1/4/20 - ג'

הרצאה #2

מבוא -

מהן המטרות שלנו בקורס הזה? למידה נמצאת בכל מקום - הרצון לנבא תחומי העדפות ספציפיים למשתמש לפי המידע שיש לגביו, או זיהוי קול, פנים, רכישת כישורים, נהיגה אוטונומית...
רשתות עמוקות אף יכולות ללמוד ע"י אימונים מול עצמן במקום מול בן-אנוש.

למה צריך ML? ישנן משימות מורכבות מדי מכדי שנוכל לבנות מע' כזו, בגלל תכנות קלאסיים. כמו כן, ישנן משימות הדורשות תגובה דינאמית לאירועים שמשתנים במהירות ואותנו לא נוכל להתכונן/לתכנן איך נטפל בהם. בנוסף ישנה משימה של עדכון שוטף של המע' תוך קבלת קלטים.

מהי למידה? ננסה לבנות מע' שלומדת בעצמה. לשם כך נצטרך לענות על השאלות:
- כיצד לומדים?
- מהם הדברים שנחוצים ללמידה?
- איך יודעים אם הצלחנו ללמוד?

שאפשרית, מבוססת על תהליך הלמידה בבני אדם

★ הגדרה: להתחיל לפענח את העולם מסביב, להפריד בין אינפורמציה מועילה ללא מועילה (לדוג': הפרדת חפץ ורקע. או ההברות אותן אנו שומעים מול רעשי רקע) למשל "יצוג יעיל" של המידע, לתפוס מושגים מופשטים, לארגן מידע לכדי ידע...

★ הגדרה: (אותה נלמד בקורס): נרצה להבין מ קב' דוגמאות את החוק, פונק' $f: X \to Y$
X נקרא ה- domain set והוא מכיל את הדוגמאות, Y נקראת ה- label set (קב' התגיות שניתן לשייך לאובייקט)
הלמידה מתבססת על מופעים $(x,y) \in X \times Y$, ומטרתה להפיק מהן קירוב לפונק' f, $\hat{f}$ (נקראת prediction rule) ואומדנה ע"י מדידת המרחק בין $f(x)$ ל- $\hat{f}(x)$ על $x \in X$.
למידה זו נקראת למידה מפוקחת (Supervised learning).

דוגמאות:

Domain Sets: Feature Vectors, Images, Time Series (סדרה שמשתנה לאורך זמן/משתנה), Geo Data, Compound (כמו טקסט ותמונה)

Label Sets: Classification $= \{\pm 1\}$, Multiclass Classification $= \{1, .., K\}$,
Regression $= \mathbb{R}$, $\underbrace{\text{Ranking, Matching (התאמה), Link prediction}}_{\text{לא נעסוק באלו בקורס הזה}}$ (ניבוי קשרים חברתיים)

משימות:
- זיהוי הונאות אשראי: X - features של טרנזקציות, Y - {OK, fraud} → קלסיפיקציה
- סינון דואר זבל: X - אימיילים (free text), Y - {OK, Spam} ← קלסיפיקציה
- קלסיפיקציה של אובייקטים בתמונה: X - תמונות, Y - מחלקות אובייקט, |Y| די קטן → Multiclass classification
- חיזוי מחירי נדל"ן - X - features של בית, Y - מחיר המכירה ← regression

תופעת אין ארוחות חינם – נניח כי ה- training data שנו הוא $\{x_i, f(x_i)\}_{i=1}^m$,
וכי f יכולה להיות כל פונקציה. אזי ה- training data לא מספק
לנו שום מידע עבור $x\in X$ שלא נמצא ב- training data.
גם ללא מידע מקדים על f, זה בלתי אפשרי ללמוד אותה.

← בגלל המסקנה הזו, נהיה חייבים להניח משהו על f בקורס הזה. למשל: נניח
ש-f שייכת למשפחת פונקציות (שידועה לנו) כלשהי $\mathcal{H}$, ולכן אלגוריתם הלמידה יפלוט
תמיד איזושהי $h\in\mathcal{H}$. $\mathcal{H}$ נקראת מחלקת ההיפותזות או מודל.

סוגים של למידה:

- Batch learning: 1. בוחרים את מחלקת ההיפותזות $\mathcal{H}$ מראש.
  2. אוספים training data: $\{(x_i,y_i)\}_{i=1}^m$
  3. מאמנים את המודל, ומקבלים $h\in\mathcal{H}$
  4. עבור דאטה חדש: $\{\tilde{x}_j\}_{j=1}^K$ נחזה: $\{h(\tilde{x}_j)=:\hat{y}_j\}_{j=1}^K$

-או-

- Online learning: 1. בוחרים מחלקת היפותזות $\mathcal{H}$ ומתחילים מהיפותזה ראשונה h_0
  2. מבצעים בלולאה $i=1,2,\dots$:
     (i) מקבלים דוגמה x_i
     (ii) משערכים את ה- label שלה: $\hat{y}_i=h_{i-1}(x_i)$.
     (iii) מקבלים פידבק מהעולם האמיתי: הערך האמיתי y_i.
     (iv) על סמך השוואה בין $\hat{y}_i, y_i$ מעדכנים את הפונק' (לקבלת): h_i.

- Supervised: ה-training set מכיל דגימות ו- lables $(\{(x_i,y_i)\}_{i=1}^m)$

או

- un-Supervised: ה-training set אינו מכיל lables, רק $\{x_i\in X\}_{i=1}^m$. ← [דוגמה: חלוקה ל-clusters או מה שמתאים לפי התקופה.]

← עסוק ב: supervised learning+batch learning: כאשר: $X=\mathbb{R}^d$
(אלו features vectors), $Y=\{\pm1\}$ (בעיית קלסיפיקציה) או $y=\mathbb{R}$ (בעיית רגרסיה).
נזכיר גם $Y=[K]$ (מולטי-קלסיפיקציה) ו- unSupervised learning.

לאן נשאף להגיע? להכיר את התאוריה, האלגוריתמים, העקרונות וגישות חישוביות (פרקטיות).
נכיר data-sets שונים, ונלמד איך לחקור data-sets חדשים, בין היתר
כדי להתמודד עם ערכים חסרים או שגויים. נכיר משימות ML, ומודלים של ML.
נלמד להרגיש אילו גישות יעבדו על איזה בעיה (ע"י ניסיון...) ואיך לשפר את
טיב הפתרון שלנו. נשתמש בקודים, סימולציות יחד עם כלים מתמטיים מתחומי תאוריות..

רגרסיה ליניארית

בעיה לדוג' - אני עובדת בחנות מקוונת ורוצה לשער את ה- lifetime value של לקוח מסוים (כמה אני צופה להרוויח ממנו לאורך חייו). נתמקד ב- batch learning:

ה- training data $=:S$ יהיה רישומי הקניות של m לקוחות קיימים: feature vector באורך d שמתאר ערכים בפרמטרים כגון - גיל, הכנסה, מיקום... נסמן: $X=\mathbb{R}^d$, $Y=\mathbb{R}$.

ונתבונן ב- $x\in X$. זו בעיית רגרסיה.

נארגן את המידע שיש לנו בצורה מטריצונית:

$$X=\begin{bmatrix} \top & & \top \\ x_1 & \cdots & x_m \\ \bot & & \bot \end{bmatrix},\quad Y=\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

(כלו': $X[i,j]$ זה הפיצ'ר ה-j של לקוח מס' i) [נסמן: z_i - וקטור, z^i - וקטור z במקום i]

הנחה: אין רעש. כלומר קיימת $f:X\to Y$ שבהינתן x מחזירה במדויק את ה-label שלו.

המטרה: ללמוד את הפונק' f מ- Y, X, כך שנוכל לחזות את ה- lifetime של $x\in X$.

★ בגלל ש"אין ארוחות חינם" נניח שיש מחלקת היפותזות $\mathcal{H}$, ונעביר את הלמידה עבור $\hat{h}\in\mathcal{H}$ בלבד.

① המקרה ה- noisless: (התווית עבור x תהיה במדויק $f(x)$, ללא רעש נוסף)

- נניח שנגלה שהיחס בין x ל-y הוא ליניארי (כלומר f ליניארית..) אז ניקח את מחלקת ההיפותזות הבאה:

$$\mathcal{H}_{lin}=\left\{(x_1,\dots,x_d)\mapsto w_0+\sum_{i=1}^{d}x_iw_i \;\middle|\; w_0,w_1,\dots,w_d\in\mathbb{R}\right\}$$

ונחפש $h\in\mathcal{H}_{lin}$. נבחין שכל $h\in\mathcal{H}_{lin}$ מאופיינת ע"י המשקלות $w_1,\dots,w_d$ (חישוב מ"פ בין הוקטור x לוקטור המשקלות) והחותך (intercept) w_0. לכן המשמעות של למידת פונק' כזו היא למידת המשקלות והחותך.

∘ נפשט: - את $x=(x^1,\dots,x^d)\in X\in\mathbb{R}^d$ נסמן - $x=(1,x^1,\dots,x^d)^T\in\mathbb{R}^{d+1}$

- נסמן $w=(w^0,w^1,\dots,w^d)^T\in\mathbb{R}^{d+1}$

⇐ עקב הסימונים הללו מתקיים: $\mathcal{H}_{lin}=\{x\mapsto x^Tw \mid w\in\mathbb{R}^{d+1}\}$

ולכן:

$$X=\begin{bmatrix} 1 & \cdots & 1 \\ \top & & \top \\ x_1 & \cdots & x_m \\ \bot & & \bot \end{bmatrix}\in\mathbb{R}_{(d+1)\times m}$$

∘ נניח שאכן קיימת f, המקיימת: $f(x)=y$.

אנחנו מוכנים למצוא את $\hat{f}\in\mathcal{H}_{lin}$: (לעיתים נסמנה h_S להדגיש שהיא תלויה ב- S sample)

① המקרה ה- Realizable: המקרה שבו $f\in\mathcal{H}_{lin}$, ולכן יש לנו תקווה לקבל את f עצמה באלג' הלמידה. במקרה זה: $S=\{(x_i,y_i)\}_{i=1}^{m}$ מקיימת: $y_i=x_i^Tw$ ל- $w\in\mathbb{R}^{d+1}$

כשהשאלה שאנו רוצים לגלות, כלו' מתקיים: $X^Tw=y$ ונרצה למצוא את W.

במילים אחרות במקרה ה-Realizable למע' הליניארית $y=X^Tw$ יש לפחות פתרון אחד. (כי f אכן ליניארית ☺)

★ למעשה זה לא מצב פיזיקלי

② המקרה ה- Non-Realizable: במקרה זה f לא ליניארית, ולא נוכל לגלות אותה במדויק כי המע' שלנו מחזירה רק פונק' ליניאריות, אלא נשאף לקבל קירוב שלה. למע' $X^Tw=y$ "יתכן שאין פתרון כלל.

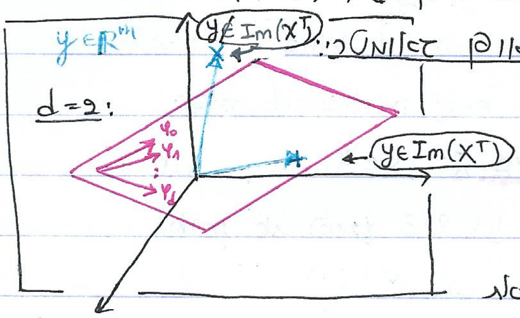
נסתכל על הבעיה מנק' מבט גאומטרית: נסמן: $X = \begin{bmatrix} \leftarrow \varphi_0 \rightarrow \\ \vdots \\ \leftarrow \varphi_d \rightarrow \end{bmatrix}$, $\varphi_i \in \mathbb{R}^m$.

• במקרה ה-Realizable: ל- $X^Tw = y$ יש פתרון (יחיד או ∞), ולכן $y \in Im(X^T)$.

• במקרה ה-Non-Realizable: ל- $X^Tw=y$ אין פתרון, ולכן לא בהכרח ש-$y\in Im(X^T)$.

★ אם $m \geq d+1$ מתקיים ש: $Im(X^d) \subseteq \mathbb{R}^m$, כמו כן: $Im(X^T) = Sp\{\varphi_0,..,\varphi_d\}$

ולכן ניתן לפרש את שייכות y ל- $Im(X^T)$ באופן גאומטרי:



★ נפרט את המקרים:

1) Realizable, פתרון יחיד: יש פתרון ולכן $X^Tw=y$

אז y הוא קומבינציה לינארית של $\varphi_0,..,\varphi_d$.

בגלל שהפתרון יחיד, אזי יש רק דרך יחידה לכתוב את הקומבינציה הזו, ולכן $\varphi_0,..,\varphi_d$ בת"ל, הם פורשים את המרחב של $\mathbb{R}^m$: $Im(X^T)$,

ולכן הם מהווים בסיס עבורו. אזי יש $f\in H_{lin}$ יחידה.

2) Realizable, ∞ פתרונות: לפי הגדרה יש ∞ פתרונות ולכן יש תלות לינארית בין עמודות X^T, כלומר בין $\varphi_0,..,\varphi_d$, ולכן יש ∞ דרכים שונות לכתוב את y באמצעות $\varphi_0,..,\varphi_d$.

כל פתרון כזה נותן $f\in H_{lin}$ שונה שפותרת את הבעיה.

⇦ נמצא את אלג' הלמידה: יחזיר $h_S \in H_{lin}$.

• במקרה ה-Realizable: נרצה להחזיר את $f\in H_{lin}$.

• במקרה ה-non-Realizable: נרצה להחזיר $h\in H_{lin}$ "קרובה" ל- $f\in H_{lin}$.

אם אנחנו לא יודעים באיזה מקרה אנחנו מבוש. לכן נשתמש בפונק' loss:

$L(\cdot,\cdot)$ שמודדת את עלות הטעות $L(y,h_S(x))$, והיא משמשת אותנו להערכת h_S.

• לדוגמא - Absolute Value Loss: $L(y,h_S(x)) = |y-h_S(x)|$

Squared Loss $L(y,h_S(x)) = (y-h_S(x))^2$

כ"א טובה מסיבות שונות ולמטרות שונות. נשתמש ב- SL כי יותר קל לחשב פונק' חלקה.

★ מטרתנו שנשתמש בפונק' loss להעריך את יכולת $h_S(x)$ שלנו, הרעיון לחפש אחד h_S שממזערת את ה- SL של S שלנו. כלומר נחשב את $\sum_{i=1}^m L(y_i,h_S(x_i))$

(ה- loss של h_S שנקבל אם הוא מתאימה). הערך הזה נקרא הסיכון האמפירי.

לדוגמא: במקרה שלנו: h_S היא פונק' לינארית: $x \to x^Tw$, ופונק' ה- loss היא SL.

לכן הסיכון האמפירי: $\sum_{i=1}^m (y_i - x_i^Tw)^2 = \|y-X^Tw\|^2 = (y-X^Tw)^T(y-X^Tw)$.

במקרה זה, מציאת מינימום הסיכון האמפירי שקולה למציאת המינימום של סכום ריבועי ההפרשים בין התוויות y_i לבין הפונק' הלינארית $h_S(x_i)$: $y_i - x_i^Tw_i$.

ההפרש הזה נקרא ה-i'th residual, והסיכון האמפירי במקרה שלנו נקרא Residual sum of Squares (RSS).

⇦ אם כן $h\in H_{lin}$, נייצג כ $w\in\mathbb{R}^{d+1}$ ונחשב עבורו את RSS(w), לבסוף ניקח את הפרמ' שממזער את RSS(w). כלומר נרצה את $\underset{w}{argmin}(RSS(w))$.

(ERM) Empirical Risk minimization

"זה נעשה תהליך של"

הסיכון האמפירי, במקרה הזה ה- RSS : $\underbrace{\|y-X^Tw\|}_{w\mapsto}$

נבחין שפונק' המחיר הנ"ל היא חלקה, והיא תבנית ריבועית במשתנה w. כמו כן אם w מדרגה מלאה, אז יש לפונק' נק' מינימום יחידה. נסמן:

$w_S = \arg\min \|y-X^Tw\|$, ונקבל: $h_S(x)=x^Tw_S$

נבחין שבזה עוסק ל2 המקרים:

(1) במקרה ה- non-Realizable: נקבל את הפונק' שמביאה ה- RSS מינימלי, וזה מה שרצינו.

(2) במקרה ה- Realizable: ה- RSS של הפונק' האמיתית f יהיה 0, ולכן יהיה מינימלי, ו-f תמצא כפי שרצינו.

אז איך נמצא את המינימום? כי למעשה בעיית מינימיזציה של RSS !

הפונק' גזירה ∞ פעמים אז קיימות לה נגזרות חלקיות. תנאי הכרחי ל-w שיהיה ה-minimizer הוא שכל הנגזרות החלקיות של RSS יתאפסו:

לכל $j\in\{0,..,d\}$: $\frac{\partial}{\partial w^j} RSS(w) = -2\sum (x_i)^j (y^i - x_i^T w) = 0$

וזו משוואה של $d+1$ משוואות לינאריות במשתנה w, ולכן הצגה שקולה היא:

$\nabla RSS(w) = -2X(y-X^Tw)=0$.

בגלל שזו תבנית ריבועית, יש מינימום יחיד שהוא הפתרון למערך המשוואות הזו.

לכן פתרון w קיים: $X(y-X^Tw)=0 \Leftarrow Xy = XX^Tw \Leftarrow (\langle\varphi_j, y-X^Tw\rangle=0 \quad \forall j\in\{0,..,d\})$

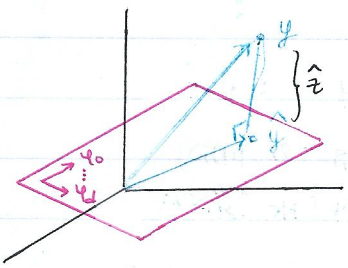
משוואות מהצורה הזו נקראות המשוואות הנורמליות. למה?

★ איך יראה פתרון כזה בצורה גאומטרית: מכך שפתרון $\hat w$ כזה יקיים: $\langle\varphi_j, y-X^T\hat w\rangle=0$, $j\in\{0,..,d\}$, ו- $Span\{\varphi_0,..,\varphi_d\}=Im(X^T)$ מתקיים שעבור $\hat w$: $y-X^T\hat w \perp Im(X^T)$.

נסמן: $\hat y = X^T\hat w \in Im(X^T)$ ונגדיר את וקטור השארית $\hat z = y-\hat y \in Im(X^T)^\perp$

כלומר $\hat y$ הוא ההטלה האורתוגונלית של y על $Im(X^T)$.

$\Leftarrow$ ולכן קוראים למשוואות המשוואות הנורמליות - כדי לפתור אותן נטיל את y על $Im(X^T)$ הטלה אורתוגונלית.



נבחין שמשפט פיתגורס מתקיים: $\|y\|^2 = \|\hat y\|^2 + \|\hat z\|^2$.

איך נמצא את הפתרון $\hat w$? נזכיר שאנו מניחים כי $m\geq d+1$.

ואנחנו רוצים למצוא את $\hat w$ שמקיים: $Xy = XX^T\hat w$

נחלק ל-2 מקרים:

(1) עמודות X^T בת"ל: נבחין שעמודות X^T בת"ל $\overset{\text{לינארית}}{\Longleftrightarrow} \dim(Ker(X^T))=0 \overset{\text{ש.מ}}{\Longleftrightarrow} \dim(Ker(X\cdot X^T))=0$, ולכן $X\cdot X^T$ הפיכה, ונקבל: $\hat w = (XX^T)^{-1}Xy$

ויש לנו אלגוריתם למידה! (נחשב את $(XX^T)^{-1}$ ונכפול $(XX^T)^{-1}Xy$ למציאת $\hat w$).

(יתכן ש-y כבר בתת המרחב)

<u>המשמעות הגאומטרית:</u> יש רק דרך יחידה לתאר את ההטלה האורתו' של y, $\hat{y}$ על $Im(X^T)$ כקומבינציה לינארית של $\{\varphi_0,...,\varphi_d\}$ בסיס ל-$Im(X^T)$. ולכן $\hat{y}\in Im(X^T)$.

(2) עמודות X^T תלויות: אז ישנם ∞ פתרונות למע' המשוואות, ולכן יש ∞ דרכים לתאר את ההטלה של y על $Im(X^T)$ (שוב, כמו במקרה ה-Realizable או non-Realizable, מדובר על ההטלה של y או על y עצמו) כקומבינציה לינארית של איברי הבסיס $Im(X^T)$. אבל ההטלה נשארת אותה ההטלה ①.

<u>איך נפתור את המע' הנ"ל, במחשב?</u>

- אלימינציית גאוס? לא. לא מדויק כשעושה ע"י מחשב.
- פירוק SVD! נפרק את: $X\in\mathbb{R}_{(d+1)\times m}$ כך: $X=U\cdot\Sigma\cdot V^T$

<u>כאשר:</u> $V\in\mathbb{R}_{m\times m}$, $U\in\mathbb{R}_{(d+1)\times(d+1)}$ מטריצות אורתונורמליות, $\Sigma\in\mathbb{R}_{(d+1)\times m}$ מטריצה אלכסונית. נסמן את איברי אלכסון Σ ב: $\sigma_1\geq...\geq\sigma_{d+1}\geq 0$ הם נקראים הערכים הסינגולריים של X. וקטורי עמודות U קרויים הוקטורים הסינגולריים השמאליים של X, (והם הוקטורים העצמיים של XX^T) וקטורי עמודות V קרויים הוקטורים הסינגולריים הימניים של X (והם הוקטורים העצמיים של X^TX). נבחין שריבוע הערכים הסינגולריים: $\sigma_1^2,...,\sigma_{d+1}^2$ הם ערכים עצמיים הן ל-X^TX, והן ל-XX^T.

★ פירוק SVD אינו יחיד. (למשל ע"י החלפת סדר הע"ע)

★ טענה: $\dim(Ker(X^T))=0 \iff \sigma_{d+1}>0$.

ולכן נוח לראות האם למע' לינארית יש פתרון יחיד.

נשתמש ב-SVD כדי לפתור את מע' המשוואות הלינארית:

<u>נגדיר</u> את: $\Sigma^+\in\mathbb{R}_{(d+1)\times m}$ ע"י:

$$\Sigma_{i,i}^+=\begin{cases}\frac{1}{\sigma_i} & :\sigma_i>0\\ 0 & :\sigma_i=0\end{cases}$$

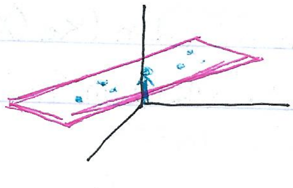
ואז: $w_s=U\Sigma^+V^Ty$.

נבחין ש-w_s מהווה פתרון למע' הלינארית:

(1) עמודות X^T בת"ל: אז XX^T הפיכה ולכן: $w=(XX^T)^{-1}Xy$

(2) עמודות X^T תלויות: w_s $\overbrace{\text{עדיין פתרון}}^{\text{(ויש } \infty \text{ פתרונות)}}$, ולמעשה הוא הפתרון עם הנורמה המינימלית. $\|\hat{w}\|=\min\{\|w\|_2 \mid X^Tw=y\}$ - וזה טוב כי נוח לעבוד איתו.

כך נקבל את הפתרון הכי קרוב לראשית -



למה נשתמש ב-SVD בתור אלגו' הלמידה המוצע ברגרסיה לינארית?

- עובד במקרים ה-Realizable וה-non-Realizable. (הטלה / ללא הטלה)
- עובד במקרים שבהם למע' יש פתרון יחיד או ∞ פתרונות. ($\{\psi_i\}$ ב"ת או תלויים)
- יציבות נומרית: מחשבים מחשבים באמצעות ביטים, יש להם דיוק סופי. לכן קשה לממש במכונה את התאורייה המתמטית, ונצטרך ללמוד איך אלגו' של למידה ממומשים, ומהם המגבלות שלהם.

יציבות נומרית מתייחסת לציון מידת דיוק האלגו' ואת שגיאות הקירוב המוערכות שלו.

- אי-יציבות נומרית נוצרת ממגבלות ייצוג מספרים במחשב, המוגבל ע"י גודל הזכרון שלו וגורם לכך שהמספר מעוגל או מקוצץ. מגבלת הייצוג עלולה לגרום לשגיאות בחישוב והן עלולות לשנות את תוצאות החישוב בצורה ניכרת. מצב זה נקרא אי-יציבות נומרית.

איך הופכים את SVD ליציב אף יותר?

לעתים המטריצה XX^T הפיכה, אך למעשה קרובה להיות סינגולרית (כלומר יש כמעט תלות לינארית בין אחד או יותר מעמודות X^T). אז חלק מהערכים הסינגולריים יהיו >0, אך מאוד קטנים, ואלגו' שאינם יציבים נומרית ישגו.

במקרה כזה במקום להשתמש ב: Σ^+ (ע"י לקיחת $\sigma_i>0$ והפיכה ל-$\frac{1}{\sigma_i}$)

נשתמש ב: $\Sigma^{+,\varepsilon}$: $\Sigma^{+,\varepsilon}_{i,i} = \begin{cases} \frac{1}{\sigma_i} & :\sigma_i>\varepsilon \\ 0 & :\sigma_i\le\varepsilon \end{cases}$, וכך נשיג יציבות נומרית.

מהו ה-Sample Size הדרוש ללמידת מע' משוואות לינארית?

כאשר מע' המשוואות $X^T w = y$ מכילה m משוואות ב-$(d+1)$ נעלמים.
לכן אנו חייבים $m \ge d+1$. (כי אם $m<d+1$ אין לנו מספיק דאטה!)
ככל ש-d גדול יותר, מחלקת ההיפותזות גדולה יותר ומורכבת יותר, (נקבעת ע"י וקטור w שיש לו יותר קואורדינטות)
ולכן נצטרך יותר דוגמאות כדי ללמוד אותה.

— רגרסיה לינארית- noiseless case: סיכום בשקף 42 —

② המקרה ה-noisy:

התווית עבור x לא תהיה בדיוק $f(x)$ אלא $f(x)+\varepsilon$ עבור רעש ε (רנדומי) שנובע מטעויות מדידה או גורמים אחרים שנכנסים לחשבון.

עד עכשיו הנחנו ש-$S=\{(x_i, f(x_i))\}_{i=1}^m$. כעת נניח ש: $\{(x_i, f(x_i)+z_i)\}_{i=1}^m$

עבור: $z_i \overset{iid}{\sim} N(0,\sigma^2)$. כלו' התפלגות ע' תוחלת 0 ושונות σ^2.

מה אנחנו רוצים ללמוד?

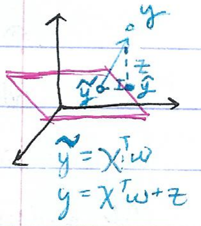
(1) במקרה ה-Realizable: $f \in H_{lin}$. נניח שיש וקטור w בלתי-ידוע

כך ש: $y_i = x^T w + z_i$ (כי קיבלנו תוויות אחרי הוספת הרעש).

נסמן את וקטור הרעש: $Z = (z^1, .., z^m)^T$, ולכן בכתיב מטריצוני:

$y = X^T w + Z$. נבחין שבמקרה כזה, כבר לא נוכל להגיד בוודאות

ש: $y \in Im(X^T)$ ולכן למצוא $y = X^T w$ לא יהיו פתרונות כלל.



אך עדיין נוכל להשתמש בשיטה שבה השתמשנו מקודם: ERM

$w_S := \min\arg_w \|y - X^T w\|$.

אז נפתור את מערך המשוואות הליניאריות הנ"ל, ונלמד את $h_S \in H_{lin}$

ובכך, כפי שעשינו מקודם, למעשה נטיל את y על $Im(X^T)$ (נסמן $\hat{y}$)

ועל ההטלה נמצא את ערך הריבועים הפחותים (השגיאה), שאותו ננסה למזער.

☆ במקרה הזה ניתן להגיע למערך המשוואות ע"י שימוש בעקרון אחר שאינו

נגזר, ולא רק דרך ERM: נראות מירבית (Maximum Likelihood)

- נניח שהרעש הוא גאוסייני (מתפלג נורמלית): $z_i \overset{iid}{\sim} N(0, \sigma^2)$
  אז $y_i \overset{iid}{\sim} N(x_i^T w, \sigma^2)$ (התוחלת - הממוצע - הוא $0 + x_i^T w$).
- נניח שאנחנו יודעים את וקטור המשקולות w.
- מה ההסתברות לחזות ב-lable שלנו? (בהינתן ש-X ו-w ידועים):

$$P(y|w) = \prod_{i=1}^{m} \left[\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(- \frac{(x_i^T w - y_i)^2}{2\sigma^2} \right) \right]$$

(כי y_i מ"מ גאוסייני עם $\mu = x_i^T w$, $\sigma = \sigma^2$ והוא הצפיפות השולית, (בת"ל זה בזה)

אז ההסתברות לראות את y כולו זה מכפלת הצפיפויות השוליות של y_i.)

- בבעית למידה זה הפוך - יודעים את y ורוצים את w.

אז מה הערך הסביר ביותר ל-w, בהינתן X, y:

עקרון הנראות המירבית אומר נבחר את ה-w מבין כל האפשריים

עבורו צפיפות ההסתברות שנקבל את y שאותו אכן קיבלנו, היא המירבית.

אז נכתוב את פונק' הנראות (likelihood function) כפונק' של w

עם X, y קבועים:
$$L(w|y) = \frac{1}{(2\pi\sigma^2)^{m/2}} \prod_{i=1}^{m} \exp\left(- \frac{(x_i^T w - y)^2}{2\sigma^2} \right)$$

נגדיר את Maximum Likelihood Estimator (MLE) עבור w:

$$\hat{w} = \arg\max_w L(w|y) \underset{\text{log מונוטונית}}{=} \arg\max_w \log L(w|y) \underset{\substack{\text{עד קבוע, בגלל המינוס} \\ \text{זה הופך למצוא argmin} \\ \text{וקבוע המכפלה לא חשוב.}}}{=} \arg\min_w \sum_{i=1}^{m} (x_i^T w - y_i)^2$$

⇓

ריבועים פחותים!

Polynomial fitting - מקרה פרטי של רגרסיה ליניארית.

דוג': בהינתן נק' $a_1,\dots,a_m\in\mathbb{R}$ ותוויות $y_1,\dots,y_m\in\mathbb{R}$

נרצה ללמוד פולינום $p:\mathbb{R}\to\mathbb{R}$ שעובר דרך הנק' $\{(a_i,y_i)\}$: $\forall i\in[m]: p(a_i)=y_i$

מחלקת ההיפותזות: $\mathcal{H}^d=\{a\mapsto \sum_{k=0}^{d} w_k a^k\}$

ונגדיר: $X_i=(1,a_i,a_i^2,\dots,a_i^d)$.

לכן אפשר לכתוב את מחלקת ההיפותזות גם כך: $\mathcal{H}^d=\{x\mapsto x^T w \mid w\in\mathbb{R}^{d+1}\}$

כאשר: $X=\begin{bmatrix} | & & | \\ x_1 & \cdots & x_m \\ | & & | \end{bmatrix}\in\mathbb{R}^{(d+1)\times m}$, והיא מדרגה מלאה.

כשמתאימים $\underbrace{\text{פולינום}}_{\text{ממעלה שנבחר d}}$ (לעבור דרך הנק' $\underbrace{\text{הללו}}_{\text{שהגדרנו}}$) משתמשים ברגרסיה ליניארית,

אם ה- training set: $i\in[30]: y_i=p(a_i), x_i=(1,a_i^1,\dots,a_i^d)$

הרגרסיה הליניארית מחזירה וקטור מקדמים $\hat{w}$ (אומגה האט), מדרגה d.

★ נבחין שככל ש-d גדל, $\mathcal{H}^d$ הופכת להיות עשירה (ומסובכת) יותר, אבל גם בעלת יכולת לקרב את הפולינום האמיתי בצורה טובה יותר.

★ עבור d'ים שונים נראה את הקירוב שקיבל לפונק' האמיתית.

ל-d קטן, $\mathcal{H}^d$ קטנה מדי ולא יכולה לתאר את הפולינום האמיתי.

נגדיר bias - עד כמה הפונק' האמיתית ניתנת לתיאור בתוך מחלקת ההיפותזות שלנו. כמו שאמרנו, ככל שמחלקת ההיפותזות גדולה יותר ה- bias קטן.

נוסיף רעש גאוסייני: $z_1,\dots,z_{30}\overset{iid}{\sim}\mathcal{N}(0,4)$. ה- training set הוא: $S=(x_i=(1,a_i^1,\dots,a_i^d), y_i=p(a_i)+z_i)\ i\in[30]$.

ואנחנו מנסים למצוא פולינום מדרגה d: $\hat{p}$ שעובר דרך $\{(x_i,y_i)\}_{i=1}^{30}$.

נבחין שמכלל שהרעש רנדומי, S משתנה כל הזמן ולכן $\hat{p}$ משתנה כל הזמן.

בג"ל שכאשר התוויות y_i רנדומיות (בגלל z_i הרכיב הרנדומי), כלל ההחלטה h_S (prediction rule) גם מקרי, ולכן יש לו שונות.

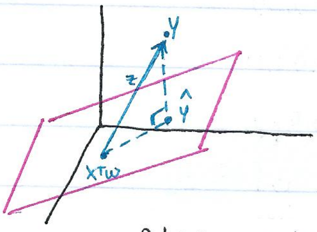
מה משפיע על השונות, ואיך? ככל שרמת הרעש גדולה יותר - תהיה יותר שונות בין המדגמים S ולכן יש יותר מקום לשונות בין כללי ההחלטה המתאימים למדגמים הללו. כמו כן, גם גודל $\mathcal{H}^d$ מחלקת ההיפותזות מכריע: ככל שיש לנו יותר אפשרויות להחזרת h_S, נמצא כלל החלטה ספציפי לכל S והוא מאוד תלוי בה. לכן עבור S מאוד שונים - h_S'ים יהיו מאוד שונים.

<u>תיאור גיאומטרי של השונות</u>: $y_i = x_i^T w + z_i$. כדי לשערך את w

אנחנו משליכים את y על $Im(X^T)$ לקבלת $\hat{y}$.

מפני שהרעש נוצר iid - לכאורה הוא מוסיף שונות בכל הכיוונים.

(הפיזור שלו שווה בכל הכיוונים).



האלגו' הלומד למעשה מבצע הטלה אורתוגונלית: $\hat{y}$.

הטלה למעשה "מנקה" את הרעש: נשארים רק עם

הרכיב של הרעש שמקביל לתת המרחב $Im(X^T)$.

<u>אז</u>: ככל ש-$Im(X^T)$ <u>גדולה יותר ב-$\mathbb{R}^m$</u>, אנחנו נתפוס יותר מהרעש

הזה, <u>והשונות</u> תהיה <u>גדולה יותר</u> כתוצאה מכך.

אז המקרה שבו נקבל שלרעש יש השפעה מינימלית יהיה כאשר <u>$m = d+1$</u>

(כי עדיין צריך שנוכל ללמוד משהו וזה רק אם $m \geq d+1$).

כמו כן כאשר <u>מגדילים את ה-training set</u> (גודלו m), אנו בהכרח <u>מקטינים את השונות</u>.

⇐ רוצים ש-$m \gg d+1$ וזה ע"י הגדלת m או הקטנת מימדו $Im(X^T)$, כדי להקטין את השונות.

<u>ה-Tradeoff בין bias ל-Variance</u>

- כאשר מחלקת ההיפותזות קטנה: בד"כ יהיה bias גבוה ושונות נמוכה. → נקרא גם סיבוכיות מודל נמוכה

(bias גבוה - קשה לתאר את הפונק' האמיתית כאשר $\mathcal{H}^d$ מוגבלת,

שונות נמוכה - האלגו' ימצא הפונק' $\mathcal{H}^d$ הטובה ביותר לצרכיו

אך $\mathcal{H}^d$ מוגבלת ולכן הפונ' הנ"ל תלויה יותר, פחות מושפעת מהרעש).

- כאשר מחלקת ההיפותזות קטנה: בד"כ יהיה bias נמוך ושונות גבוהה. → נקרא גם סיבוכיות מודל גבוהה

☆ וזה מתבטא בקטנוני m לעומת d+1...