סקירה זו היא חלק מפינה קבועה בה אני סוקר מאמרים חשובים בתחום ה-ML/DL, וכותב גרסה פשוטה וברורה יותר שלהם בעברית. במידה ותרצו לקרוא את המאמרים הנוספים שסיכמתי, אתם מוזמנים לבדוק את העמוד שמרכז אותם תחת השם deepnightlearners.

---

לילה טוב חברים, היום אנחנו שוב בפינתנו deepnightlearners עם סקירה של מאמר בתחום הלמידה העמוקה. היום בחרתי לסקירה את המאמר שנקרא:

Improving GAN Training with Probability Ratio Clipping and Sample Reweighting

---

## פינת הסוקר:

**המלצת קריאה ממייק**: מומלץ אך לא חובה לאלו שרוצים להתעמק בשיטות אימון של GANs.

**בהירות כתיבה**: בינונית פלוס.

**רמת היכרות עם כלים מתמטיים וטכניקות של ML/DL הנדרשים להבנת מאמר:** הבנה טובה ווסרשטיין גאן וכל מה שקשור אליו, הכרה בסיסית בשיטות מעולם הסטטיסטיקה כמו importance sampling, רקע בסיסי בלמידה באמצעות חיזוקים (Reinforcement learning) .

**יישומים פרקטיים אפשריים**: אימון גאן משופר במגוון תרחישים

---

## פרטי מאמר:

**לינק למאמר:** זמין להורדה.

**לינק לקוד:** .זמין כאן.

**פורסם בתאריך:** 30.10.2020, בארקיב.

**הוצג בכנס:** NeurIPS 2020.

---

## תחומי מאמר:

- גאנים.
- שיטות אימון של גאנים.

## כלים מתמטיים, מושגים וסימונים:

- וסרשטיין GAN (WGAN).
- מרחק וסרשטיין (WD).
- פונקצית ליפשיץ.
- שיטות וריאציוניות לבעיות אופטימיזציה בתחום הרשתות הגנרטיביות כמו GAN.
- גישות מתורת למידת החיזוק (RL): אופטימיזציה של פוליסי (Policy Optimization - PO) דרך פתרון של בעיית אופטימיזציה עם פונקצית מטרה חלופית - surrogate.
- שיטות דגימה: Importance Sampling (IM).
- מרחקים בין מידות הסתברות: מרחק KL ומרחק KL הפוך.
- אלגוריתמים של Expectation-Maximization (EM).

---

## תמצית מאמר:

אתם בטח יודעים שלמרות מאמצי מחקר אינטנסיביים בשנים האחרונות, האימון של GAN-ים עלול להוות משימה לא טריוויאלית עקב קושי במציאת איזון בין הגנרטור G לדיסקרימינטור D. המאמר הנסקר מציין שבעיות אלו בולטות במיוחד בתחום גנרוט טקסט עקב האופי הדיסקרטי של משימה זו (נציין שכרגע שיטות SOTA למשימות גנרוט של טקסט אינם מבוססות על GAN-ים). כדי להתגבר על סוגיות אלו, מאמר הנסקר מציע שיטה לשיפור תהליך האימון של GAN שמבוססת על שני רעיונות עיקריים:

- מניעה עדכונים גדולים מדי של הגנרטור G שעלולים לפגוע ביציבות של תהליך האימון ולהוביל לאובדן של איזון בין G לדיסקרימינטור D. איזון זה הינו חיוני להתכנסות של תהליך האימון של GAN ולפתרון איכותי עבור בעיית אופטימיזציה מינימקס ש-GAN מנסה לפתור. נזכיר שתהליך אימון של GAN הינו משחק סכום אפס כאשר G מאומן לגרום ל-D לזהות את הדאטה הסינטטי ש-G מייצר כדאטה אמיתי (מסט האימון) ובתורו D מאומן להבחין בין דגימות ש-G מייצר לאמיתיות.
- משקול של דגימות המגונרטות עי" G בתהליך האימון של D. כאמור D מאומן להבחין בין דגימות אמיתיות (מאומן לתת ציון גבוה) מסט האימון לבין דגימות המגונרטות עי" G (מאומן לתת ציון נמוך). בתהליך עדכון של D הדגימות של G באיכות טובה שמצליחות "לעבוד יותר טוב על D" (בעלי ציון גבוה) מקבלות משקל גבוה ואילו דגימות של G ה "פחות אמיתיות" מבחינת D (בעלי ציון נמוך) מקבלות משקל נמוך נמוך יותר. זה הופך את האימון של D ליעיל יותר כי (לטענת המאמר) הוא לא מתבזבז על עדכונים על דגימות קלות מדי (האינטואיציה כאן אומרת שאם D משקיע מאמץ רב יותר בלהתאמן על דגימות איכותיות יותר, הוא יהיה מספיק חזק בשביל להפגין ביצועים טובים גם על דגימות קלות יותר ב"צורה אוטומטית").

**הערה:** גישה זה מזכירה לי שיטות ממשפחת gradient boosting machines (GBM) מממשקלות דוגמאות בהתאם ל"רמת הקושי" שלהם מבחינת המודל (בגדול עד כמה השערוך של המודל מדויק).


Figure 1: Illustration of the proposed approach for stabilizing GAN training. Results are from the CIFAR-10 experiment in Sec.4.1. **Left:** The conventional and surrogate objectives for generator training, as we interpolate between the initial generator parameters $\theta_{old}$ and the updated generator parameters $\theta_{new}$ which we compute after one iteration of training. The $\theta_{new}$ obtains maximal surrogate objective. The surrogate objective curve starts decreasing after $x = 1$, showing the objective imposes a penalty for having too large of a generator update. In contrast, the conventional objective (for WGAN-GP) keeps increasing with larger generator updates. **Middle and right:** Discriminator and generator losses w/ and w/o sample re-weighting. WGAN-GP with our re-weighting plugged in shows lower variance in both discriminator and generator losses throughout training.

## הסבר של רעיונות בסיסיים:

**וסרשטיין GAN:** נקודת ההתחלה של המאמר זה WGAN, המודיפיקציה של ה-GAN המקורי, המשתמשת במרחק וסרשטיין (WD) כבסיס ל-D. כלומר G מאומן לגנרט דגימות בעלות מרחק וסרשטיין נמוך מהדוגמאות מסט האימון. מרחק וסרשטיין הינו מקרה פרטי של טרנספורט אופטימלי וכבר הסברתי על באחד הפוסטים שלי (Learning to summarize from human feedback).

היתרון הבולט של WGAN על GAN רגיל טמון ביכולת של D "להעביר גרדיאנטים" יותר יציבים ל-G גם במקרים כאשר D מצליח בקלות להבדיל בין הדגימות האמיתיות לדגימות המגונרטות. זה קורה בגלל שלהבדיל ממרחק Jensen-Shannon (JS) שאותו מנסה למזער ה-GAN הרגיל, WD הינו בעל אופי רציף יותר ולא מגיע לרוויה (כמו מרחק JS) גם כאשר התפלגות הדגימות של G רחוקה מאוד מההתפלגות של הדאטה סט (המשוערכת ע"י D).

חישוב של מרחק וסרשטיין לפי הגדרתו הינו משימה מאוד קשה ובדרך כלל פותרים את בעיית האופטימיזציה הדואלית שלה (שוויון רובינשטיין-קנטורוביץ'). הבעיה הדואליות הינה המקסום של הפרש התוחלות בין התפלגויות של דאטה האמיתי לבין הדגימות המגונרטות מעל מרחב של פונקציות k-ליפשיץ רציפה, מוכפלת ב k/1. פונקציה זו ממודלת עי" רשת נוירונים כאשר נעשים טריקים שונים, כמו קיצוץ משקלים או אילוצים על הנגזרת של הפונקציה כדי שהפונקציה הממודלת תהיה k-ליפשיץ רציפה). אז בעיית אופטימיזציה ש- WGAN מנסה לפתור, הינה מקסום של הפרש התוחלות זה מעל מרחב כל פונקציות k-ליפשיץ רציפות f, מבחינת D. הגנרטור G מצידו מנסה למזער אותו הפרש התוחלות המתואר לעיל (בעיית מינימקס). אם נתבונן בפונקציית מטרה של WGAN ניתן לראות כי G מנסה למקסם את התוחלת של פונקצית ליפשיץ f (על מרחב הדגימות שלו). ניתן למצוא דמיון בין בעיית אופטימיזציה זו לבין אופטימיזציה של פוליסי בעולם של RL, כאשר פונקציה k-ליפשיץ רציפה f משחקת תפקיד של גמול (reward) והתפלגות דגימות של G ניתן לראות כפוליסי. דמיון זה, שזוהה בכמה מאמרים של השנים האחרונות, ינוצל בבניה של פונקצית מטרה חדשה ל WGAN שהוצעה במאמר.

אחרי שהבנו מה זה WGAN ואת הקשר שלו לבעיות RL, בואו נתקדם בשינוי של פונקציית מטרה של WGAN המוצע עי" המאמר. פתרונה יוביל למניעה של עדכונים גדולים של G ומשקול דגימות, המבוסס על ה"איכות" שלהן בעדכונים של D. לאור הקשר עם בעיות של אופטימיזציה של פוליסת

ב-RL, השיטה שהמאמר מציע דומה לשיטות של אופטימיזציה של פוליסי כמו PPO ו- TRPO. שיטות אלה מחליפות את פונקצית המטרה הרגילה בפונקציה חלופית שמנסה לשפר את פונקציית הפוליסי F_p. זה נעשה עי״ מקסום התוחלת של פונקצית היתרון המוכפלת ביחס של F_p החדשה ל- F_p הישנה תחת אילוץ שמרחק KL בין F_p החדשה לישנה חסום עי״ קבוע קטן (אילוץ זה מופיע לפעמים האיבר רגולריזציה בפונקצית המטרה). בדרך זו F_p החדשה לומדת לתת הסתברויות גבוהות למצבים שבהם פונקצית היתרון מקבלת ערכים גבוהים כלומר הגמול אחרי עדכון של P_i הינו מקסימלי).

**פונקציית המטרה של המאמר:** המאמר מציע להחליף את פונקציית המטרה הסטנדרטית של WGAN בפונקציה F_imp המכילה הפרש של שני האיברים הבאים:

- איבר 1: התוחלת של פונקציה k-ליפשיץ רציפה f מעל מידת הסתברות עזר q (שתלויה בהתפלגות הדגימות המגונרטות P_g וגם בפונקצית f הממודלת עי" D בצורה מפורשת ולא פרמטרית (!!!)).
- איבר 2: מרחק KL בין q לבין P_g.

המאמר מציע לאמן את WGAN עי" מקסום של F_imp, כאשר הפרמטרים הם משקלי הרשתות של G ו-D. אם נזכר בעובדה שמרחק KL הינו תמיד אי שלילי, ניתן להבין שהמקסום של F_imp שקול למקסום של האיבר הראשון המינימיזציה של האיבר השני. אז ניתן לפרש את בעיית מקסום F_imp באופן הבא:

מקסום של תוחלת הציון הניתן עי" D להתפלגות q (האיבר הראשון) כאשר אנו מנסים לשמור את התפלגות הדגימות של G קרובה ל q.

אימון של G: מקסום של F_imp מבחינת הפרמטרים של G, הינו מקרה קלאסי של בעיית אינפרנס ורציאונית שמזכירה את בעיית אופטימיזציה שאנו פותרים למשל בVAE- Variational AutoEncoder. הדרך הטבעית לפתור אותה הינה להשתמש באלגוריתם EM קלאסי. בשלב E של EM, אנו מוצאים את ההתפלגות g שהיא בצורה של מכפלה של אקספוננט של P_g ושל f (מנורמלת). שימו לב שמה שיש מכפלה זו מהווה משקול של P_g, כאשר הדגימות עם ציון של D יותר גבוה מקבלות הסתברות גבוהה יותר, שזה מה שרצינו מההתחלה.

השלב M של האלגוריתם הינו אופטימיזציה של F_imp על הפרמטרים של G כאשר התפלגות q נתונה (חושבה בשלב E). זה למעשה מינימיזציה של האיבר השני, מרחק KL. וכאן יש לנו בעייה כי q זה בעצם פונקציה של P_g הניתנת בצורה לא מפורשת ובשביל לשערך את מרחק KL נצטרך לדגום מ-q שזה מאוד לא טריוויאלי. למזלנו ניתן להשתמש ב-KL הפוך ולהפוך את האיבר זה לסכום של מינוס התוחלת של f מעל P_g ומרחק KL בין P_g עבור האיטרציה הקודמת לבין P_g שאנו מנסים לאפטם (נוסחה 4 במאמר). בעצם אנו מנסים למקסם את התוחלת של f מעל P_g אך לא רוצים להתרחק מדי מההתפלגות P_g מהאיטרציה הקודמת. אם אתם זוכרים את ההסבר שלי על PPO ועל TRPO, מיד תזהו את הדמיון. אז בדומה לשיטות אלו, המאמר מציע להחליף את פונקציית המטרה כאן בפונקציית מטרה חלופית המכילה המכפלה של פונקציה f ביחס בין P_g

הישן לחדש r_g(!!). בנוסף הם מאלצים את r_g להיות קטן באופן מאולץ (מקצצים). אבל כאן יש לנו עוד בעיה. איך נחשב את היחס הזה על דגימה של G אם P_g נתון בצורה לא מפורשת. כאן הם עושים טריק נחמד. בנוסף ל D של WGAN, הם מאמנים דיסקרימינטור בינארי D_bin בשביל להבדיל בין הדגימות של G לדגימות האמיתיות. ניתן להוכיח (עשו זאת במאמר המקורי של GAN למשל) שעבור D_bin אופטימלי ניתן לחשב את ערך של P_g עבור הדגימה של הערך של D_bin הדגימה זו. בדרך זו ניתן לשערך את r_g עבור דגימה נתונה.

**אימון של D:** כאן אנו צריכים לאפטם רק את האיבר הראשון (התוחלת של f מעל התפלגות q נתון כאשר מאפטמים את הפרמטרים של f). כאן משתמשים כמובן ב Gradient Descent אבל נשאלת השאלה איך נחשב את הגרדיאנט עבור הפרמטרים של f אם אנחנו לא יודעים לדגום מ-q. בשביל להתגבר על הקושי הזה הם משתמשים בטכניקה קלאסית בסטטיסטיקה הנקראת IM תוך ניצול של הצורה של q (מכפלה של אקספוננט של P_g ושל f). בתור התפלגות proposal שדוגמים ממנו במקום q, הם לקחו את P_g שקל לדגום ממנה. נציין שהתוחלת של הגרדיאנט מעל q של f יוצאת שווה לתוחלת מעל P_g של המכפלה של f באקספוננט של f. כך אנו משיגים את המשקול הגבוה לדגימות בעלות ציון גבוה מ D משפיעות יותר חזק על העדכון של D כאשר השפעה של דגימות עם ציון נמוך על עדכון של D קטנה (!!).

**Algorithm 1** GAN Training with Probability Ratio Clipping and Sampling Re-weighting

Initialize the generator $p_\theta$, the discriminator $f_\phi$, and the auxiliary binary classifier $C$
**for** $t \leftarrow 1$ to $T$ **do**
  **for** certain number of steps **do**
    Update the discriminator $f_\phi$ with sample re-weighting through Eqs.(7)-(8), and maintain $f_\phi$ to have upper-bounded Lipschitz constant through, e.g., gradient penalty [15].
  **end for**
  **for** certain number of steps **do**
    Finetune the real/fake binary classifier $C$ (for 1 step)
    Estimate probability ratio $r_1(\theta)$ using $C$ through Eq.(6)
    Update the generator $p_\theta$ with probability ratio clipping through Eq.(5)
  **end for**
**end for**

## הישגי מאמר:

**דומיין של תמונות:** המאמר מראה שהשיטה שלהם משפרת את איכות התמונות מבחינת Inception Score ו- Frechet Distance מול כמה GAN-ים וביניהם אלו המבוססים על הלוס של WGAN עם טכניקות ייצוב אימון שונות וגם על כמה GAN-ים עם פונקציות לוס אחרת (לא בסגנון וסרשטיין). הם גם מראים שהם אכן מצליחים לייצב את האימון ועבור WGAN קלאסי (השונות של גרדיאנטים נמוכה יותר וההתכנסות יותר מהירה). הניסויים נעשו בעיקר על CIFAR10.

**דומיין טקסטואלי:** הם הצליחו לשפר את איכות הטקסט המגונרט - ההשוואה נעשתה עי״ BLEU. מעניין שהם גם הצליחו לשפר את איכות ביצוע המשימה של ״העברת סגנון״ (Style Transfer) כאשר המטרה כאן לשנות את סגנון המשפט (למשל סנטימנט) תוך כדי שימור התוכן.

| Length | MLE | SeqGAN [56] | LeakGAN [16] | RelGAN [35] | WGAN-GP [15] | Ours | Real |
|---|---|---|---|---|---|---|---|
| 20 | 9.038 | 8.736 | 7.038 | 6.680 | 6.89 | **5.67** | 5.750 |
| 40 | 10.411 | 10.310 | 7.191 | 6.765 | 6.78 | **6.14** | 4.071 |

Table 2: Oracle negative log-likelihood scores (↓) on synthetic data.

| Method | BLEU-2 (↑) | BLEU-3 (↑) | BLEU-4 (↑) | BLEU-5 (↑) | $NLL_{gen}$ (↓) | Human (↑) |
|---|---|---|---|---|---|---|
| MLE | 0.768 | 0.473 | 0.240 | 0.126 | 2.382 | - |
| LeakGAN [16] | 0.826 | 0.645 | 0.437 | 0.272 | 2.356 | - |
| RelGAN 100 [35] | 0.881 | **0.705** | **0.501** | 0.319 | 2.482 | - |
| RelGAN 1000 [35] | 0.837 | 0.654 | 0.435 | 0.265 | 2.285 | 3.42±1.23 |
| WGAN-GP [15] | 0.872 | 0.636 | 0.379 | 0.220 | **2.209** | - |
| Ours | **0.905** | 0.692 | 0.470 | **0.322** | 2.265 | **3.59 ± 1.12** |

Table 3: Results on EMNLP2017 WMT News. BLEU measures text quality and $NLL_{gen}$ evaluates sample diversity. Results of previous text GAN models are from [35], where RelGAN (100) and RelGAN (1000) use different hyper-parameter for gumbel-softmax. Our approach uses the same gumbel-softmax hyper-parameter as RelGAN (1000).

| Method | IS (↑) | FID (↓) |
|---|---|---|
| Real data | 11.24±.12 | 7.8 |
| WGAN-GP (2017) | 7.86±.08 | - |
| CT-GAN (2018) | 8.12±.12 | - |
| SN-GANs (2018) | 8.22±.05 | 21.7±.21 |
| WGAN-ALP (2020) | 8.34±.06 | 12.96±.35 |
| SRNGAN (2020) | 8.53 ±.04 | 19.83 |
| Ours (re-weighting only) | 8.45±.14 | 13.21±.60 |
| Ours (full) | **8.69±.13** | **10.70±.10** |

Table 1: CIFAR-10 results. Our method is run 3 times for average and standard deviation.

Figure 2: Generated samples by WGAN-GP (top-left), CT-GAN (bottom-left), and ours (right).

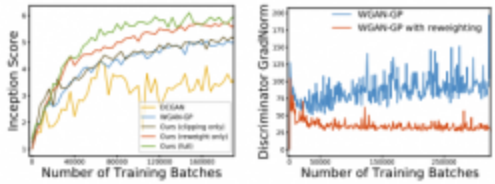


Figure 3: **Left:** Inception score on CIFAR-10 v.s. training batches (including both generator and discriminator batches). The DCGAN [39] architecture is used. **Right:** The gradient norms of discriminators on fake samples.

## נ.ב.

אחד המאמרים היפים מבחינת האלגנטיות המתמטית המתבטא השילוב טכניקות מתחומים שונים (לא ציינתי בסקירה שהם מוכיחים שהגישה שלהם מקדמת את ההתפלגות של G לכיוון של התפלגות הדאה האמיתית). לגבי הישימות של גישה זו חייבים לבחון אותה על דאטה סטים יותר מגוונים ועל משימות מורכבות יותר.

#deepnightlearners

הפוסט נכתב על ידי מיכאל (מייק) ארליכסון, PhD, Michael Erlihson.

מיכאל עובד בחברת סייבר Salt Security בתור Principal Data Scientist. מיכאל חוקר ופועל בתחום הלמידה העמוקה, ולצד זאת מרצה ומנגיש את החומרים המדעיים לקהל הרחב.