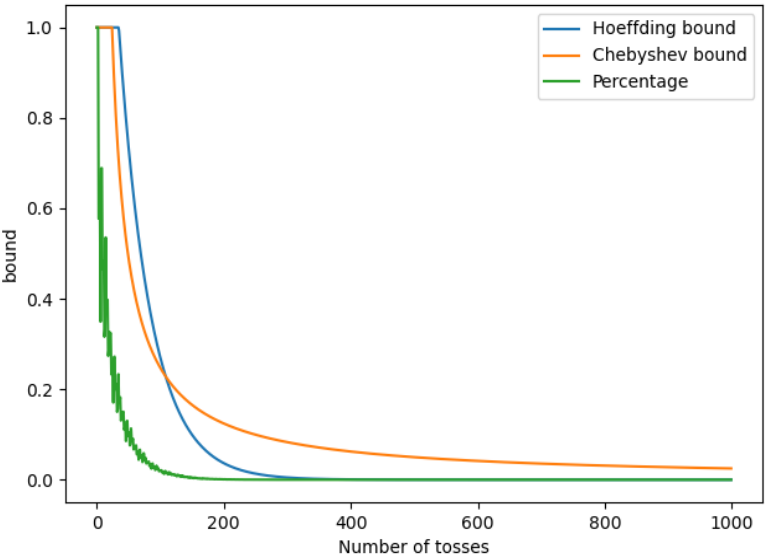
Percentage, Hoeffding and Chebyshev over many tosses with epsilon = 0.5:



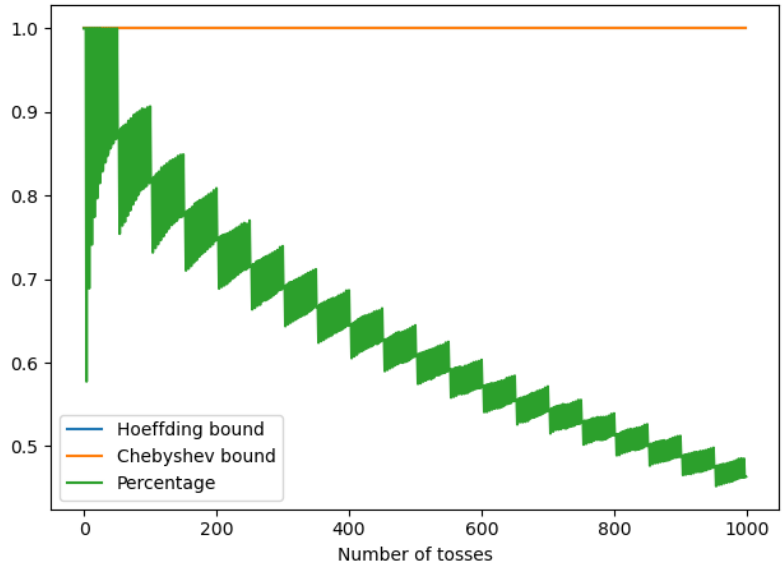
Percentage, Hoeffding and Chebyshev over many tosses with epsilon = 0.25:



Percentage, Hoeffding and Chebyshev over many tosses with epsilon = 0.1:



Percentage, Hoeffding and Chebyshev over many tosses with epsilon = 0.01:



Percentage, Hoeffding and Chebyshev over many tosses with epsilon = 0.001: