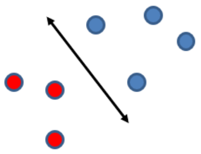
למידת מכונה הרצאה 5

ראינו ב *winnow* שבוע שעבר, שהאלגוריתם נותן לנו חוק לינארי – בעצם מישור (נרחיב על זה עוד מעט). נזכיר מה האלגוריתם עושה: מתחיל עם משקל זהה לכל פיצ'ר, בכל איטרציה מחשב את הסכום $\sum_{j=1}^{n} w_j F_j(X_i)$ ובודק אם הוא גדול מ n, ואם יש טעות מעדכן את המשקלים עד שהחוק עקבי. הרעיון הזה שמכפילים את מה שטוב ומורידים את מה שרע לאפס, זה רעיון שחוזר על עצמו בלמידת מכונה.

האלגוריתם הזה עובד על מודל מאד מסויים שמניחים שרק חלק מהפיצ'רים משמעותיים, ושאם אחד מהם קיים, אז התיוג של הנקודה הזו הוא $+$ בהכרח. זה מודל מאוד מוגבל. נרצה לראות אלגוריתם שעובד על יותר מקרים.

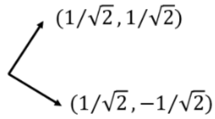
למדנו שלמישור שיש לו מרווח לפחות γ מכל הנקודות, יש מימד-VC $O(R^2/\gamma^2)$ (כאשר הנקודות חסומות במעגל ברדיוס R). בדרך כלל נגדיר את $R = 1$. איך נמצא מישור מפריד טוב? (בהנחה שבכלל אפשר להפריד ע"י מישור).



ננסה ע"י *brute force*: בדוגמה של שני מימדים, הקו שמפריד עם המרווח הכי גדול נקבע ע"י 3 נקודות לכל היותר. אז ננסה את כל האופציות לבחור 3 נקודות מתוך n. לכל קו כזה, נבדוק אם יש נקודה שנכנסת לתוך המרווח. הסיבוכיות היא $O(n^4)$. ועבור מימד d, $O(n^{d+2})$. זה זמן ריצה לא אפשרי. (לדוגמה, ל-MNIST יש $n = 60{,}000, d = 784$).

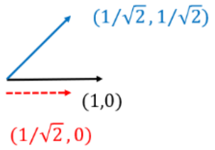
לפני האלגוריתם, נרחיב על מרחב מכפלה פנימית ומה הכוונה בכך שהחוק הוא מישור:

ניזכר מה המשמעות של מכפלה פנימית: ההגדרה הגיאומטרית היא $|a| \cdot |b| \cdot \cos(\theta)$. אינטואיטיבית אנחנו רואים, שאורכי הווקטורים מגדילים את הערך, והזווית מגדירה האם הוא יהיה חיובי או שלילי. אם הווקטורים מאונכים – המכפלה הפנימית היא 0. עבור ווקטורים מנורמלים, אם הם זהים אז המכפלה הפנימית תהיה 1.



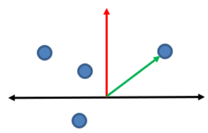
לדוגמה, נחשב: $\frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} \cdot \frac{-1}{\sqrt{2}} = 0$.

המכפלה הפנימית גם מגדירה את **ההטלה** של וקטור על וקטור אחר. כלומר, כמה הוקטור הכחול הולך בכיוון של הווקטור השחור ←

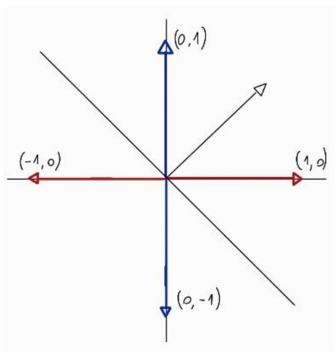


כלומר באלגוריתם *winnow*, לוקחים את הווקטור שמייצג נקודה עם n פיצ'רים. ואםם $\sum_{j=1}^{n} w_j F_j(X_i) \geq n$, אז מנחשים $+$, ואחרת, $-$. כזכור, $F_j(X_i)$ הוא הערך של הפיצ'ר ה-j של הנקודה X_i. החוק הזה הוא בעצם חישוב מכפלה פנימית של הווקטור X_i עם $W = (w_1, \dots, w_n)$.

נדמיין את המישור שמאונך לווקטור W. אם לווקטור X_i, המכפלה הפנימית שלהם חיובית, זה אומר ששניהם באותו כיוון – כלומר מאותו צד של המישור. ואם היא שלילית, הם בצדדים הפוכים. בעצם, מצאנו מישור שמפריד בין הדאטא שלנו.



בדוגמה הזו – 3 הנקודות העליונות, במכפלה פנימית עם הווקטור האדום יתנו מספר חיובי. אינטואיטיבית, אפשר לראות שההטלה שלהם על הווקטור האדום היא באותו כיוון של הווקטור.



באופן כללי, אם ניקח k ווקטורים של הבסיס האורתונורמלי, ואת הנגדיים שלהם:

$$(1,0,\dots,0), (-1,0,\dots,0), (0,1,0,\dots,0), (0,-1,0,\dots,0), \dots, (0,\dots,0,1), (0,\dots,0,-1)$$

אז המישור המפריד עם המרווח הגדול ביותר הוא $\left(\frac{1}{\sqrt{k}}, \frac{1}{\sqrt{k}}, \dots, \frac{1}{\sqrt{k}}\right)$. המרווח הוא $\frac{1}{\sqrt{k}}$.

כל וקטור מהצורה $(\alpha, \alpha, \dots, \alpha)$ ישמר את התכונה של הכיוונים. העובדה שהווקטור מנורמל נותן את התכונה שהמכפלה הפנימית היא המרחק מהמישור.

אלגוריתם *perceptron* (1958):

אנחנו בעצם נחפש את הווקטור שמאונך למישור המפריד. נתחיל עם ווקטור האפס, ובכל איטרציה נתקרב אל הווקטור הרצוי w^*.

1) נגדיר את $w_1 = \vec{0}$. (w_1 הוא הניחוש באיטרציה ה-1)

2) עבור $t = 1,2, \ldots$
a) נעבור על כל הנקודות x_i
i) אם $w_t \cdot x_i > 0$, ננחש +
ii) אחרת, ננחש −
iii) אם טעינו:
(1) אם x_i באמת +: $w_{t+1} \leftarrow w_t + x_i$
(2) אם x_i באמת −: $w_{t+1} \leftarrow w_t - x_i$
(3) נקדם את t
iv) אם עשינו סבב בלי טעויות, נסיים.

טענה: *perceptron* מוצא מישור שאחיד על S, ב- $1/\gamma^2$ איטרציות. אז זמן הריצה הוא n/γ^2.
(נניח שכל הנקודות חסומות בתוך כדור היחידה).

ראינו מקודם שהמרווח קובע את איכות הלמידה (שאם יש מרווח גדול, הדיוק גבוה, גם בלי תלות במימד-VC). עכשיו אנחנו רואים שגם זמן הריצה תלוי במרווח, וזו אותה נוסחה. זה לא במקרה – העובדה שאפשר לחשב את המישור בזמן קצר, קשורה לזה שהחוק פשוט.

יהי w^* הווקטור המנורמל המאונך למישור שמפריד את הנקודות של S עם מרווח γ. (ניזכר ש-w_t לא מנורמל). נוכיח טענות עזר:

טענת עזר 1: $w_{t+1}w^* \geq w^t w^* + \gamma$. כלומר, הווקטור מתקרב ל-w^*.

הוכחה: אם ה-x שטעינו בו הוא +, אז לפי הצעדים באלגוריתם:

$$w_{t+1}w^* =^{\text{א}} (w_t + x)w^* = w_t w^* + x w^* \geq^{\text{ב}} w_t w^* + \gamma$$

א – לפי צעד (1) באלגוריתם.
ב – כי xw^* זה המכפלה הפנימית של x עם הווקטור האופטימלי, כלומר המרחק שלו מהמישור. אם זה הווקטור האופטימלי, המרחק של כל הנקודות מהמישור הוא לפחות γ.

אם ה-x שטעינו בו הוא −, אז:

$$w_{t+1}w^* = (w_t - x)w^* = w_t w^* - x w^* \geq^{\text{א}} w_t w^* + \gamma$$

א – כי xw^* יהיה שלילי. כלומר לפחות $(-\gamma)$ בכיוון השני.

טענת עזר 2: $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 1$. בכל איטרציה, הנורמה של הווקטור גדלה ב-1 לכל היותר.
כלומר, למרות שהווקטור שאנחנו מוצאים לא מנורמל, הגודל שלו עדיין חסום.

הוכחה: אם ה-x שטעינו בו הוא +, אז לפי הצעדים באלגוריתם:

$$\|w_{t+1}\|^2 =^{\text{א}} \|w_t + x\|^2 =^{\text{ב}} \|w_t\|^2 + 2w_t x + \|x\|^2 \leq^{\text{ג}} \|w_t\|^2 + 2w_t x + 1 \leq^{\text{ד}} \|w_t\|^2 + 1$$

א – לפי צעד (1) באלגוריתם.
ב – נוסחת כפל מקוצר.
ג – כי כל הווקטורים הם בתוך כדור היחידה.
ד – ה-x הוא +, ואנחנו טעינו בניחוש, כלומר ניחשנו −. זה קורה כאשר $w_t x_i \leq 0$. כלומר $2w_t x \leq 0$.

אם ה-x שטעינו בו הוא −, אז:

$$\|w_{t+1}\|^2 =^{\text{א}} \|w_t - x\|^2 =^{\text{ב}} \|w_t\|^2 - 2w_t x + \|x\|^2 \leq^{\text{ג}} \|w_t\|^2 - 2w_t x + 1 \leq^{\text{ד}} \|w_t\|^2 + 1$$

א – לפי צעד (1) באלגוריתם.
ב – נוסחת כפל מקוצר.
ג – כי כל הווקטורים הם בתוך כדור היחידה.
ד – ה-x הוא −, ואנחנו טעינו בניחוש, כלומר ניחשנו +. זה קורה כאשר $w_t x_i \geq 0$. כלומר $2w_t x \geq 0$.

מטענה 1, בכל איטרציה המכפלה הפנימית גדלה ב-γ לפחות. כלומר באיטרציה ה-M, $w_{M+1}w^* \geq M\gamma$.

מטענה 2, (בגלל שהווקטור מתחיל עם נורמה 0), איטרציה ה-M: $\|w_{M+1}\|^2 \leq M \Longrightarrow \|w_{M+1}\| \leq \sqrt{M}$.

בסה"כ, נקבל:

$$\left(\frac{w_{M+1}}{\|w_{M+1}\|}\right) w^* \leq^{א} 1 \Longrightarrow w_{M+1}w^* \leq \|w_{M+1}\| \Longrightarrow M\gamma \leq \sqrt{M} \Longrightarrow M \leq \frac{1}{\gamma^2}$$

א – כי המכפלה הפנימית של כל שני וקטורים מנורמלים, חסומה ב-1

חסמנו את מספר האיטרציות, כנדרש.

נזכיר: אנחנו רוצים למצוא מישור מפריד עם מרווח, כי זה חוסם את המימד-VC. והוכחנו ש-$perceptron$ מוצא מישור שמפריד. לא הוכחנו שיש מרווח. במימד נמוך זה מספיק לנו, אבל במימד גבוה, חשוב שיהיה מרווח כדי שהחסם יהיה קטן.

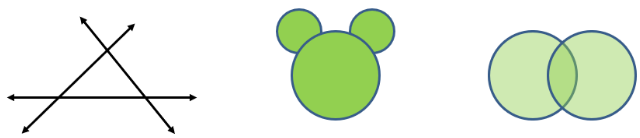
אפשר לשנות את האלגוריתם כדי שימצא מרווח קרוב ל-γ, ובזמן ריצה באותו סדר גודל. נדגים עבור $\gamma/2$:

1) נגדיר את $w_1 = \vec{0}$. (w_1 הוא הניחוש באיטרציה ה-1)
2) עבור $t = 1,2,\ldots$
   a. נעבור על כל הנקודות x_i
      i. אם $w_t \cdot x_i < \gamma/2$ אבל x_i הוא $+$:
         1. $w_{t+1} \leftarrow w_t + x_i$.
         2. נצא מסבב t.
      ii. אם $w_t \cdot x_i > -\gamma/2$ אבל x_i הוא $-$:
         1. $w_{t+1} \leftarrow w_t - x_i$
         2. נא מסבב t.
   b. אם עשינו סבב בלי טעויות, נסיים.

זמן הריצה זהה, עד כדי קבוע. וההוכחה זהה, עד כדי קבוע.

כל זה מדבר על מצב שה-γ לא ידוע לנו. אפשר "לנחש" אותו: אנחנו יודעים מה זמן הריצה אמור להיות (כי הוא חסום רק כאשר יש פתרון עבור γ), אז ננסה מספרים גדולים יותר ויותר. כשנחרוג מזמן הריצה המצופה, נבין שהגענו ל-γ המקסימלי.

איחוד וחיתוך של חוקים



מה המימד-VC של חוקים שהם איחוד או חיתוך של כמה חוקים פשוטים? יש את המשפט הבא:

יהי $H \supseteq h_1, h_2, \ldots$ אוסף של חוקים עם מימד-VC d (האוסף בעל מימד d).

יהי $H' = \left\{\bigcup_{i=1}^{s} h_{j_i}\right\}$. לדוגמה, אם החוקים ב-$H$ הם כדורים, ו- $s = 3$, אז H' מכיל את כל החוקים שהם חיבור של 3 כדורים.

אזי, $VCdim(H') < 2ds\log_2(3s)$.

כלומר, אם ייצרנו חוקים שהם לא חיבור של יותר מדי חוקים, המימד-VC נשאר יחסית קרוב למה שהוא היה.

נוכיח את המשפט:

מתקיים: $\Pi(H',P) \leq^{א} \left(\Pi(H,P)\right)^s \leq^{ב} \left(\frac{en}{d}\right)^{ds}$. חסם על מספר הצביעות האפשרי שהאוסף החדש מאפשר.

א – מספר הצביעות האפשרי של H, בחזקת s. כי חיברנו s חוקים. ב – הלמה של סאור.

יהי P אוסף הנקודות הגדול ביותר שניתן לניפוץ ע"י H'. כלומר $2^n < (en/d)^{ds}$.

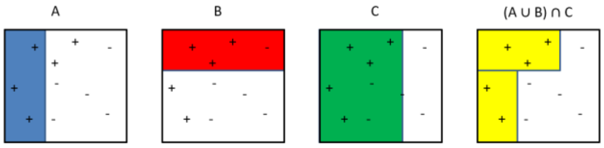
ניקח את ה-n המקסימלי שמקיים את $2^n < (en/d)^{ds}$: $n \ge 2ds\log_2(3s)$, ונציב:

$$2^{2ds\log_2(3s)} < \left(\frac{e2ds\log_2(3s)}{d}\right)^{ds}$$

$$\frac{9s}{2e} < \log_2(3s)$$

זה לא מתקיים עבור $s \ge 2$. כלומר, כדי שזה יתקיים, צריך שיתקיים: $n < 2ds\log_2(3s)$. הגבלנו את גודל האוסף P, וזה מגביל את המימד-VC.

אותה ההוכחה עובדת גם לחיתוך (אבל לא לשניהם ביחד).



אלגוריתם *adaboost* (1995):

נניח שיש לי הרבה חוקים חלשים: לכל אחד מהם יש טעות פחות מ-50% אבל לא הרבה יותר טוב מזה (אם הטעות יותר מחצי, ניקח את הנגדי שלו). (לדוגמה בגילוי ספאם – הרבה מילים שכל אחת מהן מרמזת על ספאם אבל לא כל אחת מהן חזקה מספיק. או בהימורים על מרוצי סוסים – הרבה דברים שמרמזים שהסוס ינצח, אבל אין משהו אחד שנותן אחוזים גבוהים). אלגוריתם רוצה לקחת הרבה חוקים כאלה ולייצר מהם חוק חזק.

השיטה: ניתן לכל חוק משקל – חשיבות. ונעשה "הצבעה" ממושקלת.

יש פה *bias-variance tradeoff*: ה-*bias* זה הטעות האמפירית, ה-*variance* זה כמה החוק מסובך (המימד-VC). כשאנחנו מחזקים את החוק, זה מגדיל את המימד-VC ואת החסם על הטעות שלו. זה יכול להוביל ל-*overfitting* – טעות אמפירית נמוכה אבל מימד-VC גבוה. החוק מדויק על המדגם אבל גרוע על העולם.

קלט:

- קבוצה S של נקודות, $x_i \in S$. לכל נקודה יש תיוג y_i.
- מספר איטרציות רצוי k. באלגוריתם, המשקל של כל חוק יתחיל ב-0, ובכל איטרציה האלגוריתם ייתן משקל לחוק.
- קבוצה H של T חוקים חלשים: $h_j : S \to \{-1,1\}^{|S|}$.

פלט: משקל α_j לכל חוק h_j.

פונקציית ההחלטה הסופית: $F(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$. והקביעה: $H(x) = sign(F(x))$.

במהלך הריצה, האלגוריתם נותן משקל לכל **נקודה.** סכום המשקלים האלה הוא תמיד 1 (כדי שהטעות תישאר בין 0-1).

בכל איטרציה, האלגוריתם בוחר חוק שנראה לו חשוב, ומחשב את המשקל עבורו. הטעות הממושקלת הזו היא לפי המשקלים שיש לנקודות. יש נקודות שיותר "יקר" לטעות בהן.

1) נאתחל משקלי נקודות: $D_1(x_i) = 1/n$.
2) עבור $t = 1,2,\ldots,k$:
   a. נחשב טעות ממושקלת לכל חוק $h \in H$ (הסוגריים המרובעות הן אינדיקטור). הטעות היא סכום המשקלים של הנקודות שבהן טעינו:

$$\epsilon_t(h) = \sum_{i=1}^{n} D_t(x_i)[h(x_i) \ne y_i]$$

   b. נבחר את החוק עם הטעות המינימלית:

$$h_t = argmin_h(\epsilon_t(h))$$

   c. נגדיר את המשקל של החוק לפי הטעות:

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_t(h_t)}{\epsilon_t(h_t)}\right)$$

אם $\epsilon_t(h_t)$ קרוב לאפס, המשקל של החוק גבוה. אם הוא קרוב לחצי, המשקל נמוך.

d. נעדכן את המשקלים של הנקודות:

$$D_{t+1}(x_i) = \frac{1}{Z_t} D_t(x_i) e^{-\alpha_t \cdot h_t(x_i) \cdot y_i}$$

כאשר Z_t הוא קבוע מנרמל שנותן $\sum_i D_{t+1}(x_i) = 1$. (סכום משקלי הנקודות הוא 1).

אם $h_t(x_i) = y_i$, אז $h_t(x_i)y_i = 1$, ומשקל הנקודה יורד (כי כבר יש לי חוק שתופס אותה, אז לא מפריע לי לטעות בה בפעם הבאה).
אם $h_t(x_i) \neq y_i$, אז $h_t(x_i)y_i = -1$, ומשקל הנקודה עולה.

ניתוח האלגוריתם:

$$D_{t+1}(x_i) \overset{\text{א}}{=} \frac{1}{Z_t} D_t(x_i) e^{-\alpha_t \cdot h_t(x_i) \cdot y_i} =$$

$$\overset{\text{ב}}{=} \frac{1}{Z_t Z_{t-1}} D_{t-1}(x_i) e^{-y_i\left(\alpha_t \cdot h_t(x_i) + \alpha_{t-1} h_{t-1}(x_i)\right)} =$$

$$\overset{\text{ג}}{=} \frac{1}{Z_t Z_{t-1} \cdots Z_1} D_1(x_i) e^{-y_i \sum_{j=1}^{t} \alpha_j h_j(x_i)} =$$

$$\overset{\text{ד}}{=} \frac{1}{Z} \cdot \frac{1}{n} e^{-y_i F(X_i)}$$

Z הוא הקבוע המנרמל: $Z = Z_t Z_{t-1} \cdots Z_1 \overset{\text{ה}}{=} \frac{1}{n} \sum_{i=1}^{n} e^{-y_i F(X_i)}$.

הסבר:

א – משקל הנקודה x_i באיטרציה ה-$t+1$, הוא המשקל מהאיטרציה האחרונה, כפול ה-e בחזקה (שמעלה או מוריד את המשקל).

ב – וגם באיטרציה הזו, זה נקבע מהמשקל באיטרציה הקודמת, עם אותה נוסחה.

ג – נלך אחורה עד שנגיע ל-D_1.

ד – המשקל המקורי הוא $1/n$. הסיגמה שיש בחזקה זה הנוסחה של פונקציית ההחלטה הסופית.

ה –הסכום על כל הנקודות שווה 1:

$$\sum_i D_{t+1}(x_i) = \frac{1}{Z} \cdot \frac{1}{n} \sum_{i=1}^{n} e^{-y_i F(X_i)} = 1$$