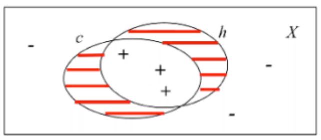
למידה חישובית ממידע: הרצאה 9 – תיאוריה

General Setting

- דגימות מגיעות מתוך $\Omega = (X, Y, P)$ כאשר X – הוא מרחב הווקטורים במימד הפיצ'רים (n), Y הם התצפיות, ה-labels, ה- target value. ו-P הינה ההתפלגות של הווקטורים והמטרות כך שההתפלגות המשותפת הינה ממימד הפיצ'רים n + 1 (כלומר n+1). אנחנו נשאף שההתפלגות הזו תהיה תלויה! אחרת אין לנו איך ללמוד. אבל הדגימות עצמן ב-training data הן אכן בלתי תלויות.
- אלגוריתם הלמידה L לוקח training data, מתוך $D \in \Omega^m$
- האלגוריתם למידה עובד עם סט היפותזות H ממרחב ההיפותזות.
- האלגוריתם מחזיר היפותזה/מודל $L(D) = h \in H$ (האינפוט הוא ה-training data והאינפוט הוא היפותזה)

- תאוריית **"אין ארוחות חינם"** תקפה כאשר אין לנו מרחב היפותזות ואז אי אפשר ללמוד.

הטעות האמתית / The True Error of h – נניח כי קיימת התפלגות מעל X. נגדיר: $TrueErr(h) = error_{\mathcal{D}}(h) = \text{Prob}_{X\sim\mathcal{D}}\,(c(X) \neq h(X))$
(כאשר D בנוסחה זו מייצג את ההתפלגות מעל X, שמסומן לעיל ב-P)



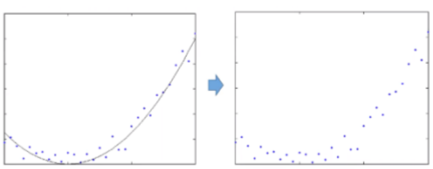
שיערוך סטטיסטי של טעות הסיווג / Statistical Estimation of the Classification Error

נשתמש ב-test set בגודל |S| = n, נניח כי ספרנו r טעויות. נשערך את טעות ההכללה על ידי: $\hat{p} = \frac{r}{n}$

מתאוריית סטטיסטיקת הדגימות נוכח להניח ב-95% **רווח סמך** – CI (Confidence Interval) עבור טעות ההכללה: $(\hat{p} - 2se, \hat{p} + 2se)$

כאשר ה-se הוא ה-standard error: $se = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$ ככל ש-n יותר גדול הטעות הסטנדרטית קטנה. כאשר הספרה 2 באינטרוול נובעת מהעובדה כי החלטנו על וודאות של 95% (אם היינו רוצים וודאות של 99% כנראה ספרה זו הייתה מגיעה ל-3~). וודאות זו משמעותה הסבירות שנקבל טעות מחוץ לטווח. חישוב טעות זו על training data, יקטן ככל שנרחיב את המורכבות של מודל.

Bias Variance Decomposition

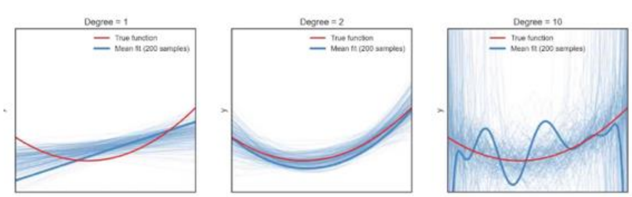


דוגמה ברגרסיה: הפונקציה (הקונספט c) היא סוג של פונקציה ריבועית וניקח דגימות עבור ה-training set עם רעש. הלמידה גם תניב פונקציה – h היפותזה. אנחנו חייבים להניח "עולם" על מרחב ההיפותזות מכיוון שאחרת נהיה ב-NFL ולא נוכל ללמוד את המודל. אם נניח "עולם לינארי" על מרחב ההיפותזות, לא משנה כמה דאטה יהיה לי, תמיד תהיה טעות. גם אם נעבוד על "עולם ריבועי" גם לא בהכרח נמצא את ה- perfect fit מפני שגם הדאטה שלנו לא בהכרח מייצג את העולם, אבל ככל שנגדיל את הדאטה, נתקרב יותר.

נאתחל את הניתוח / Analysis set-up: בהינתן המודל האמתי, c, נגריל ממנו דגימות עבור ה-training data, D, ונכניס אותן לאלגוריתם למידה L, מעל מרחב ההיפותזות H. האלגוריתם יניב מודל L(D) = h כך שניתן להגדיר אותה באופן הבא - $h(x|D) \in H$ (ההדגשה כאן היא ש-h היא פונקציה אשר תלויה ב-D). כדי להבין את כוח גישת הלמידה שלנו נרצה למצע (למצוא את התוחלת), עבור כל אינפוט ווקטור קבוע של פיצ'רים x,

ולחשב את: $E_D([h(x|D) - c(x)]^2)$. כאשר עבור כל D נקבל מספר אחר, ולכן מה שיש בתוך הריבוע הוא למעשה משתנה מקרי (משתנה מקרי פחות מספר, מהיות ש-c(x) אינו תלוי בדאטה, זהו הקונספט האמתי). לכן נרצה לחשב את התוחלת שלו, להעלות אותה בריבוע ועליה לחשב את הטעות. למעשה, **זוהי התוחלת של ההפרש בין מה שייצר האלגוריתם הלומד L עבור הדאטה, לבין המציאות.**

$y = h(x|D)$ (thin blue, individual 200 samples D)
$y = E(h(x|D))$ (thick blue, average od 200 samples D)
$y = c(x)$ (red)



H = linear functions
m = 10

H = quadratics
m = 10

H = polynomials of deg 10
m = 10

__בחזרה לדוגמה:__ הגרף האדום מייצג את הקונספט האמתי. אנחנו מתבוננים ב-200 training sets, 200 קווים דקים כחולים מייצגים את ההיפותזות שהאלגוריתם הלומד L הניב עבורם. הקו הכחול העבה מייצג את תוחלת (המיצוע של) ההיפותזות שלמדנו עבור 200 training datas.

- D הן training datasets בגודל m, שהגרלנו לפי התפלגות בלתי תלויה מעל אוכלוסיית הדאטה
- כדי לשערך את הביצועים של אלגוריתם הלמידה L שמפיק את h (x | D) כאאוטפוט עבור אינפוט D dataset, נרצה לשערך את :
- $E_D\left([h(x|D) - c(x)]^2\right)$ = זהו ה-**expected squared error** (הטעות הריבועית המצופה) בהינתן נקודה x. המקריות כן באה מה-training datasets השונים האפשריים מתוך $\Omega = (X,P)^m$. (נשים לב ש-ED מסמן תוחלת)

מושגים חשובים עד כה

- $E_D[h(x|D)]$ = the **expected** (or **mean**) **prediction value** at x = הקו הכחול העבה
- $E_D[h(x|D)] - c(x)$ = ההפרש בין הקו הכחול העבה לקו האדום בנקודה x = **ה-Bias**

  $Bias(x)$ = the expected prediction vs. true value, at x
- $E_D[(h(x|D) - E_D[h(x|D)])^2]$ = התוחלת של (משתנה מקרי פחות התוחלת שלו) בריבוע, היא ה**שונות** של הפרדיקציה ב-x. השונות מחושבת בנקודה מסוימת והיא מחשבת את המרחק בין קו כחול דק לקו הכחול העבה בנקודה x.

  $PredVar(x)$ = the prediction variance at x
- **Theorem (bias-variance decomposition)** = **משפט**: תוחלת הטעות בריבוע ב-x נתון מורכבת משני רכיבים: הביאס בריבוע והשונות.

$$E_D\left([h(x|D) - c(x)]^2\right) = (Bias(x))^2 + PredVar(x)$$

Decomposition for Squared Loss

For a given instance $x \in X$ we use the shorthand $h(x) = h(x|D)$.
All expectations are with respect to $\Omega = (X,P)^m$

$$(c(x) - h(x))^2$$
$$= \left(\underline{c(x) - E(h(x))} + \underline{E(h(x)) - h(x)}\right)^2 =$$
$$\left(c(x) - E(h(x))\right)^2 + \left(E(h(x)) - h(x)\right)^2$$
$$+\,2\left(c(x) - E(h(x))\right)\left(E(h(x)) - h(x)\right) =$$

$$E[(c(x) - h(x))]^2 =$$
$$\underbrace{\left(c(x) - E(h(x))\right)^2}_{\text{Bias}} + \underbrace{E\left[\left(E(h(x)) - h(x)\right)^2\right]}_{\text{Variance}}$$

© Zohar Yakhini and Ariel Shamir IDC

אם ניקח את ה-training data ונגביל אותו הרכיב שיפסיק להשתנות הוא ה-bias כאשר המודל שלנו הוא עדיין לא ב-perfect fit.

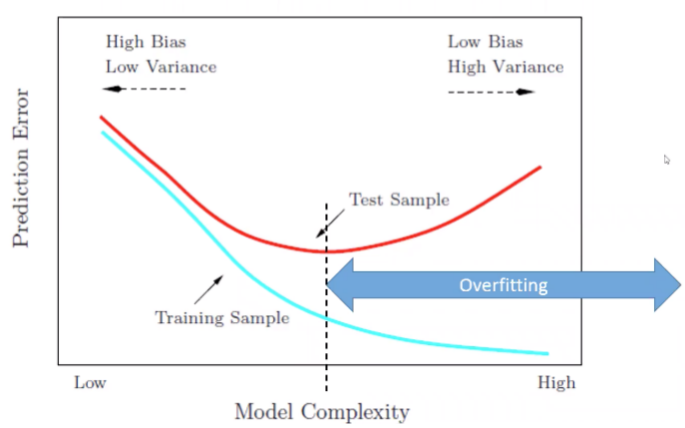
שורה שניה = שורה שלישית = פתיחת סוגריים עבור (a+b)^2

שורה רביעית = הסוגריים השמאליים הם מספר ובסוגריים הימניים h(x) הוא משתנה מקרי שתלוי ב-D, נזכור שהתוחלת עליו היא מספר. מכיוון שאנחנו מפעילים תוחלת ומלינאריות התוחלת (התוחלת תכנס לתוך הסוגריים ונקבל תוחלת על תוחלת, שזה פשוט E(h(x)) פחות התוחלת של h(x) וזה מניב 0, כפול סקלאר) השורה שמתחילה ב-2 תהיה 0.

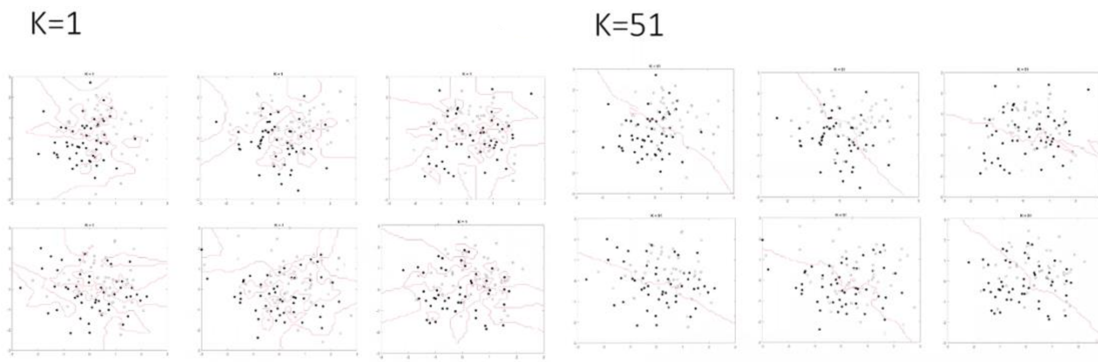
הוכחנו כי הטעות שאנחנו רוצים לשערך: $E_D([h(x|D) - c(x)]^2)$
ניתנת לפירוק לשני רכיבים ה-bias וה-variance!
ניתן לראות שה-bias תלוי באוסף ההיפותזות h וה-variance תלוי בדגימות.

- **הבייאס / Bias**: הטעות הטבועה באלגוריתם (ההגבלות על מרחב ההיפותזות)
- **השונות / Variance**: הטעות הטבועה בדאטה (האם ראינו מספיק דגימות דאטה?)

מורכבות המודל: למשל עבור למידה לינארית - אם אנחנו במודל פחות מורכב וה-bais גבוה, אם נוסיף עוד דגימות דאטה וה-bais לא ישתנה, נדע שנוכל להעלות את מורכבות לגבוהה יותר – נרצה להקטין את ה-bais ונצטרך להזהר משונות גבוהה מידי. פרקטית אנחנו לומדים מתוך ה-test את הטעות ומנסים להעריך את התוחלת על כל דגימה בו שאנו יודעים.



דוגמה של variance / bais עבור אלגוריתם KNN: עבור k-ים נמוכים ידוע שאנחנו בסכנת overfit, והמודל הוא בעל complexity גבוהה. עבור k ששווה למספר הדגימות (הגבוה ביותר) נקבל את המודל הפשוט ביותר (MLC). לכן **k גבוה** משמעותו model complexity נמוך, **ה-bias** שלו יהיה **גדול – הוא יטעה יותר** וה-**variance יהיה קטן** (ניתן לראות ש"גבול ההחלטה" עבור k=51 די דומה עבור training sets שונים, הוא די דומה).
ואילו עבור k נמוך נקבל מודל הרבה יותר מורכב ועשיר, ה-bias שלו קטן מפני שהוא טועה פחות ומדויק יותר, ואילו ה-variance שלו גבוהה מפני שניתן לראות באיורים של k=1 כי מתקבלות היפותזות שונות ומגוונות עבור training sets שונים.

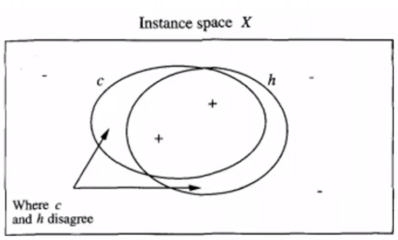


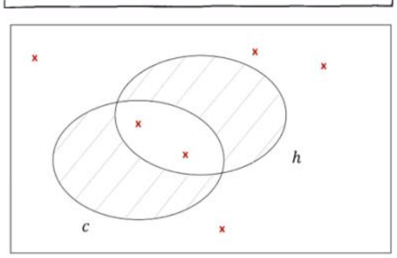
PAC Learning – Probably Approximately Correct Learning

1. Probability – מובטחת וודאות עם ביטחון של **1-delta** (1-δ certainty). ההסתברות למשל עבור 95% דלתא תהיה 5%
2. Approximation – הגבול רצוי, **epsilon**, על הטעות יהיה נקוב. האינטרוול
3. **נשערך את השימוש במשאבים**: גודל ה-training set (sample complexity) ו-זמן/מקום של למידה (הלמידה מתאפשרת בזמן פולינומיאלי עבור דגימת training)

<u>מרחבי היפותזות קונסיסטנטיים ביחס למרחב הקונספטים:</u> מרחב היפותזות H הוא קונסיסטנטי ביחס למרחב הקונספטים C אם C מוכל ב-H.
היפותזה h תקרא D-קונסיסטנטית ביחס לקונספט c **ול-training data D אם לכל נקודת דאטה d ב-D מתקיים h(d) = c(d).**
(תיאורטית אין טעויות על ה- training!)

<u>אלגוריתם למידה קונסיסטנטי:</u> אלגוריתם למידה L, שפועל על training data שמופקת על ידי קונספטים ממרחב הקונספטים C, ומשתמש במרחב היפותזות קונסיסטנטי H יקרא אלגוריתם למידה קונסיסטנטי אם לכל training data, D, ולכל קונספט c ממרחב הקונספטים C האאוטפוט L(D) (מניב היפותזה) הוא D-קונסיסטנטי ביחס ל-c, כלומר אם נסמן L(D) = h מתקיים h(d) = c(d) לכל נקודת דאטה d.
<u>הגדרה יותר כללית תהיה:</u> אלגוריתם למידה L שמשתמש במרחב היפותזות H ופועל על דאטה אימון שמופקת ע"י הקונספט c יקרא קונסיסטנטי, אם קיימת היפותזה שהיא D-קונסיסטנטית, אזי L(D) הוא אלגוריתם D-קונסיסטנטי.





If h is epsilon bad then $P(\text{err}) \geq \varepsilon$
To be the output of a consistent learning algorithm, all m training data points had to have avoided the blue region

סיבוכיות הדגימה: מרחב היפותזות סופי / Sample Complexity: Finite Hypothesis Space
היפותזה h תקרא epsilon-bad אם error(h) > epsilpon. אם נניח כי קונספט המטרה מוכל ב-H, אזי אלגוריתם למידה קונסיסטנטי חייב להניב היפותזה h קונסיסטנטית כלשהי, עבור כל m דגימות. השאלה היא מהי ההסתברות שכזו היפותזה h תהיה epsilon-bad? מהו ה-P של הדאטה אימון שמוביל ל-h כזו?

<u>החסם על P (ההסתברות ש-h הינה epsilon-bad):</u> נתבונן בהיפותזה h ממרחב ההיפותזות H, שהיא epsilon-bad. מהיות כל m הדגימות בלתי תלויות, ההסתברות (ב- $\Omega = (X, P)^m$) של היפותזה h שהיא epsilon-bad להיות קונסיסטנטית (שלא תהיה לה טעות על ה-training) עם כל m הדגימות היא קטנה שווה מ- m ^ (1 – epsilon) $\leq (1 - \varepsilon)^m$.
זה מתקיים עבור היפותזה אחת... המשך בעמוד הבא

נתבונן ב-m נקודות דאטה, $D \in X^m$, ההסתברות שקיימת היפותזה שהיא epsilon-bad ועדיין קונסיסטנטית ביחס ל-D :

$$\Pr(\exists h \text{ which is } \varepsilon\text{-}Bad \text{ and consistent}) \le \sum_{h \in \varepsilon\text{-}Bad} \Pr(h \text{ is consistent with } D_m) \le \left|\{h \text{ is } \varepsilon\text{-}Bad\}\right|(1-\varepsilon)^m \le |H|(1-\varepsilon)^m \le |H|e^{-\varepsilon m}$$

הערכת sample complexity במרחב היפותזות סופי : מה הסיכוי שקיימת היפותזה epsilon-bad שהיא קונסיסטנטית – קטן מהאיחוד של ההסתברויות שכל ההיפותזות שהן eps-bad כן קונסיסטנטיות עם הדאטה אימון בעל m הדגימות – קטן ממספר ההיפותזות שהן eps-bad כפול ההסתברות שהיפותזה eps-bad היא קונסיסטנטית. קטן מגודל מרחב ההיפותזות כפול ההסתברות ואי השוויון האחרון נובע מטור טיילור.

אם נדאג שהביטוי שהתקבל יהיה קטן מדלתא, נהיה "מסודרים".

- נשים לב שעליה לינארית במספר הדגימות מקטין את הסיכוי לטעות באופן אקספוננציאלי
- במטרה לצמצם את ההסתברות לכישלון להיות תחת רמת דלתא מסוימת נצטרך לדרוש :
- זהו אינו חסם הדוק! (בעיקר מכיוון שהחלפנו את המספר של מספר ההיפותזות שהן eps-bad בגודל של מרחב ההיפותזות H).

$$|H|e^{-\varepsilon m} \le \delta \ \text{ or}$$
$$m \ge \tfrac{1}{\varepsilon}\ln\tfrac{|H|}{\delta} = \tfrac{1}{\varepsilon}\left(\ln|H| + \ln\tfrac{1}{\delta}\right)$$

דוגמה: מרחב האינסטנסים X הוא ווקטורים בוליאניים n-ממדיים. גם מרחב ההיפותזות H וגם מרחב הקונספטים C מכילים צירופים של n ליטרליים (או המשתנה או שלילתו) בוליאניים מהצורה : $x_1 \vee \bar{x}_5 \vee \bar{x}_{22}$

<u>אין מגבלה נוספת על מרחב ההיפותזות</u> : לכל משתנה בוליאני Xi ההיפותזה שלנו יכולה להכיל את Xi או את המשלים שלו not-Xi או אף אחד מהם. אף אחד מהם אומר את הקונסטנט False עבור xi (הוא לא עוזר לנו).

נראה אם נוכל לבנות עבור מרחב זה לומד קינסיסטנטי. להלן הדאטה אימון : נתחיל עם הכל (start), למרות שזה לא במרחב שלנו, לאחר מכן נגיע לאינסטנס 1 וניקח את כל הליטרלים שמפריעים לו להלן מה שנשאר לאחר התיקון. וכן, נעבור לאינסטנס 2, איקס 2 לא יכול להופיע בצורתו הלא שלילית ולכן נוציא אותו וגם את נוט-איקס3. סיימנו בכך מפני שכבר נהיה קונסיסטנטים עבור אינסטנסים 3 ו-4.

1. Start: $x_1 \vee \overline{x_1} \vee x_2 \vee \overline{x_2} \vee x_3 \vee \overline{x_3} \vee x_4 \vee \overline{x_4}$
2. Instance 1: $x_1 \vee x_2 \vee \overline{x_3} \vee x_4$
3. Instance 2: $x_1 \vee x_4$

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1

- Consistent with Instances 3 & 4

<u>מכאן נוכל להסיק אלגוריתם כללי לדוגמה שלנו עבור מציאת אלגוריתם לומד קונסיסטנטי (=מסכים עם הדאטה אימון) :</u>

נתחיל עם : $\mathrm{h} = x_1 \vee \bar{x}_1 \vee x_2 \vee \bar{x}_2 \vee \ldots \vee x_n \vee \bar{x}_n$ (נשים לב ש-h=X כלומר הכי כללי) לכל דגימת דאטה עם ערך שלילי (0 בעמודת ה-y, c(x) = false) נסיר את כל הליטרליים המקיימים : $l_i \in \mathbf{\bar{x}}$ מההיפותזה ההתחלתית h. לכן יישאר לנו : $\mathrm{h} = l_{i_1} \vee l_{i_2} \vee l_{i_3} \vee \ldots \vee l_{i_k}$

וזהו קונסיסטנטי גם עם האינסטנסים החיוביים (1 בעמודת ה-y) מכיוון שכל הליטרליים שלהם (גם ליטרל ריק הוא תקין, כפי שנשים לב עבור הדוגמה לעיל שההיפותזה הסופית אינה מכילה את המשתנה X3 ובכל זאת ההיפותזה מסכימה עם הדאטה אימון באינסטנס הרביעי) הם בהכרח ב-h. ולכן נסיים עם h שהיא גם קונסיסטנטית עם דגימות הדאטה החיוביות

בהינתן m אינסטנסים ייקח לנו m*(גודל הממד, מס' הפיצ'רים) איטרציות ואם נניח שהמימד הוא בגודל קבוע הסיבוכיות תהיה O(m). אזי אם נוכיח שגם ה-sample complexity הוא לינארי, אנחנו נהיה עדיין באלגוריתם לינארי.

כמה אינסטנסים דרושים לנו?

- נניח כי יש לנו 10 אטריביוטים.
- בדוגמה שלנו, נשתמש בחיבור (משפט "וגם") של פיצ'רים (או חיבור/גימום ריק עבור כמה מה-i-ים) נקבל שגודל מרחב ההיפותזות הוא : $|H| = 3^{10} = 59{,}049$
- נרצה להבטיח בוודאות של 95% שההיפותזה שלנו תניב טעות שקטנה מ-10%. (הטעות על דגימות שלא ראינו)
- נצטרך לכך : $m > \frac{1}{0.1}(\ln 59049 + \ln \frac{1}{0.05}) = 10(11+3) = 140 \text{ instances}$ יותר מ-140 דגימות.
- נשים לב שהגודל של מרחב הדגימות X הוא 1024 (2^10).

- נניח כעת כי יש לנו 20 אטריביוטים.
- נקבל שגודל מרחב ההיפותזות כעת הינו :
- במקרה הזה, כדי לקבל וודאות של 95% שההיפותזה שלנו תניב טעות קטנה מ-10%, נצטרך :

$$m > \frac{1}{0.1}(\ln 3.5*10^9+\ln \frac{1}{0.05})=10(22+3) = 250 \text{ instances}$$

יותר מ-250 דגימות.

- ובמקרה זה יש לנו בערך 10^6 (בערך מיליון) אינסטנסים אפשריים (גודל מרחב הדגימות).

להלן ההגדרה הפורמלית של קונספט שהוא PAC Learnable

PAC Learnability

עבור מרחב קונספטים C, ומרחב דגימות X (כל דגימה היא ממימד n), ועבור אלגוריתם למידה L על מרחב היפותזות H, נאמר ש-C **הוא** PAC Learnable **ע"י** L **עם** H אם לכל $c \in C$, δ, ε, והתפלגות $(X, \mathcal{D})$, מתקיים:

1. מתוך דאטא שנדגם רנדומלית מתוך ההתפלגות $\mathcal{D}$, הפלט של L תהיה, בהסתברות שגדולה מ-$(1-\delta)$, היפותזה $h \in H$ כך ש- $error_{\mathcal{D}}(h) \leq \varepsilon$
2. זמן הריצה וכמות הדגימות של L היא פולינומיאלית ב-n, $\frac{1}{\varepsilon}$, $\frac{1}{\delta}$

על מנת להוכיח שמרחב קונספטים C הוא PAC Learnable בעזרת מרחב היפותזות H קונסיסטנטי ($C \subseteq H$):

1. נפעיל או נפתח אלגוריתם למידה קונסיסטנטי.
2. נבדוק שהחסם על מספר הדגימות (m) הוא פולינומיאלי ב-$\frac{1}{\varepsilon}$, $\frac{1}{\delta}$.
3. נוודא שכל צעד באלגוריתם שלנו הוא פוולינומיאלי - כלומר האלגוריתם כולו פולינומיאלי ב-n.

- Consider a class **C** of possible target concepts defined over a set of instances **X** of length (dimension) n, and a learning algorithm **L** using hypothesis space **H**.
- Definition

C is PAC-learnable by L using H
if for all $0 < \varepsilon < ½$, $0 < \delta < ½$, and for all $c \in C$ and distributions π over X, the following holds:

with data drawn independently according to π, L will output, with probability at least $(1-\delta)$, a hypothesis $h \in H$ such that $error_\pi(h) \leq \varepsilon$,
L operates in time and sample complexity that is polynomial in $1/\varepsilon$, $1/\delta$, n.

האם C = H = Disjunctions of Boolean Literals הוא PAC Learnable?

1. יש לנו אלגוריתם לומד (האלגוריתם שמוצג לעיל).
2. סיבוכיות הדגימה היא פולינומיאלית בכל הפרמטרים :

$$m \geq \frac{1}{\varepsilon}\ln\frac{|H|}{\delta} = \frac{1}{\varepsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right) = \frac{1}{\varepsilon}(\ln 3^n + \ln\frac{1}{\delta}) = \frac{n}{\varepsilon}\ln 3 + \frac{1}{\varepsilon}\ln\frac{1}{\delta}$$

ונקבל חסם על m להיות מספיק כדי להבטיח טעות epsilon עם הסתברות גדולה שווה מ- 1-delta

3. כל שלב בתהליך הלמידה הוא פולינומיאלי (בודקים דגימה אחת על ידי התחשבות ב-O(n) ליטרליים כדי לשמור על ביטוי קונסיסטנטי)

לכן, disjunction של n ליטרליים הוא PAC Learnable!

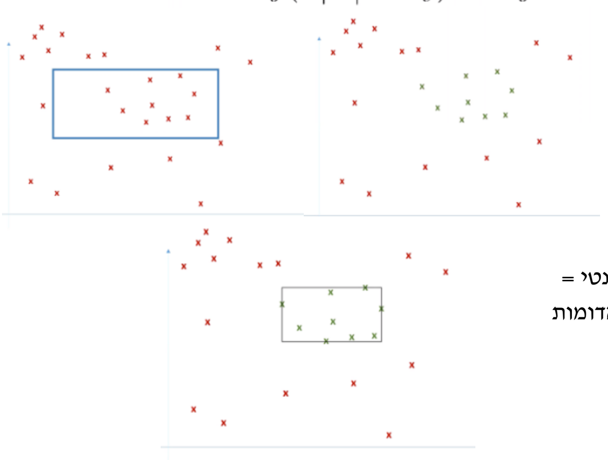
אבל, עבור קונספט = מרחב ההיפותזות שהוא כל הפונקציות הבוליאניות – לא נקבל קונספט שהוא PAC Learnable! להלן ההפרכה : (קיבלנו חסם דגימות אקספוננציאלי, ולא פולינומיאלי, כפי שההגדרה דורשת בסעיף 2)

- We have n Boolean features.
- Each instance in X is defined by any n Boolean values. Hence, $|X| = 2^n$
- A complete hypothesis space contains $|C| = |H| = 2^{|X|} = 2^{2^n}$ concepts.
- Assume that we have a consistent learner.
- If we now try to apply our simple bound we get:

$$m \geq \frac{1}{\varepsilon}\left(\ln|H| + \ln\frac{1}{\delta}\right) = 2^n \frac{1}{\varepsilon}\ln 2 + \frac{1}{\varepsilon}\ln\frac{1}{\delta}$$

בדרך כלל מרחבי ההיפותזות שלנו לא יהיו סופיים!

הכנה לדוגמה שתופיע בהרצאה 10: כל המלבנים ב-R^2 שהם מקבילים לצירים יהיה מרחב ההיפותזות שלנו. להלן קונספט מסוים (המלבן הכחול), כאשר ה-x-ים הם ה-training data שלנו. כאשר האיור שלא מכיל את הצירים מתאר את ה-training האמתי שבו לא רואים קונספט ספציפי אלא איזושהי קלסיפיקציה (הימני).



נרצה למצוא אלגוריתם למידה קונסיסטנטי : סביר להחזיר מלבן כך שהצלעות יקחו את האיקס הכי קטן והאיקס הכי גדול, (וימקמו 2 צלעות מקבילות לציר ה-y) ואת ה-y הכי קטן והכי גדול שימקמו 2 צלעות מקבילות לאורך ציר ה-x. ונטען שזהו אלגוריתם קונסיסטנטי = מתאים עם הקונספט על ה-training בוודאות מכיוון שבטוח בתוך המלבן לא יהיו נקודות אדומות מכיוון שמראש הנחנו שהנקודות הירוקות הן בתוך מלבן.