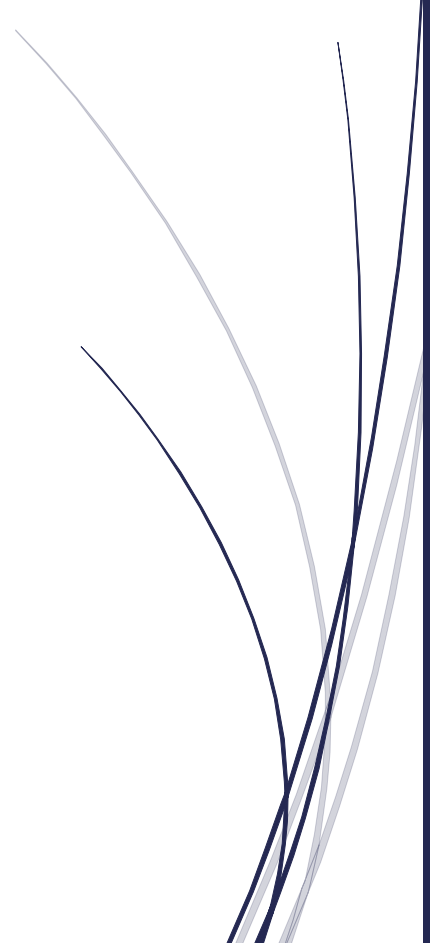


# תרגיל בית 3 – Generative Learning

למידה עמוקה – 097200

- הילה בן יוסף - 312142243
- עמית אברמוביץ' - 205810484
- שרון אנקרי - 200637742

\* קיבלנו את אישורו של גיא להגיש בשלשה



## הקדמה

### תיאור המשימה

בתרגיל זה נתבקשנו ליצור מודל ג'נרטיבי אשר מנסה למדל רשת היוצרת תמונות המחקות את מרחב הדאטה המקורי תוך שימוש בוקטור סמוי רציף ובדיד .

### הדאטה

השתמשנו בדאטה סט של [CelebA](#), אוסף תמונות גדול של מפורסמים המכיל תיוג של המפורסם ותכונות מסוימות נוספות המתארות את התמונה.

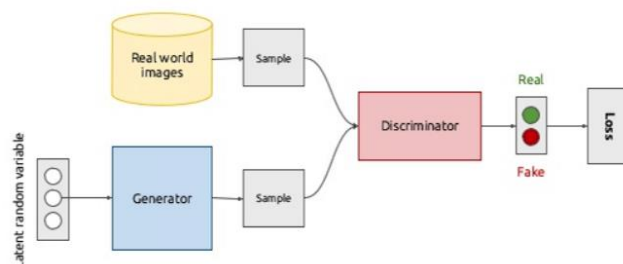
### המודל

בחרנו להשתמש במודל של GAN, שהוצג לראשונה על-ידי [Ian Goodfellow ב-2014](#) וכיום הוא אחד הפופולריים ביותר למשימות אלו.

במודל GAN, ישנן 2 רשתות נוירונים אשר מתחרות האחת בשנייה :

- Generator : אשר מנסה ליצור תמונות מזויפות המחקות את הדאטה-סט המקורי, ובעצם להטעות את ה-Discriminator. הרשת הזו מנסה למדל התפלגות התלויה בפרמטרים ותנסה לחקות את התפלגות הדאטה המקורי.
- Discriminator : אשר מקבל תמונות המיוצרות מה-Generator ותמונות אמיתיות ממרחב התמונות המקורי, ומנסה לקבוע מי מהן אמיתית ומי המזויפת.

פונקציית ההפסד (Loss Function) של המודל משקפת את אלמנט התחרות שבין שתי הרשתות. מצד אחד ישנו איבר המייצג את שיפור היכולת של ה-Generator לייצר תמונות מזויפות מוצלחות יותר, ומצד שני קיים איבר המייצג שיפור היכולת לזהות מי אמיתית ומי מזויפת. כך במקביל המערכת לומדת ומאפשרת לבסוף לקבל מודל ג'נרטיבי הבונה תמונות המחקות את הדאטה-סט שלנו בצורה טובה יותר.



איור 1 : ארכיטקטורת GAN כללית

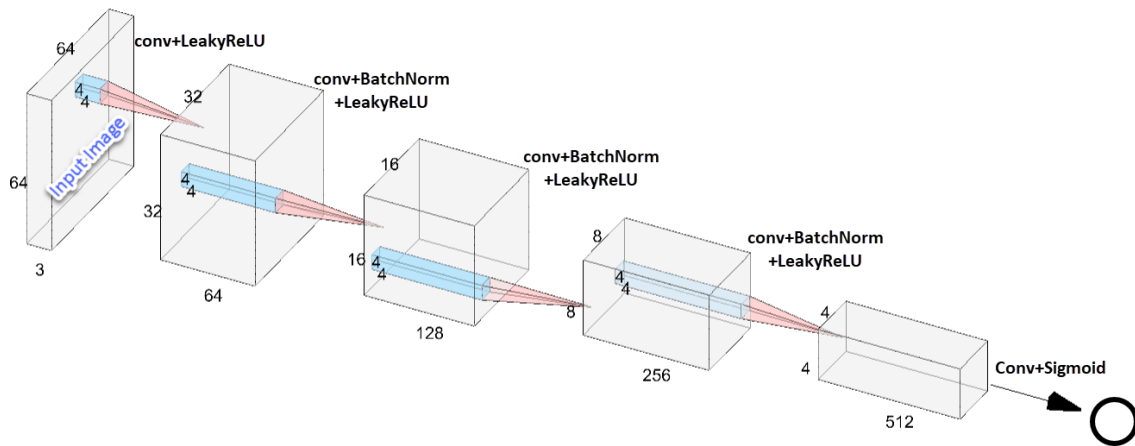
### ארכיטקטורת המודל

#### DISCRIMINATOR

בנינו את רשת קונבולוציה שתהווה את ה-Discriminator לפי הפירוט הבא :

Conv > LeakyReLU > Conv > BatchNorm > LeakyReLU > Conv > BatchNorm > LeakyReLU > Conv > BatchNorm > LeakyReLU > Conv>Sigmoid

כאשר מימדי הקונבולוציה מתוארים באיור 2 ונבחר שיפוע 0.2 עבור שכבות ה-LeakyReLU.



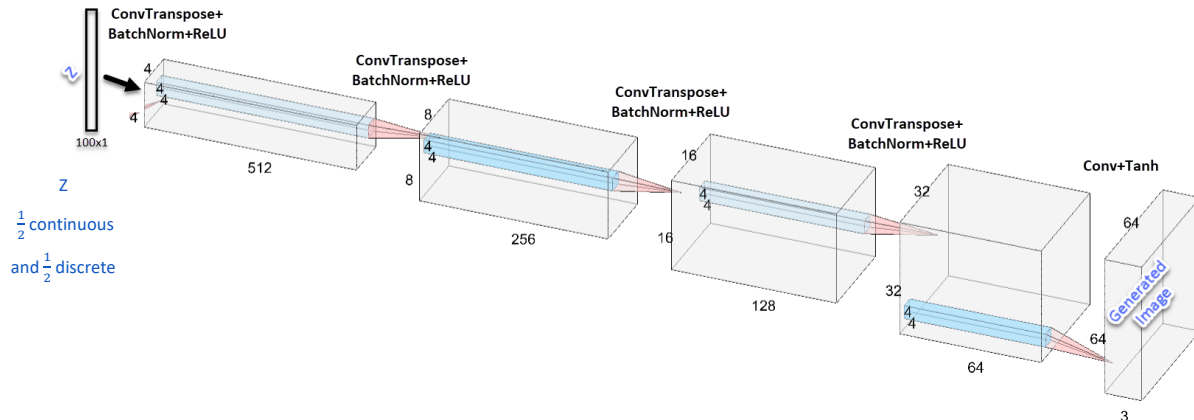
איור 2 : ה-Discriminator

## GENERATOR

בנינו את רשת הקונבולוציה שתהווה את ה-Generator לפי הפירוט הבא :

ConvTranspose>BatchNorm>ReLU> ConvTranspose>BatchNorm>ReLU> ConvTranspose>BatchNorm>ReLU>  
ConvTranspose>Tanh

נכניס וקטור  $z$  של רעש חצי דיסקרטי וחצי רציף אנו מאחדים ביניהם.



איור 3 : ה-Generator

## עיבוד מקדים לדאטה, האימון, בחירת פרמטרים

על התמונות ביצענו מספר טרנספורמציות בכדי לשפר את יעילות המודל ומהירותו – ביצענו שינוי גודל לתמונה ל-64x64 ובנוסף נרמול של התמונות.

מידלנו את הוקטור  $z$  של המרחב הסמוי כך שיכיל מידע רציף ומידע בדיד, היחס בין המידע הרציף והבדיד הינו 1:1. לאחר מכן, בכל איטרציה, נתנו למודלים להתאמן בהתאם לפונקציית ההפסד שתוארה קודם במטרה לשפר את ה-Generator בייצור תמונות מזויפות שה-Discriminator מנסה לזהות.

כמו כן, בחרנו את ההיפר פרמטרים הבאים :

Num\_epochs = 50, Learning rate=0.0002, Latent\_size = 100, Batch\_size = 128,  
Hidden\_layers = 64, Betas=0.5, 0.999, Optimization: Adam

## ניסיונות תוצאות וגרפים:

כחלק ממקצה השיפורים בחנו מס' ניסיונות תוך שינוי הפרמטרים השונים:

Latent\_size = 100, 200

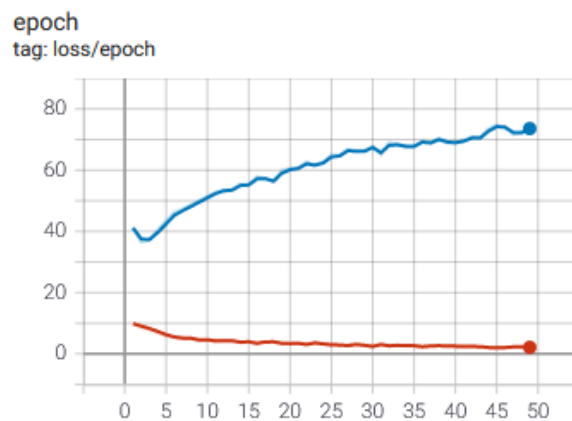
num\_epochs = 20, 50, 100

discrete latent size = 10, 20, 40, 50

continuous latent = 90, 80, 60, 50

כאשר בכולם יכלנו לזהות את שיפור היכולת של ה-Generator לייצר תמונות מזויפות מוצלחות יותר ויותר, אך התוצאה הטובה ביותר קיבלנו עבור הפרמטרים שבחרנו לעיל.

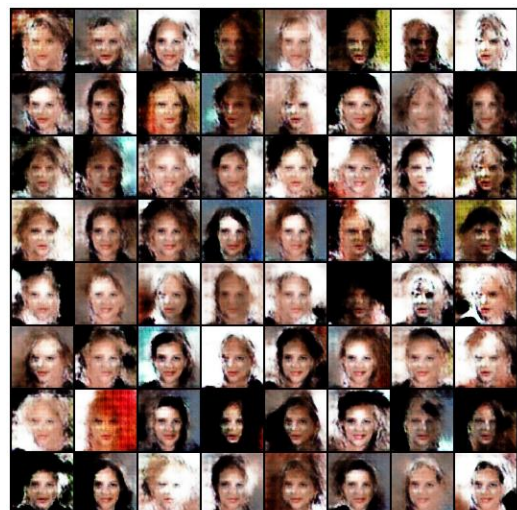
כאשר גרף ההפסד שהתקבל:



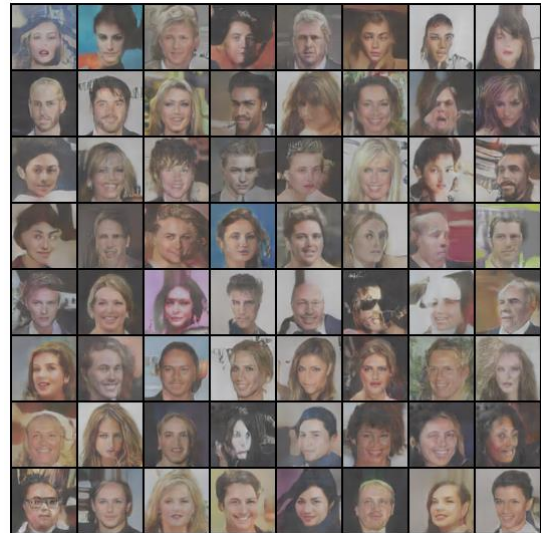
כאשר הגרף הכחול מייצג את ההפסד ה generator והגרף האדום את ההפסד של ה- discriminator

התמונות שייצרו:

התמונות אחרי שהמודל התאמן אפוק אחד –



## התמונות אחרי שהמודל התאמן 50 אפוקים



## סיכום

לסיכום, בנינו מודל GAN המורכב משתי רשתות נוירונים Generator ו-Discriminator המתחרות זו בזו – האחת מזייפת דוגמאות ממרחב הדאטה המקורי והשנייה מנסה לזהות האם התמונה מזויפת או לא. במהלך האימון שתי הרשתות לומדות ומשתפרות ביכולותיהן כך שלבסוף השתמשנו ב-Generator לייצר תמונות מזויפות. מבחינה ויזואלית ניתן לראות שאכן המודל למד לייצר תמונות שנראות מזויפות בדומה למרחב הדאטה המקורי.