**למידה חישובית ממידע: הרצאה 7 – SVM**

**המשך Linear Classifiers**

* A screenshot of a cell phone

  Description automatically generated ניתנה כאן תזכורת על **גרדיאנט דיסנט** בצורה של באץ׳ וסטוכסטית (כל מה שנאמר מופיע בהרצאות על LoR ורגרסיה לינארית).
* נשים לב כי בשקף זה בכל מקום בו מופיעה טטהi מדובר בטטהj.
* התזכורת נאמרה כאן מפני האלגוריתם הפרספטרון שלמדנו בשיעור שעבר מתנהג כמו גרדיאנט דיסנט סטוכסטי, אין גרסה אחרת לפרספטרון.
* A screenshot of a cell phone

  Description automatically generatedסטוכסטי = מקרי, רנדומי. מבחינת שינוי סדר הדגימות שיכול להיות רנדומי.
* ניתנה תזכורת על כך שהאלגוריתם LMS להפרדה לינארית יכול להיות בעייתי מפני שהוא יכול להביא לנו מודל שהוא אינו מפריד את הדאטה כמו שצריך, מפני שהוא ממזער מרחקים ולא טעויות.
* A close up of a logo

  Description automatically generatedכדי לספור את הטעויות של המפריד הליניארי ולמזער אותן אנחנו משתמשים באלגוריתם הפרספטרון (למדנו בשיעור קודם), אשר ממזער את מספר הטעויות. **להלן אלגוריתם הפרספטרון**. **הפרספטרון הוא סטוכסטי ביסודו** – הוא עובר על הדגימות ומעדכן את המשקולות אם יש צורך בכך.
* **משפט רוזנבלט** אומר שאם קיים מפריד לינארי עבור הדאטה, הפרספטרון ימצא אותו.  
  (אין התייחסות למתי תהיה התכנסות, אבל מובטח שתהיה)
* **אבל, דאטה איננה תמיד תהיה ניתנת להפרדה לינארית... (להלן דוגמה)**

A screenshot of a cell phone

Description automatically generated

**הפרספטרון ומפרידים לינאריים – מיפוי לממד גבוה יותר**

לפנינו דאטה בממד 1 שאינו ניתן להפרדה לינארית. נמפה את הדאטה הנ״ל לממד 2 כך שהנקודה -2 בממד 1 הפכה לנקודה (-2,4) בממד 2 על פי התיאור x:(x,x^2).  
**בממד הגבוה יותר יש מפריד לינארי!** לכן לאחר מיפוי לממד 2 ולפי משפט רוזנבלט, **הפעלה של פרספטרון על הדאטה במימד החדש אכן תניב מפריד לינארי.**

A close up of a map

Description automatically generatedA picture containing table

Description automatically generated

**דוגמאות נוספות:**

נגדיר את ההפרדה בממד הגבוה יותר על ידי וקטור המשקולות,  
כך שלמשל , בדוגמות אלו וקטור המשקולות w עבור הממד הגבוה  
יותר – ממד 3, יכיל 4 ערכים.

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

**נגדיר phi-separability:**

חלוקה (בינארית) , או ***דיכוטומיה*** (C0,C1), של training set, C תקרא 

A drawing of a person

Description automatically generatedניתן לחלוקה ב-phi, אם קיים וקטור w ממימד N כך ש:

A close up of a map

Description automatically generatedהרעיון מאחורי Non-linear Mapping: נמפה דאטה ממרחב אינפוט בעל ממד נמוך למרחב אימביאנטי (ambient space) גבוה ונקווה שהדאטה תהיה ניתנת להפרדה לינארית בממד החדש**. דוגמה למיפוי מממד 2 לממד 4:** 

ופונקציית הדיסקרימיננטה תהיה: A picture containing knife

Description automatically generated

הרעיון הוא שנוכל להשתמש במרחבים יותר ויותר גבוהים מבחינת ממד עד אשר נגיע להפרדה הלינארית הרצויה – שזו המטרה שלנו.

**Cover’s Function Counting Theorem: משפט קובר סופר את הדיכוטומיות שניתנות להפרדה לינארית**

**Counting Dichotomies**

* דיכוטומיה של קבוצה S היא חלוקה של S לשתי תתי קבוצות-זרות של S.
* נניח כי יש לנו k דגימות בקבוצת דגימות S.
* אזי יש לנו 2^(k-1) אפשרויות לדיכוטומיות שונות מעל הדגימות הללו
* A picture containing object, clock

  Description automatically generatedכל דיכוטומיה מגדירה משימת סיווג/קלסיפיקציה (המפרידה בין שני קלאסים)

משפט קובר Cover’s Counting Theorem: במרחב N-ממדי מספר הדיכוטומיות של k דגימות הניתנות להפרדה לינארית הוא:

A close up of a map

Description automatically generatedלכן, ההסתברות שדיכוטומיה הינה ניתנת להפרדה לינארית היא: A close up of a clock

Description automatically generated

**ככל ש-N גדל, מספר הדיכוטומיות גדל וכן גם ההסתברות גדלה (בהנחה כי k הוא מספר קבוע, שבדרך כלל גבוה הרבה יותר מ-N).**

\*\*אין משמעות להקפה הכתומה, נשאר מהשקף הקודם שזוהר הראה.

**Full Rational Varieties**

**נגדיר:** full rational variety מסדר r במרחב אינפוט מממד n מתואר על ידי כל המונומיאלים (monomials) מדרגה n של משתני האינפוט ב-x – A picture containing object, clock

Description automatically generated כאשר - 

מספר גורמי המונומיאל השונים בביטוי זה הוא: A picture containing object, clock

Description automatically generated(נכון קומבינטורית, יש לנו r אבנים ויש לנו n מקלות לשים ביניהם)

A screenshot of a cell phone

Description automatically generatedאם n=10 ו-r=5 נקבל 3000 ממדים (ממד 10 ועלינו 5). פרקטית אנחנו עולים לממדים מאוד גבוהים. נרצה להיות יותר חסכוניים מפני שבמצב זה הפרספטרון יצטרך לבצע 3000 מכפלות פנימיות! מה שמאפשר לנו לחסוך, הם קרנלים.

**קרנלים – Kernels**

פונקציה k ממרחב הזוגות הסדורים (R^n x R^n) ל-R נקראת קרנל (גרעין) אם קיים מיפוי (מסומן באות היוונית פי) מ-R^n ל-R^N  כך שמתקיים: . קרנל הוא הכללה של מכפלה פנימית.

* קרנלים עוזרים לנו להימנע מחיפוש אקספליסיטי אחר מרחב אמביאנטי (ambient space) ואחר פונקציות מיפוי, פי, לממד גבוה יותר.
* הקרנלים מעבירים את הלמידה לפעולות ישירות בממד האינפוט (שהוא מממד נמוך יותר).
* למעשה, קרנלים יכולים גם לתמוך בלמידה של מיפוי מרחבים בממד אינסופי (general Hilbert spaces).

**A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedדוגמאות לקרנל:**

A screenshot of a cell phone

Description automatically generated

**בחזרה לפרספטרון – צעד עדכון המשקלים**

1. רק לדגימות שהן מסווגות באופן שגוי יש השפעה
2. אם **האאוטפוט שלילי והטרגרט (הלייבל) חיובי**, אנחנו מעלים את המשקולות על ידי הוספת חלק/שבר **חיובי** של הדגימה
3. אם **האאוטפוט חיובי והטרגט (הלייבל) שלילי**, אנחנו מורידים את המשקולות על ידי הוספת חלק/שבר **שלילי** של הדגימה
4. **לכן, בעדכון של w, אנחנו תמיד מוסיפים חלק/שבר חיובי של Td\*Xd (מכפלה של הטרגט/הלייבל (סקלאר) עם הוקטור x, אם יש מיס-קלסיפיקציה)**

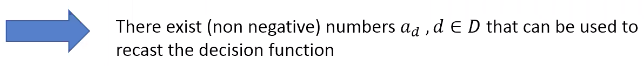
A picture containing bird

Description automatically generated

ניגע באלגוריתם המבצע המסומן ע״י פונקצייה f(x) מסווגת

לכן למעשה יש איזשהו מקדם שמסומן ad שהוא זה שכופל את Td\*Xd לכל d דגימת דאטה, לכל נקודה.  
ומכאן נובע השוויון השני שלאחר המכפלה הפנימית שבין וקטור המשקולות לוקטור x. השוויון השלישי נובע מלינאריות של מכפלה פנימית.

הנקודה שאנחנו באים לסווג הוא הווקטור x באלגוריתם המבצע. אנחנו מכפילים אותו עם כל דגימות הדאטה על פי השוויון האחרון – instance based learning.



* מהיות ש- w הוא קומבינציה של חלקים/שברים של הדגימות כפי שהסברנו לעיל, כתבנו את פונקציית ההחלטה כך: A picture containing object, clock

  Description automatically generated.
* **נשים לב כי במטרה להשתמש בפונרציית ההחלטה הזו  אנחנו צריכים רק לאחסן את כל נקודות הדאטה Xd שעבורן ad אינו 0.**
* **נקודות דאטה אלו נקראות support vectors.**

**A screenshot of a social media post

Description automatically generated**

**הפרספטון הדואלי / The Dual Perceptron**

הפרספטרון הדואלי (צד ימין) **לומד את המקדמים ad של הדגימות הקיימות בדאטה.** תמיד נוסיף חתיכה חיובית של Td\*Xd.

A screenshot of a cell phone

Description automatically generated

מעבר לממדים גבוהים יותר

באופן שקול לאלגוריתם הפרספטרון הדואלי, אנחנו יכולים לכתוב את **המעבר לממד גבוה יותר** על ידי שימוש בפונקציה הממפה (פי) על xi והדגימה xd. ההבדל היחיד הוא השימוש בפי.

**מכאן, מכיוון שקיים מיפוי כזה נוכל להשתמש בקרנל מהגדרה.**

כעת נמיר את הפרספטרון הדואלי לממדים גבוהים יותר על ידי **שימוש בקרנל** במקום במכפלה פנימית של פי עם xi ופי עם xd.

**פרספטרון קרנל / The Kernel Perceptron**

A picture containing food

Description automatically generated

A screenshot of a cell phone

Description automatically generated**SVM = Support Vector Machine  
האלגוריתם המבצע**

* תהי תת קבוצה של דגימות ה-training שנקראת support vectors
* תהי קבוצת של משקולות עבור הדגימות הללו
* תהי פונקציית קרנל שהיא (כתוצאה, באפקט שלה) לא-לינארית ממפה למרחב בממד גבוה יותר.
* נסווג +/- לפי כלל ״מבוסס דגימות״ פשוט: A close up of a clock

  Description automatically generated

מה אנחנו עושים בכלל המסווג? סכום על כל הדגימות שהן support vectors – נפעיל את הקרנל על ה-support vector ועל x הנקודה החדשה, נכפיל בלייבל ונמשקל במקדם ad, אותו עלינו ללמוד.

A screenshot of a video game

Description automatically generatedכאשר הקרנל הוא היפר-פרמטר ולכן עלינו לבצע עליו ולידציה, לבחור את הקרנל הטוב מבין הקרנלים.

**נרצה לעשות אופטימיזציה לשוליים Optimizing the margin**

**נגדיר:** השוליים (margin) של מרחב מפריד (ליניארי למשל) f(x)=0 מוגדרת כמרחק המינימלי בין דגימה למרחב מעל כל דגימות ה-training: A drawing of a face

Description automatically generated

**המטרה שלנו** במסווג מסוג SVM ולמידת SVM **היא למקסם את המרחק המינימלי**, מה שיהפוך את האלגוריתם המבצע ליותר יציב, עמיד יותר בפני ״רעשים״, פחות רגיש ובעל הכללה טובה יותר. בדוגמה נרצה למצוא את הקו הכחול.

A close up of a map

Description automatically generated

אנחנו מחפשים את המישור המפריד עם השוליים המקסימליים: את המישור המפריד מייצג w, ונרצה להביא למינימום את הפונקצייה שמופיעה משמאל, שהיא חצי מהנורמה של הווקטור w (אשר מייצג את המישור הפריד), בריבוע, **תחת אילוצים מסויימים**:  
לכל נקודה בדאטה שלי, d, עלינו שהלייבל שלה td כפול (המכפלה הפנימית של נקודה זו xd עם w ועוד w0) פחות אחד = אי שלילי. כלומר נרצה את ה-w הקטן ביותר שעדיין מקיים את זה. המשמעות = כל הנקודות הן בקלאס הנכון. וה-w הקטן ביותר מגדיר את ה-margin הגדול ביותר.

כדי לעשות זאת לא נשתמש בגרדיאנט דיסנט אלא בכופלי לגרנז׳.

A close up of a logo

Description automatically generatedהרעיון של כופלי לגרנז׳:

הצבעים מייצגים את המרחק מהראשית, הנורמה = המרחק האקלידי מהראשית. כאשר g מייצג ישר, הישר האדום. ואנחנו נחפש את הנקודה הצהובה. (נרחיב על כך בהמשך)

A close up of a map

Description automatically generated

**כיצד נמצא שוליים גדולים יותר אם נאפשר מעט שגיאות?**

**נשתמש במשתנים ״גמישים״ = משתני Slack**

* הרעיון: אנחנו מוכנים ״לסבול״ מספר מסוים (לא יותר מידי!) של נקודות training שיסווגו באופן שגוי.
* A close up of text on a white background

  Description automatically generatedמשתני “Slack” (הסימון הנחשי, קסי) יכולים לעזור כאן:

נוכל לאפשר שגיאה מסוימת על הנקודות, כך שלא כולן יסכמו יחד ל-1, אלא ל-1 פחות משתנה סלאק כלשהו לכל נקודה d שהוא אי שלילי, וכך שסכומי כל ה-slacks של כל הנקודות יהיה קטן מאיזשהו קבוע (המגבלה על הסלאק).

כך שלבסוף, נקבל מסווג שעושה טעויות על חלק מהנקודות מה שעוזר למנוע overfitting של המודל.

A close up of text on a white background

Description automatically generated

**סיכום עד כה,**  
האינפוט הוא קבוצת דגימות והלייבלים שלהן.

היפר-פרמטרים: הקרנל, שמאפשר לנו לעבור לממדים גבוהים ושולט בממד אליו אנחנו ממפים (שהסיכוי למצוא בו מפריד גבוה יותר.) וכן, הקבוע ששולט במשתני ה-slack.

האאוטפוט הוא תת קבוצה של דגימות training שהם ה-  
support vectors וכן קבוצה של משקולות עבור דגימות אלו.  
בעזרתם ניצור את המסווג.

A close up of text on a black background

Description automatically generated