מבוא ללמידה חישובית (20942) | סיכום הרצאה 7

מנחה: שי מימון

סמסטר: 2022א'

נכתב על ידי: מתן כהן

1 Soft-Margin SVM

גם במקרה של SVM כמו עם הפרספטרון, נוכל להתקל במצבור דוגמאות שלא ניתן להפרדה מושלמת בצורה לינארית. במקרה וננסה לפתור את הבעיה עם Hard-Margin SVM נבחין כי זה בלתי אפשרי - כלומר לנסות לפתור בעיה בה קבוצת האימון שלנו לא ספרבילית בצורה מושלמת על ידי מפריד לינארי על פי המתכון של Hard-Margin SVM - **לא ניתן!**

1.1 הגדרת ה Soft-Margin SVM

ניזכר כי הבעיה הראשונית שלנו הייתה כזו:

$$\begin{aligned}&\underset{\mathbf{w},b}{\textbf{minimize}}\ \frac{1}{2}\mathbf{w}^T\mathbf{w}\\ &\textbf{s.t}\quad y_n\cdot\left(\mathbf{w}^T\mathbf{x}_n+b\right)\geq 1,\quad n=1,..,N\end{aligned}$$

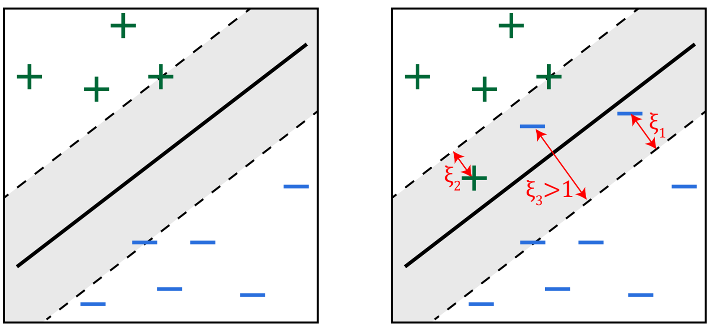
ואמרנו שהאילוצים שלנו לא יכולים להתקיים כאשר המידע שלנו לא ספרבילי.

על מנת לגשר על הפער - נשנה את האילוצים כך שנרשה לחלק מהנקודות להיות בתוך השוליים שלנו ונגדיר מחדש את בעית האופטימיזציה עם נקודות שיקראו "**slack variables**":

$$\begin{aligned}&\underset{\mathbf{w},b,\boldsymbol{\xi}}{\textbf{minimize}}\ \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\cdot\sum_{n=1}^{N}\xi_n\\ &\textbf{s.t}\quad y_n\cdot\left(\mathbf{w}^T\mathbf{x}_n+b\right)\geq 1-\xi_n,\quad n=1,..,N\\ &\qquad\ \xi_n\geq 0,\quad n=1,..,N\end{aligned}$$

כאשר:

- "נשלם" על כל נקודה שעברה את השוליים וננסה **להביא את המחיר למינימום**
  - ישנו trade-off בין הרחבת השוליים לבין צמצום הנקודות שנכנסו לשוליים
- כל נקודה $\mathbf{x}_n$ עבורה $\xi_n\in(0,1)$ תניב לנו **סיווג נכון אך בתוך השוליים**
- כל נקודה $\mathbf{x}_n$ עבורה $\xi>1$ הן נקודות **שלא תסווגנה נכון**



Hard-Margin　　Soft-Margin

1.2 הגדרת הבעיה בצורה שונה

כעת, כפי שעשינו ב- Hard-Margin SVM נוכל גם כאן להשתמש בלגראנז'יאן.
אסור לשכוח להביא את האילוצים לצורה $f_i \leq 0$ ולכן האילוץ החדש שהגדרנו יהפוך ל:

$$\begin{aligned} & y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \xi_n \\ \iff & 1 - \xi_n - y_n(\mathbf{w}^T\mathbf{x}_n + b) \leq 0 \end{aligned}$$

$$\boxed{\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \left(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{x}_n + b)\right) - \sum_{n=1}^{N} \beta_n \xi_n}$$

כעת נגזור את הלגראנז'יאן על פי $\mathbf{w}$, b ו-$\boldsymbol{\xi}$ ונשווה ל-0 על מנת למצוא מינימום

- על פי ξ:

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \quad \Rightarrow \quad C = \alpha_i + \beta_i$$

- על פי b:

$$\frac{\partial \mathcal{L}}{\partial b} = -\sum_{n=1}^{N} \alpha_n y_n = 0$$

- על פי $\mathbf{w}$:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{w} - \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n \quad \Rightarrow \quad \mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n$$

נכניס את הערכים למשוואה המקורית:

$$\begin{aligned} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \left(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{x}_n + b)\right) - \sum_{n=1}^{N} \beta_n \xi_n \\ \backslash \beta_i = C - \alpha_i &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \left(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{x}_n + b)\right) - \sum_{n=1}^{N} (C - \alpha_n) \cdot \xi_n \\ &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n - \sum_{n=1}^{N} \alpha_n \xi_n + \sum_{n=1}^{N} \alpha_n \left(1 - y_n(\mathbf{w}^T\mathbf{x}_n + b)\right) - C \cdot \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \xi_n \\ &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{n=1}^{N} \alpha_n \left(1 - y_n(\mathbf{w}^T\mathbf{x}_n + b)\right) \\ &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{n=1}^{N} \alpha_n - \sum_{n=1}^{N} \alpha_n y_n b - \sum_{n=1}^{N} \alpha_n y_n \mathbf{w}^T\mathbf{x}_n \\ \backslash -\sum_{n=1}^{N} \alpha_n y_n b = 0 &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{n=1}^{N} \alpha_n - \mathbf{w}^T \cdot \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n \\ \backslash \mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n &= -\frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{n=1}^{N} \alpha_n - \mathbf{w}^T\mathbf{w} \\ &= -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \mathbf{x}_n^T \mathbf{x}_m + \sum_{n=1}^{N} \alpha_n \end{aligned}$$

קיבלנו בעיה זהה לבעית ה - Hard-Margin SVM ולכן נוכל לרשום גם בעית אופטימיזציה זהה עם האילוצים שקיבלנו בעת הגזירה והשוואה ל-0:

$$\begin{aligned}
\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \ & \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n^T\mathbf{x}_m - \sum_{n=1}^{N}\alpha_n \\
\text{s.t} \ & \alpha_n \geq 0, \quad n=1,..,N \\
& \beta_n \geq 0, \quad n=1,..,N \\
& \alpha_n + \beta_n = C, \quad n=1,..,N \\
& \sum_{n=1}^{N}\alpha_n y_n = 0
\end{aligned}$$

נבחין כי: $\alpha_n + \beta_n = C \quad \Rightarrow \quad \alpha_n = C - \beta_n$ כיוון ש $\beta_n \geq 0$ הרי ש:

$$\alpha_n \leq C$$

וקיבלנו בעיה זהה לבעית ה Hard-Margin SVM למעט האילוץ על C:

$$\boxed{\begin{aligned}
\min_{\boldsymbol{\alpha}} \ & \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n^T\mathbf{x}_m - \sum_{n=1}^{N}\alpha_n \\
\text{s.t} \ & \alpha_n \geq 0, \quad n=1,..,N \\
& \alpha_n \leq C, \quad n=1,..,N \\
& \sum_{n=1}^{N}\alpha_n y_n = 0
\end{aligned}}$$

את בעיה זו ניתן לנסח בתור convex quadratic programming ולקבל $\boldsymbol{\alpha}^* \leftarrow QP(Q_0, \mathbf{p}_0, A_0, \mathbf{c}_d)$.
כמו כן:

$$\mathbf{w}^* = \sum_{\alpha_s^*>0}\alpha_s^* \cdot y_s \cdot \mathbf{x}_s$$

כמו-כן ה-Complementary Slackness של ה-KKT:

$$\begin{aligned}
&1. \ \alpha_n \cdot \big(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{x}_n + b)\big) = 0, \quad n=1,..,N \\
&2. \qquad\qquad \beta_n \cdot \xi_n = (C - \alpha_n) = 0, \quad n=1,..,N
\end{aligned}$$

מכך נסיק:

(1) $\underline{\alpha_s^* > 0}: \ \boldsymbol{y_s}(\mathbf{w}^{*\boldsymbol{T}}\mathbf{x}_{\boldsymbol{s}} + \boldsymbol{b}^*) = 1 - \xi_s \leq 1$

(2) $\underline{\beta > 0 \iff (C - \alpha_s^*) > 0 \iff \alpha_s^* < C}: \ \xi_s = 0 \Rightarrow \boldsymbol{y_s}(\mathbf{w}^{*\boldsymbol{T}}\mathbf{x}_{\boldsymbol{s}} + \boldsymbol{b}^*) \geq 1$

(3) מ- $(1)+(2)$ נסיק שאם: $\underline{0 < \alpha_s^* < C}$:

(א) $\boldsymbol{y_s}(\mathbf{w}^{*\boldsymbol{T}}\mathbf{x}_{\boldsymbol{s}} + \boldsymbol{b}^*) = \mathbf{1}$

(4) לכן הנקודה $\mathbf{x}_s, y_s$ נמצאת על השוליים (margin support vector), נוכל לחלץ את b^*:

(א) $y_s(\mathbf{w}^{*T}\mathbf{x}_s + b^*) = 1 \overset{\backslash y_s}{\iff} b^* = y_s - \mathbf{w}^{*T}\mathbf{x}_s$

(5) לכל הנקודות עבורן $\alpha_s^* = C$ נקרא bounded/non-margin support vectors

2 מידע ניתן להפרדה לא ליניארית

2.1 שימוש בטרנספורמציות

ניזכר כי כאשר רצינו לבצע סיווג בעזרת פרספטרון למידע שניתן להפרדה אך לא בצורה ליניארית, עשינו טרנספורמציה לפיצ'רים לסט פיצ'רים אחר או ממימד גבוה יותר ושם פתרנו את הבעיה בעזרת פרספטרון.

גם כאן נוכל להשתמש באותו "טריק".

נגדיר טרנספורמציה (**לא בהכרח ליניארית**):

$$\phi : \mathbb{R}^d \longrightarrow \mathbb{R}^{\tilde{d}}$$

כך שעבור סט פיצ'רים $\mathbf{x}_n \in \mathbb{R}^d$ נקבל:

$$\mathbf{z}_n = \phi(\mathbf{x}_n) \in \mathbb{R}^{\tilde{d}}$$

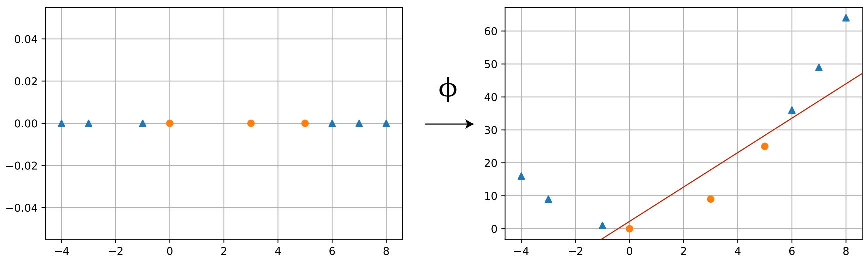
ונפתור את בעית ה-Hard-Margin SVM:

$$\begin{aligned} &\underset{\tilde{b}\in\mathbb{R},\mathbf{w}\in\mathbb{R}^{\tilde{d}}}{min} \frac{1}{2}\tilde{\mathbf{w}}^T\tilde{\mathbf{w}} \\ &\textbf{s.t} \quad y_n(\tilde{\mathbf{w}}^T\mathbf{z}_n + \tilde{b}) \geq 1, \quad n = 1, .., N \end{aligned}$$

נקבל $QP \leftarrow \begin{bmatrix} \tilde{b}^* \\ \tilde{\mathbf{w}}^* \end{bmatrix}$ והחזאי האופטימלי:

$$\boxed{g(\mathbf{x}) = sign(\tilde{\mathbf{w}}^{*T}\phi(\mathbf{x}) + \tilde{b}^*)}$$

דוגמה: שימוש בטרנספורמציה $\phi(x) = (x, x^2)$



בפועל - בד"כ נשתמש בטרנספורמציה **לא ליניארית** וב- **Soft-Margin SVM**!

הערה: ***יש להזהר*** עם אופן הגדלת המימד מ-2 סיבות עיקריות:

- **ככל שנגדיל** את המימד ונקבל מימד שגודלו כמספר הנקודות שיש לנו בסט הדוגמאות - **כך אנחנו עלולים לגרום ל- overfit** - נדבר על כך בהמשך הקורס.
- במצבים מסויימים נגדיל את המימד עד כדי כך שהבעיה הופכת להיות מורכבת מבחינה חישובית, בהכרח אם נפתור בעיות ממרחב אינסופי

נפתור את בעית גודל המימד בתתי הסעיפים הבאים.

2.1.1 התמודדות עם מימדים גדולים

על מנת להתמודד עם ממדים גדולים נוכל להשתמש בדואליות - **שם אנחנו תלויים במספר הדוגמאות ולא במימדן!**
ניזכר כי הביטוי לבעיה הדואלית של ה-SVM הוא:

$$\begin{aligned} &\min_{\boldsymbol{\alpha}} \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n^T\mathbf{x}_m - \sum_{n=1}^{N}\alpha_n \\ &\textbf{s.t}\ \ \alpha_n \geq 0, \quad n = 1,..,N \\ &\qquad \sum_{n=1}^{N}\alpha_n y_n = 0 \end{aligned}$$

כאשר:

- $\boldsymbol{\alpha} \in \mathbb{R}^N$
- ישנם $N+1$ אילוצים

לאחר המעבר למרחב $\mathbb{R}^{\tilde{d}}$ נרשום את הבעיה:

$$\begin{aligned} &\min_{\tilde{\boldsymbol{\alpha}}\in\mathbb{R}^N} \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\tilde{\alpha}_n\tilde{\alpha}_m y_n y_m \mathbf{z}_n^T\mathbf{z}_m - \sum_{n=1}^{N}\tilde{\alpha}_n \\ &\textbf{s.t}\ \ \tilde{\alpha}_n \geq 0, \quad n = 1,..,N \\ &\qquad \sum_{n=1}^{N}\tilde{\alpha}_n y_n = 0 \end{aligned}$$

נשתמש ב QP על מנת לפתור את הבעיה ולקבל $\tilde{\mathbf{w}}^* = \sum_{n=1}^{N}\tilde{\boldsymbol{\alpha}}_n^* \cdot y_n \cdot \mathbf{z}_n$ ו- $\tilde{b}^*$ ולקבל חזאי

$$\begin{aligned} g(\mathbf{x}) &= sign(\tilde{\mathbf{w}}^T\phi(\mathbf{x}) + \tilde{b}^*) \\ &= sign\left(\sum_{n=1}^{N}\tilde{\boldsymbol{\alpha}}_n^* \cdot y_n \cdot \phi(\mathbf{x}_n)\cdot\phi(\mathbf{x}) + \tilde{b}^*\right) \end{aligned}$$

עם זאת, צריך להבחין בעובדה שהמכפלה $\mathbf{z}_n^T\mathbf{z}_m$ היא עדיין מכפלה של וקטורים ממימד $\tilde{d}$ ונתמודד עם העניין בעזרת **Kernel Trick**

2.1.2 Kernel Trick

ראינו שעל מנת לפתור את בעית האופטימיזציה יש צורך לפתור פונקציות מהצורה הבאה:

$$\underbrace{K(\mathbf{x}', \mathbf{x}'')}_{\text{Kernel}} = \phi^T(\mathbf{x}')\cdot\phi(\mathbf{x}'')$$

בצורה הישירה היינו מחשבים את $\phi(\mathbf{x}')$ ואת $\phi(\mathbf{x}'')$ ומכפילים
אך ברצוננו לעשות דבר יעיל יותר, ו - Kernel Trick בא לענות על השאלה:

האם ניתן לחשב את המכפלה הפנימית מבלי לעבור דרך המרחב החדש שיצרנו?

התשובה לשמחתינו - היא **כן!**

דוגמה: נתבונן ב: $K(\mathbf{x}',\mathbf{x}'') = \left(1+\mathbf{x}'^T\cdot\mathbf{x}''\right)^2, \quad \mathbf{x}'\in\mathbb{R}^2, \mathbf{x}''\in\mathbb{R}^2$

ונפתור:

$$\begin{aligned}
K(\mathbf{x}',\mathbf{x}'') &= \left(1+\mathbf{x}'^T\cdot\mathbf{x}''\right)^2, \quad \mathbf{x}'\in\mathbb{R}^2, \mathbf{x}''\in\mathbb{R}^2 \\
&= \left(1+\begin{bmatrix} x_1' & x_2' \end{bmatrix}\cdot\begin{bmatrix} x_1'' \\ x_2'' \end{bmatrix}\right)^2 \\
&= \left(1+x_1'x_1''+x_2'x_2''\right)^2 \\
&= 1+\left(x_1'\right)^2\left(x_1''\right)^2+\left(x_2'\right)^2\left(x_2''\right)^2+2x_1'x_1''+2x_2'x_2''+2\cdot x_1'x_2'\cdot x_1''x_2'' \\
&= \underbrace{\begin{bmatrix} 1 \\ \sqrt{2}\mathbf{x}_1' \\ \sqrt{2}\mathbf{x}_2' \\ \left(x_1'\right)^2 \\ \left(x_2'\right)^2 \\ \sqrt{2}x_1'x_2' \end{bmatrix}^T}_{\phi^T(\mathbf{x}')} \cdot \underbrace{\begin{bmatrix} 1 \\ \sqrt{2}\mathbf{x}_1'' \\ \sqrt{2}\mathbf{x}_2'' \\ \left(x_1''\right)^2 \\ \left(x_2''\right)^2 \\ \sqrt{2}x_1''x_2'' \end{bmatrix}}_{\phi(\mathbf{x}'')}
\end{aligned}$$

מסקנה: הצלחנו לחשב מכפלה פנימית שעולה לנו $O(d)$ $(d=2)$ ולהגיע למכפלה פנימית בממד הגבוה יותר $\tilde{d}$

ניתן להרחיב את מה שהראינו ל **Kernel פולינומיאלי**:

$$\boxed{\begin{aligned}
K(\mathbf{x}',\mathbf{x}'') &= \phi(\mathbf{x}')\cdot\phi(\mathbf{x}'') \\
&= \left(\xi+\zeta\cdot\mathbf{x}'^T\cdot\mathbf{x}''\right)^Q \quad \text{- Valid Kernel}
\end{aligned}}$$

ניתן גם לרחיב עבור **Kernel גאוסי (RBF - Radial Basis Function)** שמדמה מעבר **למרחב אינסופי**:

$$\boxed{K(\mathbf{x}',\mathbf{x}'') = exp\left\{-\gamma\cdot\left\|\mathbf{x}'-\mathbf{x}''\right\|^2\right\}}$$

הערה: על מנת ש-Kernel יהיה ולידי (Valid) יש צורך שיקיים:

- סימטריות
- לכל N ולכל $x_1,..,x_N$:

$$\begin{bmatrix} K(\mathbf{x}_1,\mathbf{x}_1) & K(\mathbf{x}_1,\mathbf{x}_2) & \dots & K(\mathbf{x}_1,\mathbf{x}_N) \\ \vdots & \ddots & \ddots & \vdots \\ K(\mathbf{x}_N,\mathbf{x}_1) & K(\mathbf{x}_N,\mathbf{x}_2) & \dots & K(\mathbf{x}_N,\mathbf{x}_N) \end{bmatrix} \succeq 0$$