מבוא למערכות לומדות - הרצאה 1 - מודל PAC ללמידה

16 במאי 2015

ההרצאה היום תסוב בעיקר סביב ההגדרה של בעיית למידה. כפי שנראה, עצם ההגדרה איננו טיריוויאלי כלל. המשימה הבסיסית ביותר בלמידה חישובית היא ללמוד **מיפוי**

$$h^* : X \to Y$$

כאשר X הוא **מרחב של קלטים** ו־Y הוא **מרחב של פלטים**. דוגמאות אופייניות למיפויים שנרצה ללמוד בעזרת למידה חישובית הם:

1. **זיהוי אובייקטים בתמונות.** כאן, מרחב הקלטים יכול להיות, למשל, אוסף התמונות שחור־לבן בגודל 100×100. ואז X עשוי להיות מרחב המטריצות

$$X = M_{100\times 100}([0,1])$$

מרחב הפלטים יכול להיות אוסף של אובייקטים שעשויים להופיע בתמונות למשל,

$$Y = \{\text{"dog"}, \text{"cat"}, \text{"plane"}, \text{"non of the aobve"}\}$$

והמיפוי $h^* : X \to Y$ פשוט יתאים כל תמונה לאובייקט שמופיע זה

2. **זיהוי ספאם.** כאן, מרחב הקלטים יהיה אוסף קבצי הטקסט המכילים, נאמר, פחות ממליון תווים. אז (אם יש לנו שני תווים) אפשר לקחת

$$X = \cup_{n=0}^{10^6}\{\pm 1\}^n$$

מרחב הפלטים יהיה מאד פשוט - $Y = \{\text{"spam"}, \text{"not-spam"}\}$ והמיפוי $h^* : X \to Y$ פשוט יתייג כל מייל אפשרי בתור ספאם או מייל כשר.

3. **התאמת מקטעי שמע לטקסט.** כאן, נרצה ללמוד מיפוי המתאים למקטע קול (נאמר, קובץ WAV) טקסט המתאר את מה ששומעים.

4. **תרגום טקסט.**

5. **התאמת רצפי DNA לתכונות גנטיות.**

6. **אבחון מחלות לפי סימפטומים ונתונים רפואיים.**

7. **חיזוי מזג אויר על סמך נתונים מטארולוגיים.**

8. **חיזוי ביצועים של מניה על סמך נתונים כלכליים.**

נעיר שבכל הבעיות הנ"ל המערכות שמשיגות את הביצועים הטובים ביותר כיום מבוססות על למידה חישובית.

מרחב הקלטים X

ייצוג הקלטים. הקלטים של המיפויים אותם נרצה ללמוד שונים ומגוונים - תמונות, מקטעי קול, קבצי טקסט ועוד. על מנת לפתח אלגוריתמים כלליים, שלא תלויים בבעיה הספציפית, המרחב X לא יהיה מרחב ה"קלטים עצמם", אלא מרחב המכיל את הייצוגים האפשריים של הקלטים. האלגוריתמים וניתוחם לא יהיו מודעים לאופן שבו מופו הקלטים ה"אמיתיים" לאיברים ב־X, ומבחינתם X הוא מרחב הקלטים ה"אמיתי". נעיר שהאופן שבו מיוצגים הקלטים חשוב מאד לביצועים שנקבל, ובתרגול תלמדו מספר דרכים סטנדרטיות לייצג תמונות וטקסטים.

הגודל של X ויעילות. בד"כ נייצג את הקלטים ע"י וקטורים (או מטריצות) עם n ערכים דיסקרטיים או רציפים. בפרט, מספר הביטים הנדרשים לייצוג כל קלט הוא פרופורציונאלי ל־n, והגודל של X אקספוננציאלי ב־n. ברוב הדוגמאות - n יהיה גדול וינוע בין כמה עשרות לכמה מליונים. למשל, n יכול להיות מספר הפיקסלים בתמונה או מספר המילים במילון שלנו (מספר המילים שעשויות להופיע בקובץ). אנו ניקח את n בתור "פרמטר הסיבוכיות" שלנו, ונדרוש מהאלגוריתמים להיות פולינומאליים ב־n. נעיר שדרישת היעילות שלנו דומה לדרישה הסטנדרטית באלגוריתמים, כלומר, להיות פולינומיאלים בגודל הקלט. עם זאת, בשונה מאלגוריתמים "רגילים", הדרישה להיות פולינומיאלי ב־n חלה לא רק על הזמן שלוקח לחשב את h^* על קלט בודד, גם על הזמן שלוקח ללמוד את h^* וגם על מספר הדוגמאות שנזדקק לו על מנת ללמוד את h^*.

מרחב הפלטים Y

כמו מרחב הקלטים, גם מרחבי הפלטים משתנים מאד כאשר עוברים בין בעיות שונות - הם יכולים להיות אובייקטים העשויים להופיע בתמונה, משפטים, מספרים ממשיים וכו'. בניגוד ל־X שהוא תמיד מאד גדול, Y יכול להיות מאד גדול (למשל, מרחב המשפטים), אך לפעמים הוא גם מאד קטן (למשל, בזיהוי ספאם Y מכיל שני איברים). ברוב הקורס אנחנו נביט על בעיות בהן מרחב הקלטים הוא קטן ופשוט - אוסף סופי וקטן $Y=\{1,\ldots,k\}$ או מספר ממשי בודד $Y=[-M,M]$.

1 לימוד מנתונים

נניח שאנחנו רוצים לכתוב תכנה המחשבת מיפוי $h^*:X\to Y$. באלגוריתמיקה קלאסית (למשל מה שלמדתם בקורסים "מבוא למדעי המחשב", "מבני נתונים" ו־"אלגוריתמים") מפתחים אלגוריתמים המחשבים את המיפוי $h^*:X\to Y$. כלומר, המתכנת כותב קוד

(יעיל) אשר בהינתן קלט $x \in X$ מפעילה אלגוריתם המחשב את $h^*(x)$. כפי שכבר ראיתם לא פעם ולא פעמיים, השיטה הנ"ל עובדת מצויין בשורה של משימות (סידור מערך, מציאת מסלול קצר בגרף, ...). עם זאת, בהרבה משימות השיטה הנ"ל נכשלת. לעיתים הסיבה לכשלון היא שבכלל לא קיים אלגוריתם יעיל המחשב את h^* (על זה תלמדו \ למדתם בקורס "חישוביות"). במקרים אחרים, הסיבה לכשלון הפרדיגמה הקלאסית היא אחרת - לפעמים קיים אלגוריתם יעיל המחשב את המיפוי, אך התיאור שלו מאד מורכב, או שאנחנו לא יודעים את התיאור שלו. נביט בבעיה קונקרטית - התאמת תמונות לעצמים המופיעים בהן. אנחנו יודעים שקיים אלגוריתם יעיל הפותר את הבעיה - המוח האנושי מסוגל להתאים לתמונה את האובייקט המופיע בה. עם זאת, התיאור של האלגוריתם ככל הנראה מאד מורכב - המעגל החשמלי שמחשב את ההתאמה הנ"ל ככל הנראה מאד גדול ומורכב. גרוע מכך, אנחנו לא יודעים כלל מהו!

למידה חישובית מהווה גישה להתמודד עם סיטואציות כאלו. במקום לקודד ישירות את האלגוריתם, נביט **במדגם אימון**, כלומר, באוסף של זוגות של קלט ופלט

$$(x_1, h^*(x_1)), \ldots, (x_m, h^*(x_m)) \tag{1}$$

ונריץ אלגוריתם למידה שיקבל את המדגם הנ"ל בתור קלט ויחזיר בתור פלט פונקציה $h : X \to Y$ המקיימת $h(x) = h^*(x)$ לפחות עבור רוב הקלטים.

הגדרה 1.1 אלגוריתם למידה הוא אלגוריתם המקבל בתור קלט מדגם אימון

$$(x_1, y_1), \ldots, (x_m, y_m)$$

ומחזיר פונקציה $h : X \to Y$.

נעיר שליתר דיוק, אלגוריתם הלמידה יחזיר תיאור של אלגוריתם יעיל המחשב את h. עם זאת, הרבה פעמים נוח לחשוב על אלגוריתם הלמידה בתור אלגוריתם המחזיר פונקציה.

מודל ייצור הנתונים

הגישה הנ"ל מעלה שתי נקודות שצריך לתת את הדעת עליהן:

- **כיצד מיוצר המדגם?** ישנם הרבה מאד דרכים לבחור את אוסף הנקודות $x_1, \ldots, x_m$ שנזין לאלגוריתם בתור מדגם אימון. מהי הדרך הנכונה לבחור את מדגם האימון?
- **כיצד נבחן את ביצועי האלגוריתם?** הרבה פעמים, הפונקציה h שהאלגוריתם יחזיר לא תהיה שווה ל־h^* אלא רק תקרב אותה על "רוב" הקלטים. כיצד נדע שהפונקציה אותה החזיר האלגוריתם היא באמת טובה?

אינטואיטיבית, נרצה שהפונקציה h "תתנהג כמו h^*" על "רוב" הקלטים עליהם נעריך את h **בפועל**. למשל, אם אנחנו רוצים לזהות ספאם, אנו נרצה שהפלט של האלגוריתם יהיה מוצלח על מיילים אמיתיים שלגביהם נידרש להכריע האם הם ספאם או לא. כדי להשיג את המטרה הנ"ל מועיל שהדוגמאות שעליהן האלגוריתם יאומן ייוצרו באופן דומה לדוגמאות עליהן h תופעל. מלבד כלל אצבע כיצד לייצר את מדגם האימון, האבחנה הנ"ל מהווה בסיס לאופן בו המודל הסטנדרטי ללמידה נותן מענה לשתי הנקודות הללו. אנו נניח שהדוגמאות עליהן תופעל h בפועל מיוצרות בדיוק באותו אופן כמו הדוגמאות עליהן אומן האלגוריתם. קונקרטית, נניח שקיימת התפלגות $\mathcal{D}$ על X כך ש־

- הדוגמאות $x_1, \ldots, x_m$ נדגמו לפי $\mathcal{D}$ באופן בלתי תלוי. כלומר $x_1, \ldots, x_m$ הם משתנים מקריים בלתי תלויים שההתפלגות של כל אחד מהם היא $\mathcal{D}$.
- הפלט של האלגוריתם הוא "טוב" אם כאשר x הוא משתנה מקרי המתפלג לפי $\mathcal{D}$ מתקיים ש-$h(x) = h^*(x)$ בהסתברות גבוהה, או, כאשר הפלט רציף, $h(x)$ "קרוב" ל-$h^*(x)$.

פורמאלית, נגדיר

הגדרה 1.2 (זמנית) התפלגות מייצרת נתונים היא התפלגות $\mathcal{D}$ על X.

הגדרה 1.3 פונקציית הפסד (loss function) היא פונקציה $l : Y \times Y \to \mathbb{R}_+$ המקיימת $l(y, y) = 0$.

שתי דוגמאות עיקריות לפונקציית הפסד הן

- **zero-one loss**. $l_{0-1}(\hat{y}, y) = \begin{cases} 1 & y \neq \hat{y} \\ 0 & y = \hat{y} \end{cases}$
- **square loss**. כאן, $Y = \mathbb{R}$ ו-$l(\hat{y}, y) = (y - \hat{y})^2$

הגדרה 1.4 ההפסד (loss) או **השגיאה (error)** של $h : X \to Y$ היא $L_{\mathcal{D},h^*}(h) = E_{x \sim \mathcal{D}} l(h(x), h^*(x))$.

במילים, השגיאה היא המרחק הממוצע בין הפלט של h לפלט של h^*. למשל, כאשר $l = l_{0-1}$, ההפסד של h הוא פשוט ההסתברות שהתחזית של h שגויה. כלומר, ההסתברות ש-$h(x) \neq h^*(x)$ כאשר x נדגם לפי $\mathcal{D}$.

מודל ייצור הנתונים - הכללה להתפלגויות לא פרידות (אגנוסטיות)

עד כה הנחנו שכל קלט $x \in X$ ממופה באופן דטרמיניסטי לפלט $h^*(x)$. הרבה פעמים בלמידה אנחנו נרצה ללמוד מיפויים שאינם דטרמיניסטיים. כלומר, מקרים בהם הקלט x לא קובע ביחידות את הפלט, אלא רק מאפשר להסיק מהו בצורה אולי יותר מושכלת. לדוגמא,

- נניח שאנחנו רוצים לחזות האם מניה תרד או תעלה על סמך נתונים כלכליים מסוימים. הקלט כאן הוא הנתונים הכלכליים, בעוד הפלט הוא "עלייה" ו-"ירידה". במקרה כזה, הפלט (ביצועי המנייה) לא יקבע ביחידות על סמך הקלט (הנתונים הכלכליים). עם זאת, הקלט בהחלט עשוי לתת אינדיקציה מסויימת לגבי הפלט.
- דוגמא נוספת היא כאשר אנו רוצים לחזור מחלה על סמך סימפטומים. גם כאן, הסימפטומים לא תמיד יקבעו באופן יחיד את המחלה, אך בהחלט יפסלו חלק גדול מהמחלות ויעלו את הסבירות למחלות אחרות.

כדי למדל סיטואציות כאלו, במקום להניח שיש לנו התפלגות על X ומיפוי $h^* : X \to Y$, אנו נניח שיש התפלגות $\mathcal{D}$ על $X \times Y$. כאשר $(x,y) \in X \times Y$ הוא משתנה מקרי הנדגם לפי $\mathcal{D}$. אנחנו נחשוב של (x,y) בתור זוג של קלט ופלט. במידול הנ"ל כל קלט x משרה התפלגות, $\mathcal{D}(y|x)$, על Y.

הגדרה 1.5 התפלגות מייצרת נתונים היא התפלגות $\mathcal{D}$ על $X \times Y$.

נשים לב שהמקרה הקודם, בו יש התפלגות $\mathcal{D}'$ על X ומתקיים $y = h^*(x)$, מתקבל בתור מקרה פרטי - פשוט ניקח את $\mathcal{D}$ להיות ההתפלגות של $(x, h^*(x))$ כאשר x מתפלג לפי $\mathcal{D}'$. במקרה הנ"ל נאמר ש־$\mathcal{D}$ **פרידה (realizable) ע"י** h^*. נכליל גם את הגדרת ההפסד.

הגדרה 1.6 ההפסד של $h : X \to Y$ ביחס להתפלגות $\mathcal{D}$ ופונקציית הפסד l הינו

$$L_{\mathcal{D}}(h) = E_{(x,y)\sim\mathcal{D}} l(h(x), y)$$

שוב, במילים, ההפסד של h הוא המרחק הממוצע בין התחזית של h לפלט הנכון.

הערה על מינוחים - למידה מול אלגוריתמים

האיברים ב־X בד"כ נקראים **דוגמאות** ולא קלטים. בהתאם, X נקרא **מרחב מדגם**. כמו כן, פונקציות $h : X \to Y$ נקראות **היפותזות** ולא מיפויים.

2 אין ארוחות חינם - הצורך בהנחות ובידע מוקדם

אידיאלית, היינו רוצים אלגוריתם יעיל הפועל טוב על כל התפלגות ועל כל פונקציית מטרה $h^* : X \to Y$. כפי שנראה, התקווה הנ"ל לא תתממש, כלומר, לא קיים אלגוריתם כנ"ל. יתרה מזאת, אנו נראה שלא קיים אלגוריתם כנ"ל אפילו כאשר מקלים מאד את דרישת היעילות, ודורשים רק שמספר הדוגמאות יהיה פולינומיאלי ב־n (כלומר, כאשר מוותרים על הדרישה שהזמן שלוקח להריץ את אלגוריתם הלמידה ואת h הוא פולינומיאלי). אנו נתרכז במקרה שבו $X = \{\pm 1\}^n,\ Y = \{0,1\}$ ו־$l = l_{0-1}$. נגדיר

הגדרה 2.1 נאמר שאלגוריתם למידה $\mathcal{A}$ הוא **"מושלם"** אם קיים קבוע c כך שלכל התפלגות $\mathcal{D}$ ולכל $h^* : X \to Y$, אם מריצים את $\mathcal{A}$ על n^c דוגמאות בלתי תלויות המתפלגות לפי $\mathcal{D}$, הוא יחזיר בסיכוי $\frac{1}{2}$ (על פני בחירת מדגם האימון) פונקציה $h : X \to Y$ המקיימת $L_{\mathcal{D},h^*}(h) < \frac{1}{10}$.

משפט 2.2 (no free lunch) לא קיים אלגוריתם מושלם.

מפאת חוסר זמן, לא נוכיח את המשפט. רעיון ההוכחה הינו פשוט - אם אין לנו כל ידע מוקדם על h^*, אז אין לנו שום דרך לדעת מהו הערך של $h^*(x)$ עבור ערך $x \in X$ שלא הופיע במדגם. לכן, על מנת להחזיר פונקציה עם שגיאה קטנה מ־$\frac{1}{10}$ נהיה מוכרחים לראות במדגם כ־90 אחוז מהאיברים ב־X. במילים אחרות, גודל המדגם חייב להיות לפחות

$0.9 \cdot |X| = 0.9 \cdot 2^n$. בפרט, אם $\mathcal{A}$ רץ רק על n^c דוגמאות, אז לפחות עבור n גדול, הוא יחזיר פונקציה עם שגיאה גדולה.

המסקנה מהמשפט היא שעל מנת להצליח להחזיר פונקציה עם שגיאה נמוכה, חייבים להניח איזשהן הנחות. או על h^*, או על ההתפלגות $\mathcal{D}$ או על שניהם. במילים אחרות, צריך איזשהו "ידע מוקדם". האבחנה שצריכים לעשות הנחות פותחת מגרש מגרש משחקים מאד רחב - יש הרבה מאד הנחות שהיינו יכולים לעשות על ההתפלגות ועל פונקצייה המטרה. ההנחה שעומדת מאחורי מודל PAC היא ש-h^* היא (לפחות בקירוב) פונקציה "פשוטה". כלומר, קיימת **מחלקת היפותזות** $\mathcal{H}$ (אותה אנחנו "יודעים מראש") המכילה "פונקציות פשוטות", כך ש-h שייכת ל-$\mathcal{H}$ או לפחות קרובה לפונקציה מ-$\mathcal{H}$.

3 מחלקות היפותזות

הגדרה 3.1 מחלקת היפותזות $\mathcal{H}$ היא אוסף של פונקציות מ-X ל-Y.

נביט במספר דוגמאות בסיסיות למחלקות היפותזות.

1. **פונקציונאלים אפיניים.** כאן, $X \subset \mathbb{R}^n$, $Y = \mathbb{R}$ ו-$\mathcal{H}$ היא אוסף הפונקציות האפינית מ-X ל-Y. כלומר, אוסף הפונקציות $h : X \to Y$ מהצורה
$$h(x) = a_1x_1 + \ldots + a_nx_n + b$$

2. **פולינומים מדרגה** $\leq d$. כאן, $X \subset \mathbb{R}^n$, $Y = \mathbb{R}$ ו-$\mathcal{H}$ היא אוסף הפולינומים (ב-n משתנים) מדרגה $\leq d$. נשים לב שהדוגמא הקודמת (פונקציונליים אפיניים) היא המקרה הפרטי $d = 1$.

3. **חצאי מרחבים.** כאן, $X \subset \mathbb{R}^n$, $Y = \{\pm 1\}$ ו-$\mathcal{H}$ היא אוסף הפונקציות $h : X \to Y$ מהצורה
$$h(x) = \operatorname{sign}(a_1x_1 + \ldots + a_nx_n + b)$$
כאשר $\operatorname{sign(x)} := \begin{cases} 1 & x > 0 \\ -1 & x \leq 0 \end{cases}$. נעיר שהסיבה שהפונקציות ב-$\mathcal{H}$ נקראות חצאי מרחב היא שכאשר $X = \mathbb{R}^n$, עבור כל $h \in \mathcal{H}$ מתקיים שהקבוצה $h^{-1}(1) \subset \mathbb{R}^n$ היא חצי מרחב.

4. **עצי החלטה בגודל** $\leq t$. כאן, $X = \{\pm 1\}^n$, $Y = \{\pm 1\}$ ו-$\mathcal{H}$ היא אוסף הפונקציות מ-X ל-Y הניתנות למימוש ע"י עץ החלטה בגודל $\leq t$ (ניתן הגדרה מדוייקת בהמשך).

4 מודל PAC (Probably Approximately Correct)

כעת אנו מוכנים להגדרת מודל PAC ללמידה. **בעיית למידה** נקבעת ע"י רביעיה $(X, Y, \mathcal{H}, l)$, כאשר X היא קבוצה הנקראית **מרחב דוגמאות**, Y היא קבוצה הנקראית **מרחב פלטים**, $l : Y \times Y \to \mathbb{R}_+$ היא פונקצייה המקיימת $\forall y,\ l(y, y) = 0$ ונקראית **פונקצייה הפסד** ו-$\mathcal{H}$ היא אוסף של פונקציות הנקרא **מחלקת היפותזות**.

הגדרה 4.1 **השגיאה** של $\mathcal{H}$ ביחס להתפלגות $\mathcal{D}$ על $X \times Y$ היא $L_{\mathcal{D}}(\mathcal{H}) = \inf_{h\in\mathcal{H}} L_{\mathcal{D}}(h)$.

אנו נדרוש מאלגוריתם למידה להחזיר היפותזה עם שגיאה הקרובה ל־$L_{\mathcal{D}}(\mathcal{H})$. אחד המושגים הבסיסיים במודל שלנו הוא **סיבוכיות המדגם** של אלגוריתם. כלומר, מספר הדוגמאות שעליו "לראות", **ללא תלות בהתפלגות** $\mathcal{D}$, על מנת להחזיר (בהסתברות גבוהה) היפותזה עם שגיאה קטנה (קרובה ל־$L_{\mathcal{D}}(\mathcal{H})$). פורמאלית, נגדיר,

הגדרה 4.2 יהא $\mathcal{A}$ אלגוריתם למידה[1]. **סיבוכיות המדגם** של $\mathcal{A}$ עם **פרמטר שגיאה** $\epsilon > 0$ **ופרמטר ודאות** $\delta > 0$ הוא המספר המינימאלי $m_{\mathcal{A}}(\epsilon, \delta)$ כך שלכל התפלגות $\mathcal{D}$, אם

$$S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$$

הוא מדגם הנדגם[2] לפי $\mathcal{D}$ ו־$m \geq m_{\mathcal{A}}(\epsilon, \delta)$ אז מתקיים

$$\Pr_S \left(L_{\mathcal{D}}(\mathcal{A}(S)) > L_{\mathcal{D}}(\mathcal{H}) + \epsilon\right) \leq \delta$$

אנחנו נאמר ש־$\mathcal{H}$ היא **למידה** אם קיים אלגוריתם למידה $\mathcal{A}$ עם סיבוכיות מדגם סופית. אנחנו נאמר ש־$\mathcal{H}$ היא **למידה ביעילות** אם סיבוכיות המדגם פולינומיאלית ב־$\frac{1}{\epsilon}, \frac{1}{\delta}$ ופרמטר הסיבוכיות שלנו (בד"כ X יהיה תת קבוצה של $\mathbb{R}^n$, ואז פרמטר הסיבוכיות יהיה פשוט n), ובנוסף $\mathcal{A}$ הוא אלגוריתם יעיל (רץ בזמן פולינומיאלי בגודל הקלט שלו) והפלט שלו (הפונקציה $h : X \to Y$) ניתן לחישוב ביעילות[3]. אנו נגדיר את **סיבוכיות המדגם של** $\mathcal{H}$ להיות סיבוכיות המדגם של האלגוריתם הטוב ביותר $m_{\mathcal{H}}(\epsilon, \delta) = \inf_{\mathcal{A}} m_{\mathcal{A}}(\epsilon, \delta)$.

ההנחה המובלעת מאחורי מודל PAC היא שבמקרים מסויימים קיימת ב־$\mathcal{H}$ היפותזה המקרבת מאד את המיפוי אותו אנו רוצים ללמוד. כלומר, השגיאה, $L_{\mathcal{D}}(\mathcal{H})$, היא קטנה. אידאלית, היינו רוצים שההתפלגות $\mathcal{D}$ תהיה פרידה ע"י היפותזה $h^* \in \mathcal{H}$. כלומר, מתקיים $y = h^*(x)$ בהסתברות 1, כאשר $(x, y) \sim \mathcal{D}$ (ובפרט, $L_{\mathcal{D}}(\mathcal{H}) = 0$). במקרה הנ"ל נאמר ש־$\mathcal{D}$ **ממומשת (או פרידה או realizable)** ע"י $\mathcal{H}$. הרבה פעמים נתייחס באופן פרטני למקרה הפריד. לכן נגדיר:

הגדרה 4.3 יהא $\mathcal{A}$ אלגוריתם למידה[4]. **סיבוכיות המדגם** של $\mathcal{A}$ במקרה הפריד עם **פרמטר שגיאה** $\epsilon > 0$ **ופרמטר ודאות** $\delta > 0$ הוא המספר המינימאלי $m^r_{\mathcal{A}}(\epsilon, \delta)$ כך שלכל התפלגות פרידה $\mathcal{D}$, אם

$$S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$$

הוא מדגם הנדגם[5] לפי $\mathcal{D}$ ו־$m \geq m^r_{\mathcal{A}}(\epsilon, \delta)$ אז מתקיים

$$\Pr_S \left(L_{\mathcal{D}}(\mathcal{A}(S)) > \epsilon\right) \leq \delta$$

---

[1] כזכור, אלגוריתם למידה מקבל כקלט **מדגם** $(x_1, y_1), \ldots, (x_m, y_m) \in X \times Y$ ומחזיר כפלט פונקציה $h : X \to Y$.

[2] כלומר $(x_1, y_1), \ldots, (x_m, y_m)$ הם מ"מ ב"ת המתפלגים לפי $\mathcal{D}$.

[3] באופן מדוייק, הפלט של אלגוריתם למידה יעיל הוא תיאור של אלגוריתם יעיל המחשב פונקציה $h : X \to Y$. עם זאת, אנחנו נחשוב על h עצמה בתור הפלט של אלגוריתם הלמידה. בד"כ יהיה ברור מההקשר מהו האלגוריתם המחשב את h, אותו מחזיר אלגוריתם הלמידה.

[4] כזכור, אלגוריתם למידה מקבל כפלט **מדגם** $(x_1, y_1), \ldots, (x_m, y_m) \in X \times Y$ ומחזיר כפלט פונקציה $f : X \to Y$.

[5] כלומר $(x_1, y_1), \ldots, (x_m, y_m)$ הם מ"מ ב"ת המתפלגים לפי $\mathcal{D}$.

באופן דומה מגדירים את המושגים **למידה במקרה הפריד, למידה ביעילות במקרה הפריד, וסיבוכיות המדגם של $\mathcal{H}$ במקרה הפריד**. נשים לב שסיבוכיות המדגם (של כל אלגוריתם ושל $\mathcal{H}$) במקרה הפריד לעולם לא גדולה מסיבוכיות המדגם הרגילה (למה?). לכן, אם $\mathcal{H}$ למידה אז $\mathcal{H}$ למידה גם במקרה הפריד (למה?) ואם $\mathcal{H}$ למידה ביעילות אז $\mathcal{H}$ למידה ביעילות גם במקרה הפריד.

5 אלגוריתמי ERM וסיבוכיות המדגם של מחלקות סופיות

בשעה טובה נלמד על אלגוריתם הלמידה הראשון שלנו. האלגוריתם הנ"ל, הנקרא Empirical Risk Minimizer (ERM), הוא מאד כללי (לכל מחלקה $\mathcal{H}$ קיים אלגוריתם כנ"ל), מאד אינטואיטיבי וכמעט תמיד אופטימאלי. הסיבה שהוא לא פותר את כל הבעיות בלמידה היא שבד"כ הוא לא יעיל. על מנת לתת מוטיבציה לאלגוריתם נניח שקיבלנו מדגם

$$S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$$

נניח, כמו כן, שבידנו כוח חישוב בלתי מוגבל. נזכור שאנחנו רוצים להחזיר היפותזה h עם שגיאה הקרובה ככל האפשר לשגיאה של ההיפותזה הכי טובה ב־$\mathcal{H}$. דרך אחת לעשות את זה היא לעבור על כל ההיפותזות $h \in \mathcal{H}$ ולבחור את זו עם השגיאה, $L_{\mathcal{D}}(h)$, הכי קטנה. לצערנו לא ניתן לעשות זאת, אפילו אם יש בידנו כח חישוב בלתי מוגבל, מכיוון שאנחנו לא יודעים מהי ההתפלגות $\mathcal{D}$. עם זאת, המדגם שלנו מהווה ייצוג אמפירי של ההתפלגות $\mathcal{D}$. לכן, אנלוג טבעי להצעה הנ"ל היא לנסות להעריך, באמצעות המדגם, את $L_{\mathcal{D}}(h)$, ולהחזיר את ההיפותזה $h \in \mathcal{H}$ עם ההערכה הכי נמוכה. נגדיר, אם כן, את **השגיאה האמפירית** של h ביחס למדגם S להיות

$$L_S(h) = \frac{1}{m}\sum_{i=1}^{m} l(h(x_i), y_i)$$

אלגוריתם למידה הוא **ERM** אם הוא מחזיר $h \in \mathcal{H}$ המקיימת $L_S(h) = \inf_{h' \in \mathcal{H}} L_S(h')$.

הערה 5.1 1. נשים לב שלכל מחלקה קיים אלגוריתם ERM. עם זאת, הוא לא בהכרח יחיד - יכולה להיות יותר מהיפותזה אחת ב־$\mathcal{H}$ הממזערת את השגיאה האמפירית.

2. במקום לעבור על כל ההיפותזות ב־$\mathcal{H}$, היינו יכולים לעבור על אוסף גדול יותר. עם זאת, מכיוון שהאלגוריתם נדרש להתחרות רק בהיפותזות מ־$\mathcal{H}$, נראה שאין בזה צורך. יתרה מזאת, גישה כזו יכולה להזיק - אם נעבור על אוסף גדול יותר של פונקציות, השיערוך של השגיאה (כלומר, השגיאה האמפירית) יהיה פחות מדוייק.

המשפט הראשון שנוכיח יתן חסם עליון על סיבוכיות המדגם של אלגוריתמי ERM עבור מחלקות סופיות. כפועל יוצא אותו חסם חוסם את סיבוכיות המדגם של $\mathcal{H}$

משפט 5.2 נסמן $B = \sup_{y, y' \in Y} l(\hat{y}, y)$. לכל אלגוריתם ERM $\mathcal{A}$ מתקיים

$$\begin{aligned} m_{\mathcal{A}}(\epsilon, \delta) &\leq 2B^2 \cdot \frac{\log(|\mathcal{H}|) + \log\left(\frac{2}{\delta}\right)}{\epsilon^2} \\ m^r_{\mathcal{A}}(\epsilon, \delta) &\leq B \cdot \frac{\log(|\mathcal{H}|) + \log\left(\frac{1}{\delta}\right)}{\epsilon} \end{aligned}$$

רעיון ההוכחה וההוכחה עצמה פשוטים מאד - על מנת להוכיח שאלגוריתם ERM מחזיר היפותזה טובה, די להראות שאם מספר הדוגמאות במדגם S גדול מ־$m(\epsilon,\delta)$, אז בהסתברות לפחות $1-\delta$, לכל ההיפותזות ב־$\mathcal{H}$, השגיאה האמפירית מקרבת שאת השגיאה האמיתית עד כדי $\frac{\epsilon}{2}$. כלומר

$$\forall h\in\mathcal{H},\ \ |L_S(h)-L_{\mathcal{D}}(h)|\le\frac{\epsilon}{2} \tag{2}$$

אכן, במקרה הזה יתקיים

$$L_{\mathcal{D}}(\mathcal{A}(S))\le L_{\mathcal{S}}(\mathcal{A}(S))+\frac{\epsilon}{2}=\inf_{h'\in\mathcal{H}}L_S(h')+\frac{\epsilon}{2}\le\inf_{h'\in\mathcal{H}}L_{\mathcal{D}}(h')+\epsilon=L_{\mathcal{D}}(\mathcal{H})+\epsilon$$

האסטרטגיה של ההוכחה תהיה כדלקמן. ראשית, נראה שעבור $h\in\mathcal{H}$ בודדת, אי־שיוויון (2) מתקיים בהסתברות מאד גבוה. מכאן, ע"י חסם איחוד, נסיק שאי־השוויון מתקיים עבור כל ההיפותזות ב־$\mathcal{H}$.

אנו נשתמש בחסם הופדינג:

משפט 5.3 (הופדינג) יהיו $Z_1,\ldots,Z_m\in[0,B]$ מ"מ ב"ת. נסמן $\bar{Z}=\frac{1}{m}\sum_{i=1}^m Z_i$ אזי

$$\Pr\left(\left|\bar{Z}-E[\bar{Z}]\right|>\epsilon\right)\le 2\exp\left(-\frac{2\epsilon^2 m}{B^2}\right)$$

הוכחה: נוכיח רק את החסם עבור סיבוכיות המדגם במקרה הכללי (המקרה הפריד מושאר כתרגיל). יהא $\mathcal{A}$ אלגוריתם ERM ויהא

$$S=\{(x_1,y_1),\ldots,(x_m,y_m)\}$$

מדגם הנדגם לפי $\mathcal{D}$ כאשר $m\ge 2B^2\cdot\frac{\log(|\mathcal{H}|)+\log\left(\frac{2}{\delta}\right)}{\epsilon^2}$. כפי שהוסבר ברעיון ההוכחה, די להראות שבהסתברות $\le 1-\delta$ על פני בחירת המדגם, מתקיים שוויון (2) לכל $h\in\mathcal{H}$. נסמן ב־U_h את המאורע ששוויון (2) לא מתקיים. נסמן, כמו כן, ב־$U=\cup_{h\in\mathcal{H}}U_h$ את המאורע ששוויון (2) לא מתקיים עבור איזשהי פונקציה $h\in\mathcal{H}$. די להראות ש־$\Pr_S(U)<\delta$

נקבע $h\in\mathcal{H}$. מתקיים ש־

$$L_S(h)=\frac{1}{m}\sum_{i=1}^m l(h(x_i),y_i)$$

הוא סכום של משתנים מקרים ב"ת ש"ה עם תוחלת $L_{\mathcal{D}}(h)$ לכן, מאי־שוויון הופדינג, מתקיים

$$\Pr_S(U_h)=\Pr_S\left(|L_S(h)-L_{\mathcal{D}}(h)|>\frac{\epsilon}{2}\right)\le 2\exp\left(-\frac{\epsilon^2 m}{2B^2}\right)$$

מכאן, לפי חסם האיחוד, ומפני ש- $m \geq 2B^2 \cdot \frac{\log(|\mathcal{H}|)+\log\left(\frac{2}{\delta}\right)}{\epsilon^2}$, נובע

$$\begin{aligned}
\Pr_S(U) &= \Pr_S(\cup_{h\in\mathcal{H}} U_h) \\
&\leq \sum_{h\in\mathcal{H}} \Pr_S(U_h) \\
&\leq 2\sum_{h\in\mathcal{H}} \exp\left(-\frac{\epsilon^2 m}{2B^2}\right) \\
&\leq 2|\mathcal{H}|\exp\left(-\frac{\epsilon^2 m}{2B^2}\right) \\
&\leq 2|\mathcal{H}|\exp\left(-\frac{\epsilon^2\left(\frac{B}{\epsilon}\right)^2 \cdot 2\log\left(\frac{2|\mathcal{H}|}{\delta}\right)}{2B^2}\right) \\
&= 2|\mathcal{H}|\exp\left(-\log\left(\frac{2|\mathcal{H}|}{\delta}\right)\right) = \delta
\end{aligned}$$

■

6 מבנה הקורס

תורת ההכללה (שבוע 1-5). בשבועות הראשונים נפתח תורה שנקראית תורת ההכללה. תורה זו תאפשר לנו לחשב, עבור רוב המחלקות המעניינות אותנו, את סיבוכיות המדגם. אנו נתרכז בעיקר בבעיות קלסיפיקציה (כאלו שפונקצייה ההפסד היא l_{0-1}). בנוסף, בתרגילים, תכירו מספר בעיות קונקרטיות (זיהוי אובייקטים בתמונות וזיהוי ספאם), תבינו כיצד הקלטים (התמונות או ההודעות) מיוצגים, ותפעילו עליהם אלגוריתמי למידה מאוד מאוד בסיסיים.

אלגוריתמי למידה (שבוע 6-10). בחלק הזה נעבור על מספר מחלקות היפותזות ונלמד אלגוריתמים הלומדים את המחלקות הללו. כמו כן, נראה כיצד האלגוריתמים שנלמד עובדים על הבעיות הקונקרטיות הנ"ל.

ייצוג מידע (שבוע 11-12). בחלק הזה נתעמק יותר בנושא של ייצוג הקלטים. נלמד איך לייצג אותם בצורה שתהיה יותר קומפקטית, ותאפשר לאלגוריתמי למידה לעבוד טוב יותר.

נושאים נוספים (שבוע 13-14).