Principal Components Analysis (PCA)

נוריד את הדאטה ל-k מימדים כך שנישאר עם Var מקסימלי. הוקטורים העצמיים מאונכים זה לזה. נקודה $x_j \in \mathbb{R}^d$ בהטלה ל-k מימדים תיוצג ע"י:
$$\sum_{i=1}^k \alpha_{ji} e_i$$
$$error = \left\| x_j - \sum_{i=1}^k \alpha_{ji} e_i \right\|$$
המטרה: מזער את $\sum_{j=1}^n \left\| x_j - \sum_{i=1}^k \alpha_{ji} e_i \right\|$.

1. חשב תוחלת $\hat{\mu} = \frac{1}{n}\sum_{i=1}^n x_i$, נגדיר $z_i = x_i - \mu$.
2. חשב את $S = \sum_{j=1}^n z_j z_j^T$ ומצא k וקטורים עצמיים עם הערכים העצמיים הגדולים ביותר
3. נוריד מימד ע"י $Y = E^T Z$ ($E = [e_1, \dots, e_k]$)

האלגוריתם הוא unsupervised, לא מתייחס ל-labels. שחזור ע"י כפל ב-E^T (והוספת μ)

Logistic Regression

$y(x) = sign\left(w^T x + w_0\right)$ נרצה לגרום לפונקציה להיות יותר חלקה וגזירה, לכן נשתמש ב-sigmoid: $\sigma(z) = \frac{1}{1+e^{-z}}$. יש חשיבות למרחק של הנקודה מקו ההפרדה. למציאת ה-w האופטימלי נמזער את פונקציית ההפסד עם gradient descent, לכל $j \in [1,d]$:
$$w_j^{t+1} = w_j^t - \eta \sum_i x_i^j \left(\mathbb{P}(c=1 \mid x_i; w) - y_i\right)$$
$$\ell(w) = \sum_{i=1}^n \ln\left(1 + e^{-y_i a^t x_i}\right) = \frac{1}{n}\sum_{i=1}^n y_i \log h(x_i) + (1-y_i)\log(1-h(x_i))$$

Linear discriminant analysis (LDA)

נמצא הטלה ל**קו** שתשמר את הפרדה (עדיין יש צורך לסווג לאחר מכן) נרצה למקסם את מרחק תוחלות ההטלה ולמזער את השונויות, נמקסם את $J(v) = \frac{(\widetilde{\mu_1} - \widetilde{\mu_2})^2}{\widetilde{S_1}^2 + \widetilde{S_2}^2}$.
לא מובטחת הפרדה מוחלטת, אבל נקבל הפרדה טובה ביותר. v שממקסם את J מקיים $S_B v = \lambda S_W v$ כאשר $S_W = S_1 + S_2$.
$$S_j = \sum_{x_i \in c_j} (x_i - \mu_j)(x_i - \mu_j)^T$$
$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$
אם S_W הפיכה נמצא ישירות: $v = S_W^{-1}(\mu_1 - \mu_2)$.
לעומת PCA, משנה רק הכיוון של הוקטור שמצאנו ולא מיקומו במרחב. LDA ממירה מכל מספר פיצ'רים למימד אחד. האלגוריתם הוא supervised. MDA - הכללה עבור $c \geq 2$ מחלקות, ניתן להוריד עד $c-1$ מימדים, נסווג למחלקה עם ערך הכי גבוה.

ε-Representitives

קבוצת אימון S היא ε-ייצוגית אם לכל היפותזה h: $|L_S(h) - L_D(h)| \leq \varepsilon$. כלומר, S מייצגת את המציאות עד כדי ε.
$\mathcal{H}$ מתכנסת באופן אחיד אם בהסתברות לפחות $1-\delta$ מדגם $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ יהיה ε-מייצג לכל היפותזה h.
אם $\mathcal{H}$ מקיימת תכונה זו אז היא למידה ב-Agnostic PAC עם סיבוכיות דגימות $m_{\mathcal{H}}(\varepsilon,\delta) \leq m_{\mathcal{H}}^{UC}\left(\frac{\varepsilon}{2}, \delta\right)$, וגם $ERM_{\mathcal{H}}$ יעבוד טוב עבורה.

k-nearest neighbors (KNN)

לא פרמטרי, ודיסקרימינטיבי. ב-KNN נשערך את ה-posterior. נקבע ערך k ונמצא V שיכיל בתוכו k נקודות. יכול להתכנס לאותה תוצאה כמו PW. KNN לא תחשב פונקציית צפיפות, וקו ההחלטה לא גזיר.
$$p(c_i \mid x) = \frac{k_i}{k}; \quad p(x, c_i) \approx \frac{k_i/n}{V}$$
עבור 0-1 loss, כלומר MAP, המחלקה עם ה-k_i הגדול ביותר תיבחר. סיבוכיות גדולה מאוד, $\mathcal{O}(ndk)$ ומספר הפרמטרים גדל יחד עם הקלט.

- k קטן מדי יגרום ל-decision boundaries קופצניים ולא מאוזנים (Var גבוהה)
- k גדול מדי יגרום לגבולות החלטה חלקים מדי (bias גבוה)

להסתכל על היחס בין הצירים בגרף! בנוסף, אין תהליך אימון.

Naive Bayes

מניחים מאפיינים ב"ת בהינתן y. $\mathbb{P}(c_i \mid X) = \frac{\mathbb{P}(c_i)\times\mathbb{P}(X \mid c_i)}{\mathbb{P}(X)}$ אם באמת ב"ת, אופטימלי:
$$\mathbb{P}(w_j) = \frac{|w_j|}{|D|}; \quad c^* = \arg\max_{c_i} \mathbb{P}(c_i) \times \prod_j \mathbb{P}(X_j \mid c_i)$$
laplace smoothing - כאשר במחלקה כלשהי לא מופיעה דגימה נחליק את הנתונים:
$$\mathbb{P}(X = x \mid w_k) = \frac{\#x \in w_k + 1}{|w_k| + d}; \quad d = \#\text{פיצ'רים}$$
Bayesian Estimation: θ הוא RV עם התפלגות משלו.
$$\mathbb{P}(X \mid D) = \int \mathbb{P}(X \mid \theta) \times \mathbb{P}(\theta \mid D)\, d\theta; \quad \mathbb{P}(\theta \mid D) = \frac{\mathbb{P}(D \mid \theta) \times \mathbb{P}(\theta)}{\int \mathbb{P}(D \mid \theta) \times \mathbb{P}(\theta)\, d\theta}$$
כאשר $n \to \infty$ Bayesian Estimation ישאף ל-MLE. לא תמיד בייס יותר טוב מ-MLE, תלוי ב-prior ובמספר הדגימות.

Parzen Windows

לא פרמטרי, וגנרטיבי. ב-Parzen Windows נשערך את ה-likelihood. נסמן
$$p(x) = p(x \mid c_i) \approx \frac{k/n}{V}$$
ב-PW קובעים את החלון, קוביה d-מימדית עם צלע h: $V = h^d$. נבדוק האם נקודה x נמצאת בחלון (region) נשתמש בפונקציית האינדיקטור:
$$\Phi\left(\frac{x_i - x}{h}\right) = \begin{cases} 1, & \frac{|x_{ik} - x_k|}{h} \leq \frac{1}{2}, k \in \{1, \dots, d\} \\ 0, & else \end{cases}$$
$$p(x) \approx \frac{k/n_j}{V} = \frac{1}{n}\sum_{i=1}^{n_j} \frac{1}{h^d}\Phi\left(\frac{x_i - x}{h}\right); \quad k = \sum_{i=1}^n \Phi\left(\frac{x - x_i}{h}\right)$$
נדרוש $\Phi \geq 0$ וגם $\int_{\mathbb{R}} \Phi(u)\, du = 1$. נרצה לבחור h שיגרום לצפיפות להיות חלקה.

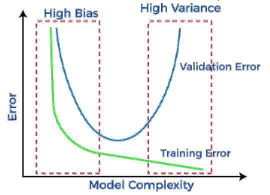
- h קטן מדי יגרום לפונקציה להיות מאוד קופצנית, כל דגימה תשפיע קצת מדי
- h גדול מדי יגרום לפונקציה להיות חלקה מדי, כל דגימה תשפיע יותר מדי

נסווג את המחלקה שממקסמת את $p(x)$. קשה להעריך את ה-h האופטימלי.
נוכל להחליק את הצפיפות ע"י φ גאוסיאנית, וכך הצפיפות הסופית תהיה ממוצע גאוסיאנים (שמתפלגים $\mathcal{N}(0,1)$).

מסווג MLE: החיזוי הוא $\arg\max_{c_i} \mathbb{P}(x \mid c_i)$
מסווג MAP: החיזוי הוא $\arg\max_{c_i} \mathbb{P}(x \mid c_i) \cdot \mathbb{P}(c_i)$
ה-prior הוא $\mathbb{P}(c_j)$, ה-likelihood היא $\mathbb{P}(x \mid c_j)$
וה-posterior הוא $\mathbb{P}(c_j \mid x)$ (מזה אכפת לנו)
סיכון מותנה: $R(a_i) = \sum_{j=1}^m \mathbb{P}(c_j \mid x) \times \lambda(a_i \mid c_j)$ סיכון כולל: $R(a_i) = \int R(a(x) \mid x) \times \mathbb{P}(x)\, dx$ ב-bayesian decision rule נרצה למזער את הסיכון המותנה עבור 0-1 loss. כלל החלטה ביאסייני שקול ל-MAP (אם אין priors אז MLE)
$$\mathbb{P}(err \mid \ell) = 1 - \mathbb{P}(pred \mid \ell); \quad \mathbb{P}(err) = \int_{-\infty}^{\infty} \mathbb{P}(err \mid x) \times \mathbb{P}(x)$$
ב-MLE נשערך את הפרמטר θ שיקרב באופן הטוב ביותר את הדגימות להתפלגות. מניחים כי המאפיינים ב"ת. גוזרים ומשווים $\nabla = 0$.
משערך $\hat{\theta} = \arg\max \ln \mathbb{P}(x_1, \dots, x_n \mid \theta)$ הוא unbiased אם $\mathbb{E}(\hat{\theta}) = \theta$. בנוסף, נשאף לשונות נמוכה. $Uni(0,\theta) : \hat{\theta} = \max_i X_i$.
$$\mathcal{N}(\mu, \sigma) : \hat{\mu} = \frac{1}{n}\sum_{i=1}^n X_i, \hat{\sigma} = \frac{1}{n}\sum_{i=1}^n (X_i - \hat{\mu})^2$$

Error Decomposition

השגיאה $\min_{h \in \mathcal{H}} L_D(h)$ תלויה רק ב-$\mathcal{H}$. כאשר היא קטנה ומניחים עליה יותר - bias גבוה - underfitting. אם מגדילים את סיבוכיות $\mathcal{H}$ שגיאת האימון תקטן, החל ממקום מסוים השגיאה בפועל תעלה - variance גבוה - overfitting. שגיאות: optimization - ממודל לא מאומן למודל מאומן למודל הטוב ביותר במחלקה - estimation/generalization. מהטוב ביותר למחלקה לאופטימום - modeling.



Perceptron

סיווג לא נכון כאשר $a^t y_i < 0$. כל עוד קיימת דוגמא שמסווגת לא נכון, עבור על כל דוגמא ועדכן במידה ומסווגת לא נכון: $a^{k+1} = a^k - \eta^k X_i$.
אם הדאטה לא ניתן להפרדה ליניארית נרצה $\eta^k \to 0$ כאשר $k \to \infty$. למשל, $\eta^k = \frac{\eta^1}{k}$. אין לנו הבטחה שהתהליך יעצור במקום טוב במידה והדאטה לא ניתן להפרדה ליניארית. batch rule: בכל פעם נעדכן עם כל הדוגמאות שמסווגות שגויה, $a^{k+1} = a^k - \eta^k \nabla J(a_k)$. כך התהליך חלק יותר, לעומת ה-single sample rule/stochastic בו יש הרבה קפיצות. הפסד: $J(a) = \sum_{i=1}^n \max\{0, -a^t y_i X_i\}$.

Mean Squared Error (MSE)

מאתחלים את הוקטור b (לרוב באחדות), נוסיף מימד אחד לכל דוגמא עם הערך 1, ונהפוך את הערכים לכל דוגמא במחלקה השנייה. אם Y הפיכה נחשב את $a = bY^{-1}$. נסה לחשב את $a = \left(Y^t Y\right)^{-1} Y^t b$. אם נכשלנו נעדכן בכל פעם:
$$J(a) = \sum_{i=1}^n \left(a^t y_i - b_i\right)^2; \quad a^{k+1} = a^k - \eta^k Y^t\left(Y a^k - b\right)$$
אם $\eta^k = \eta^1$ אז a^k יתכנס ל-a כך ש-$Y^t b = Y^t Y a$. לעומת פרספטרון, תמיד מתכנס אך רגיש ל-outliers.

Clustering

למידה unsupervised, למזער מרחק בתוך cluster ולמקסם בין cluster־ים. פונקציית דמיון $s(x_i, x_k)$ גדלה ככל ש־x_i ו־x_k דומים. מרחק d להפך. d פופולריות: נורמה־2 (אוקלידי), נורמה־1 (מנהטן) או נורמה־∞ (צ'בישב). פונקציית דמיון cosine: $s(x_i, x_j) = \frac{x_i^T x_j}{\|x_i\| \cdot \|x_j\|}$. אדישה ל־scaling (אכפת רק מהזווית). דרישות על פונקציית מרחק: סימטריות, $d(a, a) = 0$ ואי־שוויון המשולש.
לא טוב לנרמל את הדאטה. נעדיף מספר דגימות זהה בכל cluster. כשהקבוצה לא סגורה וחסומה נתקשה להפריד. $J_{SSE} = \sum_{i=1}^{k} \sum_{x \in D_i} \|x - \mu_i\|^2$. SSE עובד כאשר הדגימות ניתנות לעטיפה ע"י אליפסות כאלה, כשה־cluster־ים ניתנים להפרדה.
ב־k-means ננסה למצוא מספר cluster־ים שימזערו את J_{SSE}. נבחר k ו־k מרכזים (אפשר להגריל). כל עוד החלוקה משתנה:

- עבור כל כל דגימה ושייך אותה ל־cluster עם המרכז הכי קרוב
- עבור כל cluster נחשב את הממוצע והפוך אותו למרכז החדש

השגיאה קטנה בכל שלב, נשים לב שניתן להיתקע או לתת תוצאות שגויות כתוצאה מאתחול לא אופטימלי של המרכזים.

Decision Trees

לא פרמטרי, צמצום האנטרופיה אם נבחר בפיצ'ר A:

$$Gain(A) = i(N) - \sum_{v \in Val(A)} \frac{|S_v|}{|S|} \cdot i(N_v)$$

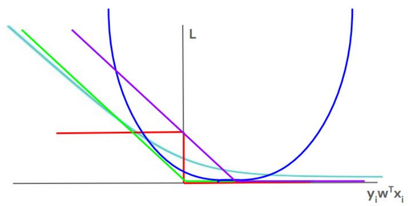
מניעת overfitting ע"י עצירה לפני העץ המלא ואפשור שגיאות אימון. בנוסף, ניתן לגדל עד הסוף ואז להעיף חלק מהצמתים (pruning). העץ הטוב ביותר נבחר ע"י tradeoff בין גודל העץ ושגיאת האימון. ב־pruning נוריד צמתים שתורמים מעט, באופן חמדן החל מאבות העלים. עבור כל צומת:

- הורד את הצומת ותת העץ שלה, החלף בעלה עם class נפוץ ביותר
- נלך ל־test ונבדוק את השגיאה, נשאיר רק אם הוא טוב מאחרים

נסיים כאשר הגענו למצב בו כמעט ואין עוד שיפורים, ונבחר את העץ עם השגיאה הקטנה ביותר (כמה עצים עם אותה שגיאה ־ ניקח את הקטן ביותר).

Ensemble Methods

הרבה מסווגים פשוטים, כל אחד עם שגיאה קטנה מ־$1/2$ על אותה קבוצת אימון. נוכל ליצור קבוצות זרות, להגריל דוגמאות לכל מסווג או להשתמש בכל הדאטה אבל עם אתחול משקלים שונה.
ב־Bagging, בהינתן דאטה לאימון S, נדגום דאטה לכל מסווג ונאמן, בהסקה ניקח את המחלקה הנפוצה בקרב המסווגים (ברגרסיה אפשר ממוצע). טוב כאשר המסווג רגיש לרעש, משפר ביצועים וחלק יותר. ב־Random Forests נעשה Bagging על הדאטה וגם על הפיצ'רים. בבניית כל עץ נגריל את המאפיינים שיוכל להשתמש בהם ונבנה את העץ באמצעותם. עדיף מ־Bagging על עצי החלטה ־ מפחית תלות בין עצים שונים.



0-1 loss ־ אדום
preceptron ־ ירוק
MSE ־ כחול
SVM ־ סגול
logistic ־ תכלת

Probably Approximately Correct (PAC)

נניח שקיים מסווג מושלם f, נגדיר את השגיאה בתור:

$$L_{D,f}(h) = \mathbb{P}_{x \sim D}[h(x) \neq f(x)]$$

נניח כי הדגימות i.i.d וכי $y_i = f(x_i)$ לכל x_i, וכך $L_{D,f}(f) = 0$.
לא נוכל להבטיח ש־$L_{D,f}(h) = 0$, או אפילו $L_{D,f}(h) \leq \varepsilon$. משפט No Free Lunch:
יהיו $\varepsilon < 1/2$, $0 < \delta < 1$, אזי לכל אלג' למידה A וקבוצת אימון בגודל m קיים עולם D, f כך ש־$\mathbb{P}\left(L_{D,f}(A(S)) \geq \varepsilon\right) \geq \delta$. אם אין מידע נוסף, לא נוכל לעשות משהו טוב יותר מאשר ניחוש. לא ידוע שום דבר על $\mathcal{H}$ ־ ה־bias הוא 0 והשונות ענקית.
עבור מחלקת היפותזות סופית $\mathcal{H}$, $ERM_{\mathcal{H}}(S)$ ימזער את:

$$L_S(h) = \frac{1}{m} |\{i \mid h(x_i) \neq y_i\}|.$$

כפלט נקבל היפותזה כלשהי, אם $f \in \mathcal{H}$ אז נקבל את f: $L_S(f) = 0$. $L_S(h)$ היא שגיאת אימון ו־$L_{D,f}(h)$ היא השגיאה בפועל. אם $\mathcal{H}$ מחלקת היפותזות לינארית אז $\mathbb{E}_{S|x}[L_S(h)] = L_{D,f}(h)$.

מחלקת היפותזות $\mathcal{H}$ היא למידה ב־PAC אם קיימת פונקציה $m_{\mathcal{H}} : (0,1) \to \mathbb{N}$ ואלג' למידה כך שלכל $0 < \delta < 1, D, f : \mathcal{X} \to \{0,1\}$, אם נאמן את האלג' על $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ דוגמאות מאותה התפלגות ו־i.i.d נקבל היפותזה h כך ש־$\mathbb{P}\left(L_{D,f}(h) \leq \varepsilon\right) \geq 1 - \delta$. $m_{\mathcal{H}}$ היא סיבוכיות הדגימות של $\mathcal{H}$. אם $\mathcal{H}$ סופית, אז $\mathcal{H}$ למיד ב־PAC עם סיבוכיות דגימות של $m_{\mathcal{H}}(\varepsilon, \delta) \leq \frac{\log(\mathcal{H}/\delta)}{\varepsilon}$ אם נשתמש ב־$ERM_{\mathcal{H}}$.

ב־Agnostic PAC לא מניחים שקיים מסווג מושלם או שהוא ב־$\mathcal{H}$. מחלקה למידה ב־Agnostic PAC ביחס לקבוצה $Z = \mathcal{X} \times \mathcal{Y}$ ופונקציית הפסד $\ell : \mathcal{H} \times Z \to \mathbb{R}^+$ אם קיימת פונקציה $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ ואלג' A כך שלכל $0 < \varepsilon, \delta < 1, m \geq m_{\mathcal{H}}(\varepsilon, \delta)$:

$$D^m\left(\left\{S \in Z^m \mid L_D(A(S)) \leq \min_{h \in \mathcal{H}} L_D(h) + \varepsilon\right\}\right) \geq 1 - \delta$$
$$L_D(h) = \mathbb{P}_{(x,y) \sim D}(h(x) \neq y) = \mathbb{E}_{z \sim D}[\ell(h, z)]$$

אם $\mathcal{H}$ סופית, פונקציית הפסד בתחום $[0,1]$ אז היא למידה ב־Agnostic PAC עם $ERM_{\mathcal{H}}$, עם סיבוכיות דגימה של $m_{\mathcal{H}}(\varepsilon, \delta) \leq \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\varepsilon^2} \right\rceil$. עבור $\mathcal{H}$ אינסופית $m_C(\varepsilon, \delta) \in \mathcal{O}\left(\frac{1}{\varepsilon^2}\left(VCdim(\mathcal{H}) + \ln\left(\frac{1}{\delta}\right)\right)\right)$.

Linear Regression

בניגוד ל־classification, נחזה ערך רציף $y = w^T x + b$.

$$J = \sum_{i=1}^{n} \left(y_i - \left(w^T x_i + b\right)\right)$$

נגזור ונקבל:

$$\sum_{i=1}^{n} x_i \left(y_i - w^T x_i - b\right) \quad ; \sum_{i=1}^{n} \left(y_i - w^T x_i - b\right)$$

אם המשתנים תלויים, מימד גבוה מדי או מעט מדי דוגמאות אימון נקבל משקלים עצומים. נפתור עם רגולריזציה: ridge ו־lasso. ב־ridge נגדיר:
$J(W) = \sum_{i=1}^{n} \left(y_i - b - w^T x\right)^2 + \lambda \|w\|_2^2, \lambda \geq 0$
λ גדול ימנע מהמשקלים להתפוצץ. בנוסף, $w_{ridge} = \left(X^T X + \lambda I\right)^{-1} X^T y$ וכך $X^T X$ הפיכה. lasso רגישה יותר ל־outliers מאשר ridge.
ב־ridge המקדמים הפחות חשובים יקטנו אך עדיין יהיו בעלי משקל כלשהו, ב־lasso הם יתאפסו לגמרי. ל־ridge נוסחא סגורה ול־lasso אין. $J_{lasso}(w) = \sum_{i=1}^{n} \left(y_i - b - w^T x\right)^2 + \lambda \|w\|_1$.

VC-dimension

עבור מחלקת היפותזות $\mathcal{H}$, ה־VCdim שלה הוא גודל הקבוצה הגדולה ביותר של דגימות שיכולה לקבל את כל התוויות האפשריות ע"י מסווגים ב־$\mathcal{H}$. אם נרצה להוכיח ש־$VCdim(\mathcal{H}) = d$ נראה כי:

- קיימת קבוצה בגודל d ש־$\mathcal{H}$ מנתצת
- כל קבוצה בגודל $d+1$ לא ניתנת לניתוץ ע"י $\mathcal{H}$

ככל שהממימד קטן כך המחלקה פחות מסובכת וקל יותר ללמוד. עבור $\mathcal{H}$ סופית מתקיים $VCdim(\mathcal{H}) \leq \log_2 |\mathcal{H}|$. סיבוכיות הדגימות של למידת PAC על מחלקת המסווגים הבינאריים עם $VCdim(\mathcal{H}) = d$ היא:

$$C_1 \cdot \frac{d + \log(1/\delta)}{\varepsilon} \leq m_{\mathcal{H}}(\varepsilon, \delta) \leq C_2 \cdot \frac{d \cdot \log(1/\varepsilon) + \log(1/\delta)}{\varepsilon}$$

אם $VCdim(\mathcal{H}) = \infty$ לא נוכל לבצע למידה. אם נתונות $m < VCdim(\mathcal{H})$ דגימות ונרצה לחזות את הדגימה ה־$m+1 \leq VCdim(\mathcal{H})$, עדיין ניתן לנתץ את הקבוצה וקיימות שתי היפותזות שקולות שמסווגות באופן שונה את הדוגמא הנוספת.

Support Vector Machine (SVM)

לא פרמטרי. נשאף להפרדה טובה ־ רחוקה כמה שיותר מכל הדגימות, margin מקסימלי אפשרי (מרחק הנקודות הקרובות ביותר אליו). נמקסם את $J(w) = \frac{2}{\|w\|}$ תחת האילוץ $y_i\left(w^T x + b\right) \geq 1$ לכל i. בשלב הסיווג נחזה את $\hat{y} = sign\left(w^T x + b\right)$, כאשר:

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i; \quad b = y_i - w^T x_i$$

דגימה x_i היא Support Vector (SV) אמ"מ $\alpha_i > 0$.

ב־soft SVM יהיה slack לכל דגימה ־ כמה היא יכולה לשבור את תנאי המרחק מה־hyperplane המפריד. נרצה למזער את ה־hinge loss: $\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i$ תחת האילוץ $\xi_i \geq \max\left\{0, 1 - y_i a^T x_i\right\}$ לכל i. C קטן ־ נאפשר slacks גדולים וכך שגיאת אימון גדולה יותר (בנוסף, יותר נקודות בתוך ה־margin). במילים אחרות, ההפסד הוא:

$$J(w, \xi_1, \ldots, \xi_n) = C\sum_{i=1}^{n} \max\left\{0, 1 - y_i a^T x_i\right\} + \lambda \|w\|^2$$

עבור λ גדול נקבל $\|w\|^2$ קטן, וכך margin גדול יותר.

כשהדאטה לא פריד לינארית נרצה לסבך את המודל ע"י גרעין $K(x, y) = \varphi(x)^T \varphi(y)$. נטיל את הנקודות למימד גבוה יותר באמצעות φ ונפריד לינארית במימד החדש: $g(x) = w^T \varphi(x) + w_0$. נעלה מימדים באופן ישיר ע"י הגרעין. צירוף לינארי של גרעינים הוא גרעין.

$$g(x) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b$$

סיבוכיות המסווג תלויה במספר ה־SV שלו: רק עבורם $\alpha > 0$ והמודל תלוי רק בהם.

Cross Validation

LOOCV: עבור על כל נקודה באימון, הסר אותה מהדאטה (זמנית), אמן את המודל על שאר הדאטה וחזה את הנקודה שנפלה. בסוף התהליך ־ חשב שגיאה ממוצעת.
ב־k-fold בכל פעם נעיף חתיכה אחת (מתוך k), נאמן על שאר הדאטה ונבדוק עבור מה שהעפנו. ככל שיש יותר מידע, נוכל ועדיף k קטן.