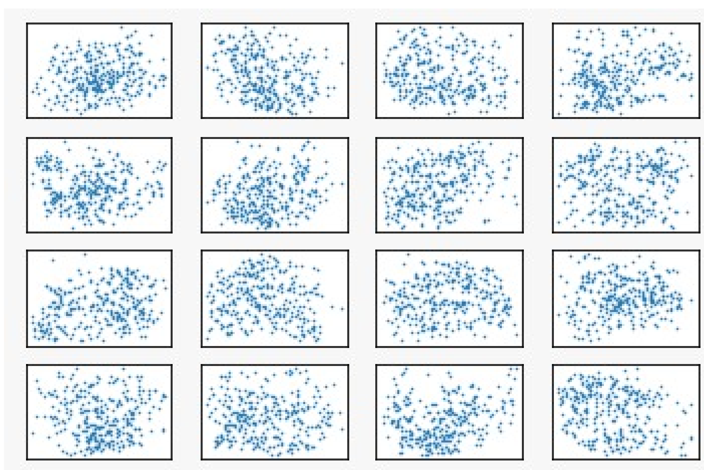
פתרון תרגיל מספר 3 - רשתות נוירונים בתמונות

שם: מיכאל גרינבאום, **ת.ז:** 211747639, **שם:** ניבי שנקר, **ת.ז:** 207227687

<u>חלק מעשי:</u>

1. כיוון שהארכיטקטורה מהתרגיל הקודם עבדה טוב החלטנו להישאר איתה, ואת הרשת הזאת אימנו כנדרש.

2. הגרפים הבאים מציגים את התמונות המקודדות, כאשר בכל גרף בחרנו שני מימדים שונים מתוך ה-latent space להיות הצירים. ניתן לראות שבכל הגרפים התמונות לא מפוזרות בכל המרחב.



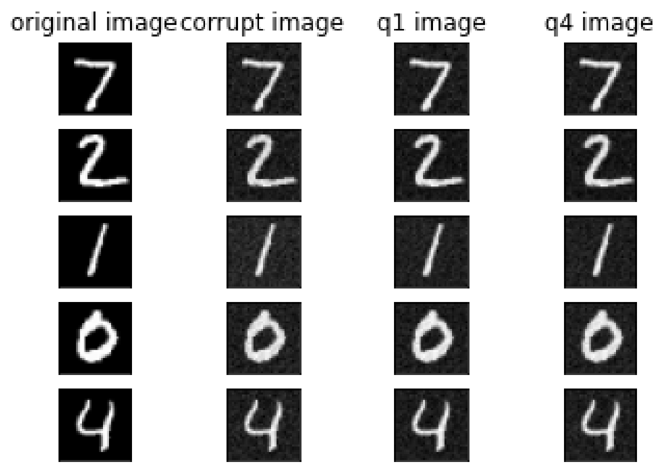
3. נשים לב כי מרחב התמונות המקודדות $C = \{ Encoder(I) \mid I \in Dataset \} \neq \{0,1\}^d$.
בהינתן $x \notin Dataset$ כך ש- $Encoder(x) \notin C$, ייתכן שעדיין יתקיים כי
$Decoder\big(Encoder(x)\big) = x$ כיוון שלא היה אילוץ על ה- AE למנוע שחזור של תמונות לא מהדאטאסט. דבר זה יגרום לבעיות בשחזור תמונות מלוכלכות או רועשות.

4. פתרון לבעיה שהוצעה בסעיף הקודם היא להכריח את C לקיים $C = \{0,1\}^d$. נוכל לעשות זאת על ידי כך שנדאג שהקידוד של כל $z \in C$ יתפלג בצורה יוניפורמית על כל $\{0,1\}^d$. נוכל לעשות זאת על ידי הוספת אילוץ על התוחלת, שונות, קורטוסיס וסקיונס כך שיהיה זהה לאלו של ההתפלגות היוניפורמית.
האילוץ יבוא לידי ביטוי בפונקציית הלוס שתראה בצורה הבאה:

$$L(img) = \|AE(img) - img\| + (\mathbb{E}[encoder(img)] - 0.5)^2 + \left(Var[encoder(img)] - \frac{1}{12}\right)^2 + (Skewness[encoder(img)] - 0)^2 + \left(Kurtosis[encoder(img)] - \left(3 - \frac{6}{5}\right)\right)^2$$

5. a. denoising:

כך נראות התוצאות של הרשת המתוקנת לעומת הרשת הרגילה-



אפשר לשים לב שהתמונות שיצאו לאחר הוספת הפתרון משאלה 4 (העמודה הימנית) אכן פחות מורעשות. בגלל שהרעש חלש, כפי שביקשו בהוראות, נראה שגם הרשת ללא האילוצים שהוספנו הצליחה לנקות מעט רעשים אך עדיין ניתן להבחין שבצורה פחות טובה.

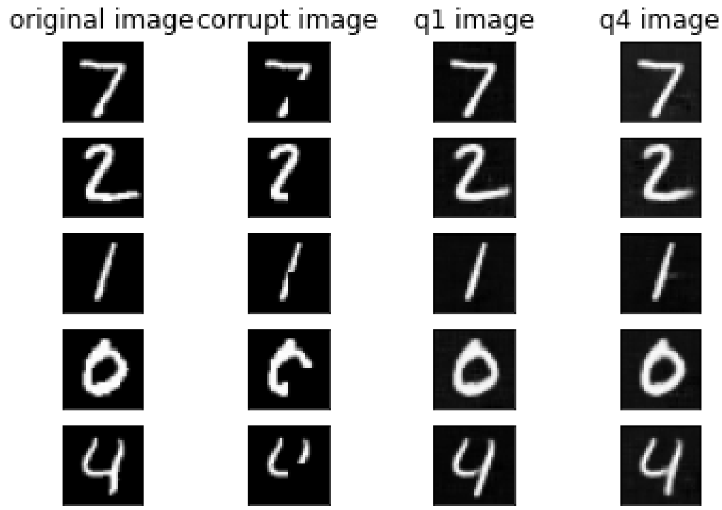
הלוס שאיתו פתרנו את המשימה הוא-

$$\max_{\mathrm{I}} P(I \mid I_C) = \max_{\mathrm{I}} P(\, I_C \mid I\,) \cdot P(I) = \max_{\mathrm{I}} [\log P(\, I_C \mid I\,) + \log P(I)]$$
$$= \max_{\mathrm{I}} \left[\log e^{-\frac{\|I - I_C\|^2}{2 \cdot \sigma^2}} + \log e^{-\frac{\|AE(I) - I\|}{T}}\right]$$
$$= \max_{\mathrm{I}} \left[-\frac{\|I - I_C\|^2}{2 \cdot \sigma^2} + -\frac{\|AE(I) - I\|}{T}\right] = -\min_{\mathrm{I}} \left[\frac{\|I - I_C\|^2}{2 \cdot \sigma^2} + \frac{\|AE(I) - I\|}{T}\right]$$

כלומר $\mathrm{L}(I, I_C) = \frac{\|I - I_C\|^2}{2 \cdot \sigma^2} + \frac{\|AE(I) - I\|}{T}$.

:demasking .b

המסיבה שבחרנו היא הסתרת הרבע הימני התחתון של המסך, כך נראות התוצאות-



גם כאן אפשר לשים לב שהתמונות שיצאו לאחר הוספת הפתרון משאלה 4 שוחזרו בצורה טובה יותר.
לדוגמא, בספרה 4, אפשר לראות שבתמונה הימנית יש השלמה של החלק החסר כשבתמונה מהרשת של שאלה 1 יש פס שחור מרוח.

הלוס שאיתו פתרנו את המשימה הוא

$$\max_{I} P(I \mid I_C) = \max_{I} P(I_C \mid I) \cdot P(I) = \max_{I}[\log P(I_C \mid I) + \log P(I)]$$
$$= \max_{I}\left[\log e^{-\|(I-I_C)\odot M\|} \quad + \log e^{-\frac{\|AE(I)-I\|}{T}}\right]$$
$$= \max_{I}\left[-\|(I-I_C)\odot M\| \quad + -\frac{\|AE(I)-I\|}{T}\right]$$
$$= -\min_{I}\left[\|(I-I_C)\odot M\| \quad + \frac{\|AE(I)-I\|}{T}\right]$$

כלומר $\mathrm{L}(I, I_C) = \|(I-I_C)\odot M\| \quad + \frac{\|AE(I)-I\|}{T}$.

חלק תיאורטי:

1. למה Gram טוב ל- textures?
מטריצת Gram שימושית לחישוב המרחק בין ההתפלגויות ה- features הפנימיים המחושבים ל- textures שונים*. מניסויים אמפיריים אנחנו יודעים שרשתות קונבולוציה לומדות ב- features שלהם רמות אבסטרקציה שונות של התמונה. לכן, נרצה שעבור textures דומים ה- features שהרשת לומדת עליהם יתנהגו באופן דומה וזה בדיוק מה שמדד Gram מאפשר לנו.
מה היתרון של Gram על הרבה שכבות קונבולוציה? לעומת שכבות קונבולוציה, מטריצות Gram מאפשרת לבדוק תכונות של ה- textures בשכבות שונות שמגיעות לרמות אבסטרקציה שונות של ה- texture.
אם נשתמש ב- Gram רק בשכבות הראשונות נקבל רק את ה-low level features. למשל, שבשניהם יהיו edges במקומות דומים ואת אותם צבעים אבל לא בהכרח אותם אובייקטים. אם נשתמש ב- Gram רק בשכבות האחרונות נקבל את הדבר ההפוך - רק את ה-high level features. למשל שבשניהם יהיו אותם אובייקטים אבל לא בהכרח באותם צבעים ומיקומים.

*לפי המאמר "Demystifying Neural Style Transfer" משנת 2017:

"As shown in Eq. 9, matching Gram matrices in neural style transfer can been seen as a MMD process with second order polynomial kernel."
כלומר, Gram מתנהג כמו MMD, מדד לחישוב מרחק בין התפלגויות.

2. ב- AdaIn מציגים את ה- texture כ- $\mu(x), \sigma(x)$ כש- x הוא $features$ ברשת והתוחלת והשונות הוא על ה- $channels$.
זה לא מתעלם מהתלויות בין הצ'אנלים השונים בגלל שמתקיים שעבור צ'אנלים x_i, x_j שהם ממופים ל- $(x_i, x_j) \to \frac{\sigma(y)}{\sigma(x)} * \left[(x_i, x_j) - \mu(x)\right] + \mu(y)$. התלות בינהם נשמרת עד כדי הוספה של קבוע והזזה וזה משהו שהרשת יכולה ללמוד.

3.
   a. <u>איזה מה- non parametric generators לומד את $P(I)$?</u>
   נסתכל על כלל ה- non parametric generators שלמדנו:
      i. <u>AE רגיל</u> לומד למזער את $\|AE(I) - I\|$ ולכן בהינתן $I \in TrainingSet$ $\|AE(I) - I\|$ יהיה קטן אבל הכיוון ההפוך לא בהכרח נכון. לכן, לא ניתן להסיק את $P(I)$ מהמודל אפילו שמקובל להגיד כי $P(I) \sim e^{-\frac{\|AE(I)-I\|}{T}}$
      ii. <u>VAE</u> לומד למזער את $\|AE(I) - I\|$ ולכן בהינתן $I \in TrainingSet$ $\|AE(I) - I\|$ יהיה קטן אבל גם כאן הכיוון ההפוך לא בהכרח נכון ולכן לא ניתן להסיק את $P(I)$ מהמודל אפילו שמקובל להגיד כי $P(I) \sim e^{-\frac{\|AE(I)-I\|}{T}}$

iii. GLOW לומד את $P(x)$ ולכן הוא יודע כי $P(I) = \left|\frac{\nabla M^{-1}(x)}{\nabla x}\right| \cdot P_z\left(M^{-1}(x)\right)$ כש- M נלמדת.

iv. GLO לא לומד את $P(I)$ אפילו שהוא יכול להגיד אם התמונה קרובה בנורמה לאחת התמונות המקוריות על ידי $-\min_{i}\|G(z_i) - I\|$.

v. $IMLE$: במאמר שמציג את המודל נאמר כי:

"Our method relies on the following observation: a model distribution that maximizes the likelihood of the data should assign high density to each of the data examples, and so if samples were drawn from the model, samples would be more likely to lie near data examples than elsewhere."

בהנחה שהטענה בציטוט נכונה נוכל להגיד כי $\mathrm{P}(I) \sim 1 - \min_{I' \in \mathrm{TrainingSet}} \|I - I'\|$.

כלומר, המודל IMLE לא לומד את $\mathrm{P(I)}$ אך ניתן להסיק אותו.

vi. WAE לא לומד את $P(I)$. אך ניתן לחשב את ה- MMD בצורה שלומד המודל שיכול לתת מדד למרחק של I מההתפלגות של תמונה אחרת ולמצוא קירוב ל- $P(I) \sim \max_{I' \in \mathrm{Dataset}} \frac{MMD(I,I')}{P(I')}$

לסיכום – רק המודל GLOW לומד באופן ישיר את $\mathrm{P}(I)$

b. איזה מה- non parametric generators ממשיך להשתפר ?

נעבור על כלל ה- non parametric generators שלמדנו:

i. AE רגיל: עשוי להשתפר עם יותר נוירונים אך יכול להגיע ל- $overfit$ של שינון התמונות הנתונות.

ii. VAE: כמו AE רגיל, יכול להשתפר עם יותר נוירונים אבל עלול להגיע ל- $overfit$ של שינון התמונות עם שונות 0.

iii. GLOW: יישאף למקסם את ה- MLE על ההתפלגות ולכן יישתפר יותר עם יותר נוירונים. כיוון שהוקטורים שמהם מיוצרת התמונה מוגרלים כל פעם הרשת לא יכולה לשנן ולהגיע ל-$overfit$.

iv. GLO: יכול להשתפר עם יותר נוירונים אבל יכול גם להגיע ל- $overfit$ של שינון התמונות וה- z.

v. $IMLE$: יישאף למקסם את הנורמת 2 מבחינת KNN על התמונות ולכן יישתפר יותר עם יותר נוירונים (כיוון שכמו GLOW, הוקטורים מהם מיוצרת התמונה מוגרלים כל פעם ולכן לא יצליח לשנן).

vi. WAE: יישאף למקסם את ה- MMD על ההתפלגות ולכן יישתפר יותר עם יותר נוירונים (גם כאן כיוון שהוקטורים מהם מיוצרת התמונה מוגרלים כל פעם המודל לא יצליח לשנן).

כל המודלים שציינו ישתפרו עם יותר דוגמאות כיוון שבכל שניתן למודל יותר דוגמאות המודל ילמד להכליל בצורה טובה יותר.

c. <u>איזה activation עדיף ל-GLOW, ReLU או ש- LeakyReLU?</u>

ניזכר ש- GLOW צריך פונקציות הפיכות ולכן אסור לו להשתמש ב- ReLU (אלא אם כן בדק קודם שהכל חיובי, נניח על ידי EXP) ולכן מומלץ להשתמש ב- LeakyReLU.

d. <u>מימוש IMLE בעזרת ANN</u>

ניתן לממש בעזרת יצירת ENCODER ו-DECODER כשהמימד שאליו ממפה ה-ENCODER קטן מאוד.

אופציה נוספת היא ללמוד $x = D_2(D_1(E_1(E_2(x))$ כאשר גם מתקיים $D_1(E_1(y)) = y$ ולבצע את החיפוש במימד שממופה על ידי E_1 אחרי שלימדנו מראש את E_1, D_1.

4. <u>האם הרעיון של deep image prior יעבוד על סיגנלים עם FC?</u>

אנחנו לא מצפים שרשת כזו תעבוד. הסיבה ש- deep image prior עובד היא שקובנולוציות משמרות סביבה חלקה תוך כדי למידה (בעקבות הלוקליות שלהם), דבר שמייצר תמונות חלקות ויפות תוך כדי הלמידה. אם נחליף את הקונבולוציות בשכבות FC כל נוירון ינסה להצליח כמה שהוא יכול על הפלט האישי שלו ללא קשר לפלטים האחרים, מה שיוביל לרצף של סיגנלים ששואף לסיגנל המקורי אך ללא קשר בין שני חלקים עוקבים בסיגנל. אם נקשיב לתוצאות הביניים במצב כזה זה יישמע כמו ג'יבריש.