(תיאורטי)

Prove that following Hard-SVM optimization problem is Quadratic Programming problem: (1)

$$\underset{}{\operatorname{argmin}} \|w\|^2 \quad s.t. \quad \forall i \quad y_i(\langle w_i, x_i\rangle + b) \geq 1 \quad (1)$$

That is, find matrices Q and A and vectors a and b s.t the above problem can be written in the following format

$$\underset{v \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} v^T Q v + a^T v \quad s.t \quad Av \leq d \quad (2)$$

Hint: Observe that $\|w\|^2 = w^T I w$

$$\forall i \quad y_i(\langle w, x_i\rangle + b) \geq 1 \iff -y_i x_i^T w - y_i b \leq -1 \iff$$

$$\begin{pmatrix} -y_1 x_1 & \cdots & -y_1 \\ \vdots & & \vdots \\ -y_m x_m & \cdots & -y_m \end{pmatrix} w - y_i b \leq -1$$

כך שאם נבחר את a^T לפיה ורגע A אקסטרים, $Q=2I$ אזי, נקבל שמתקיים:

$$\underset{v \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} v^T Q v + a^T v \quad st \quad Av \leq d$$

Consider the Soft-SVM optimization problem: (2)

$$\underset{w, \{\xi_i\}}{\operatorname{argmin}} \frac{\lambda}{2} \|w\|^2 + \frac{1}{m} \sum_i \xi_i \quad s.t \quad \forall i \quad y_i \langle w, x_i\rangle \geq 1 - \xi_i \wedge \xi_i \geq 0 \quad (3)$$

Denote the hinge-loss functions as $\ell^{hinge}(a) := \max\{0, 1-a\}$. Show that the Soft-SVM optimization problem is equivalent to the following unconstraint optimization problem:

$$\underset{w,\{\xi_i\}}{\arg\min} \frac{\lambda}{2}\|w\|^2 + \frac{1}{m}\sum_i \ell^{hinge}(y_i\langle w,x\rangle) \qquad (4)$$

נראה כי

$\sum l^{hinge}$ ו- $\sum \xi_i$ שקולות - עפ"י הגדרות מתקיים

כי - $\xi_i \geq 0$ וגם - $\xi_i \geq 1 - y_i\langle w, x_i\rangle$

כיון נחלק למקרים -

עבור $y_i\langle w, x_i\rangle > 1$ נבחר $\xi_i = 0$ והתנאים מתקיימים.

אחרת, נבחר $\xi_i \geq 1 - y_i\langle w,x_i\rangle$ כמו המינימלי שמקיים את התנאים

ולכן נקבל: $\xi_i = l^{hinge}(y_i\langle w,x_i\rangle) = \max(1-y_i\langle w,x_i\rangle, 0)$

(3)

The Gaussian Naive Bayes Classifier assumes a multinomial prior and independent feature-wise Gaussian likelihoods:

$$y \sim \text{Multinomial}(\pi)$$
$$x_j | y = k \overset{ind}{\sim} N(\mu_{k,j}, \sigma^2_{k,j}) \qquad (6)$$

for π a probality vector: $\pi \in [0,1]^K$, $\sum \pi_j = 1$.

(a) Suppose $x \in \mathbb{R}$ (i.e each sample has a single feature). Given a trainset $\{(x_i,y_i)\}_{i=1}^m$ fit a Gaussian Naive Bayes classifier solving (5) under assumptions (6). Fitting ~~[illegible]~~ means finding the expressions for the maximum likelihood estimators.

(חוק לפי מסווג בודד. לכל המחלקות, רצוי למצוא את התנאי כ"ס ניתן לכתוב במונה בלבד? (במונה שכינה לפי חוקי בייס).

$\arg\max f_{X|Y=k}(x) \cdot f_Y(k) = \int_{X,Y} f_{X,Y}(x,y) f_Y(k) = \arg\max_k \pi_k N(\mu_k|\sigma_k^2)$ לכן -

בסכום של ההוכחה שראינו בהרצאה נורמלית, $\pi_k = \frac{n_k}{m}$ ו- $\mu_k^{MLE} = \frac{1}{n_k}\sum_{[y_i=k]} x_i$ נוכל את ה- argmax

(6) Suppose $X \in \mathbb{R}^d$ (i.e each sample has d feature). Given a trainset $S = \{(x_i, y_i)\}_{i=1}^m$ fit a Gaussian Naive Bayes classifier solving (5) under assumptions (6). You are encouraged to use the results from (3.a).

כיון יש d פיצ'רים, וכיון שהם בלת"ל כ"ו לכן כפל. לכן.

$$\arg\max f_{X|Y=k}(X) = \prod_{i=1}^{d} \pi_k \cdot \frac{1}{\sqrt{2\pi}\sigma_{kj}} e^{-\frac{1}{2}(X-\mu_{kj})^T (\sigma^2_{kj})^{-1} (X-\mu_{kj})}$$

The Poisson Naive Bayes classifier assumes a multinomial prior and independent feature-wise Poisson likelihoods:

$$y \sim \text{Multinomial}(\pi)$$
$$X_j | y=k \overset{ind}{\sim} \text{Poi}(\lambda_{k,j}) \qquad (7)$$

for π a probability vector: $\pi \in [0,1]^K, \sum \pi_j = 1$.

(a) Suppose $X \in \mathbb{R}$ (i.e each sample has a single feature). Given a trainset $\{(X_i, y_i)\}_{i=1}^m$ fit a Poisson Naive Bayes classifier solving (5) under assumptions (7).

$$\arg\max f_{X|Y=k}(X) \cdot f_Y(x) = \arg\max_k \pi_k \text{Poi}(\lambda_k) =$$
$$= \arg\max_k \pi_k \prod \frac{\lambda_k^{x_i}}{x_i!} e^{-\lambda_k} \Rightarrow \log(\arg\max) = \log(\pi_k) + \sum \left(\log\left(\frac{\lambda_k^{x_i}}{x_i!}\right) e^{-\lambda_k}\right)$$

(b) Suppose $X \in \mathbb{R}^d$ (i.e each sample has d feature). Given a trainset $S = \{(x_i, y_i)\}_{i=1}^m$ fit a Poisson Naive Bayes classifier solving (5) under assumptions (7). You are encouraged to use the results from (4.a).

גם כאן נגדל אז לב הפיצ'רים הם בלתי תלויים, ומכיוון שהסכום של משתנים פואסונים מתפלג גם כן פואסוני, נקבל:

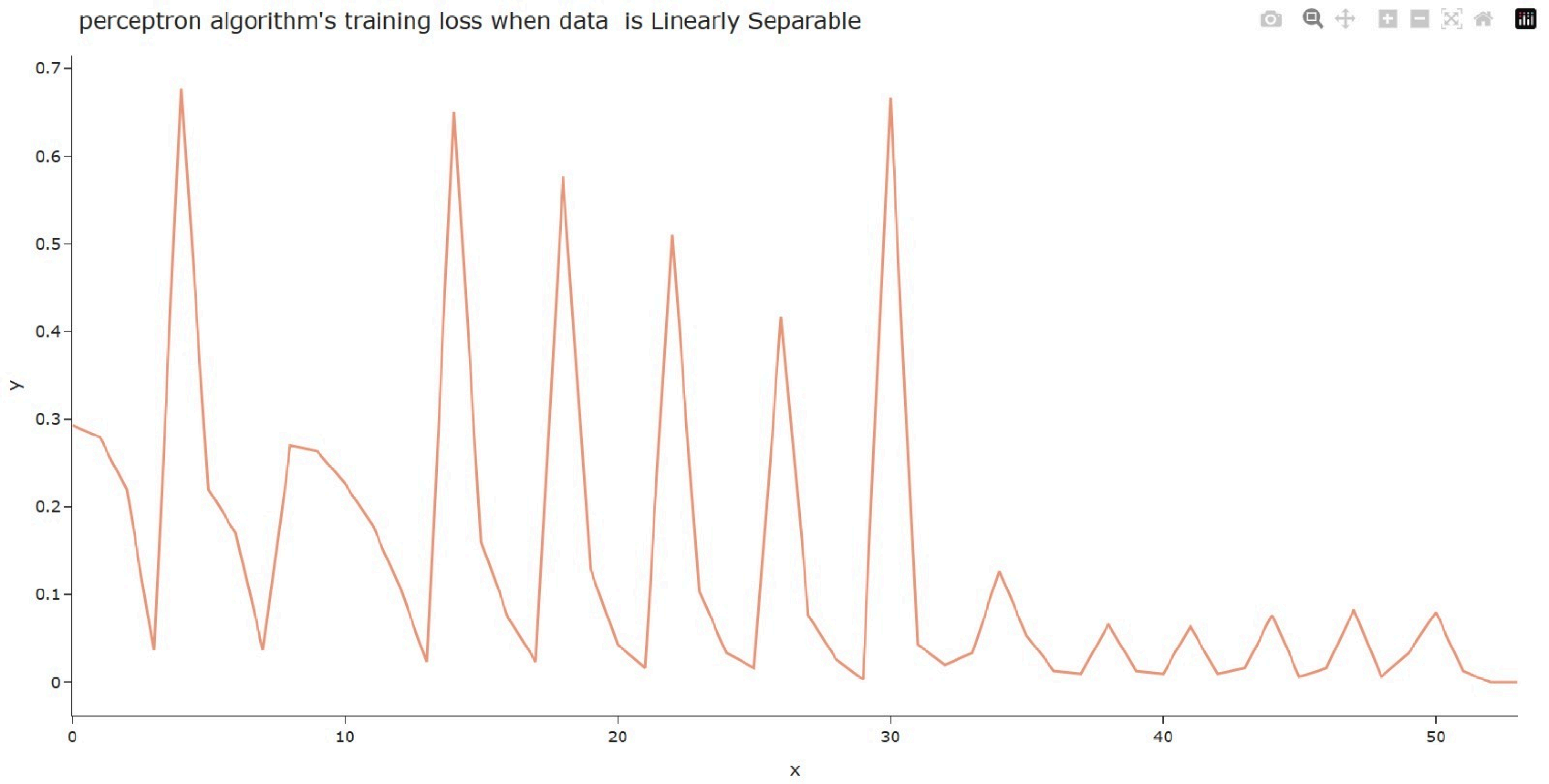
$$\arg\max = \log(\pi_k) + \log\left(\prod \frac{\lambda_k^{\sum_{j=1}^{d} x_i}}{x_i!} e^{-\lambda_k}\right)$$

פרקטי

301 סעיף 1

* מסיווג שאנו מתמודדים בנדבך זה במקרה הרינגרי, לאורך המנקד כיוון להפרדה לינארית ניתן לראות מהגרף איך שהוקטור איטרציות ותיקונים של w הפגע שלנו נורדף בהתאם עד שהוא מגיע לערך ס.

* ניתן לראות שהאלגוריתם מתקשה לאתר קלסיפיקציה סגורה זו מסוייםת, וככבירות שהתיקון של w עבור גדול מדי ולכן נוצרות הרעיות החדות בגרף. מהתגרה ה-30 האלגוריתם כנראה מדגים לו את אותם דגימות בדיוק בכונה ומשם לוחב ~~[illegible]~~ תנודות הוא מדויק להגיע לשגיאה אפסית.

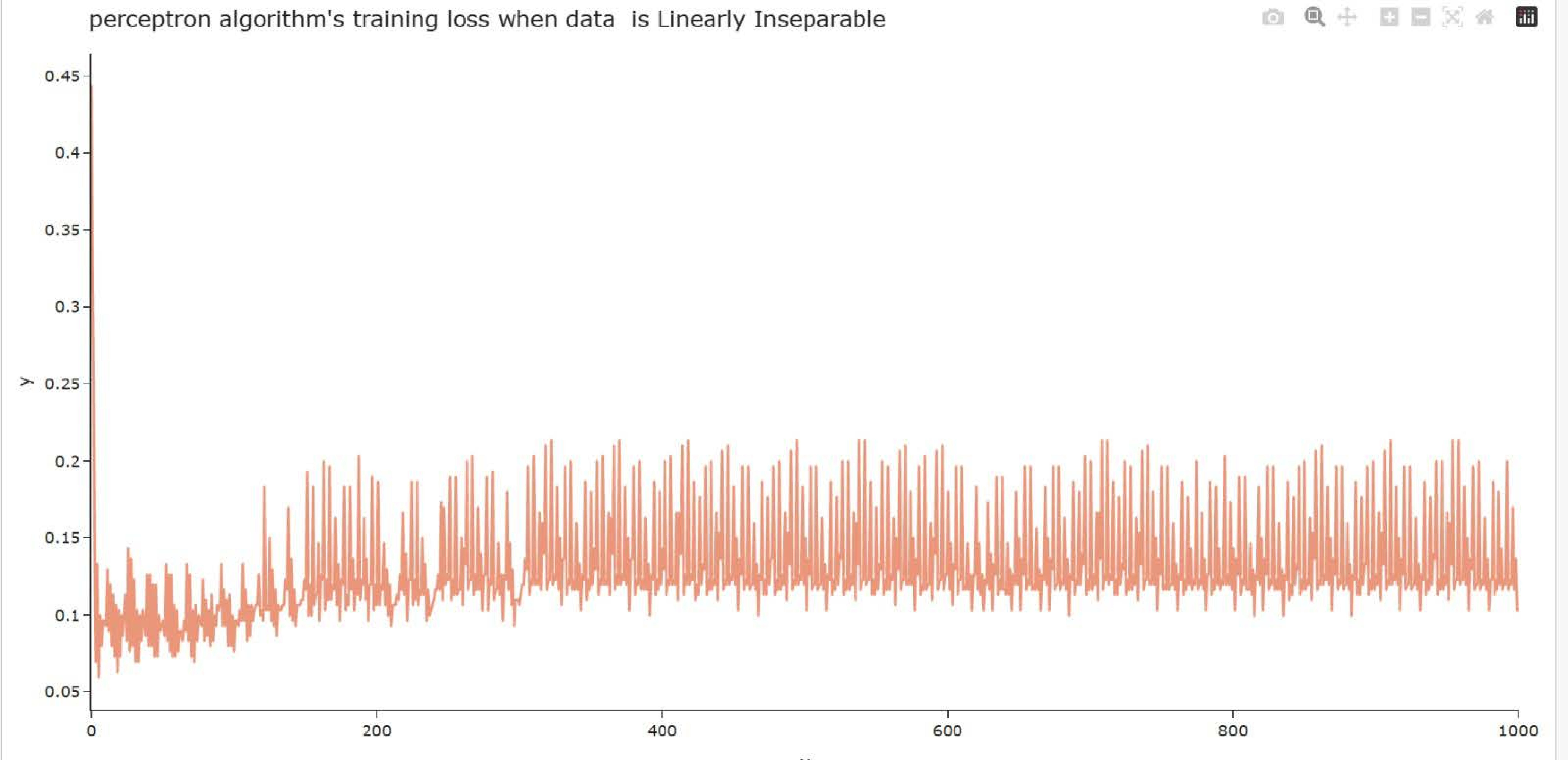

perceptron algorithm's training loss when data is Linearly Separable
y
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
x
0
10
20
30
40
50

3.1 סעיף 2

* בניגוד למקרה הקודם האלגוריתם לא מתכנס ועוצר רק כאשר הוא מגיע למספר האיטרציות המקסימלי, מכיוון שכאשר המידע לא ניתן להפרדה לינארית אנו בעצם נמצאים בבעיה שאין לה פתרון, ולכן האלגוריתם ממשיך לחפש היפר מישור שמפריד את המידע בלי שממש יכול למצוא אחד כזה.

* אין בעצם (w,b) כך שיתקיים ש: $y_i(\langle x_i, w\rangle + b) \geq 0 \quad \forall i \in [m]$

* האלגוריתם ממשיך למצוא דוגמאות שעליהן הוא טועה ובהתאם מתקן את w ובכך בכל פעם גורם לו לטעות על דוגמאות אחרות.

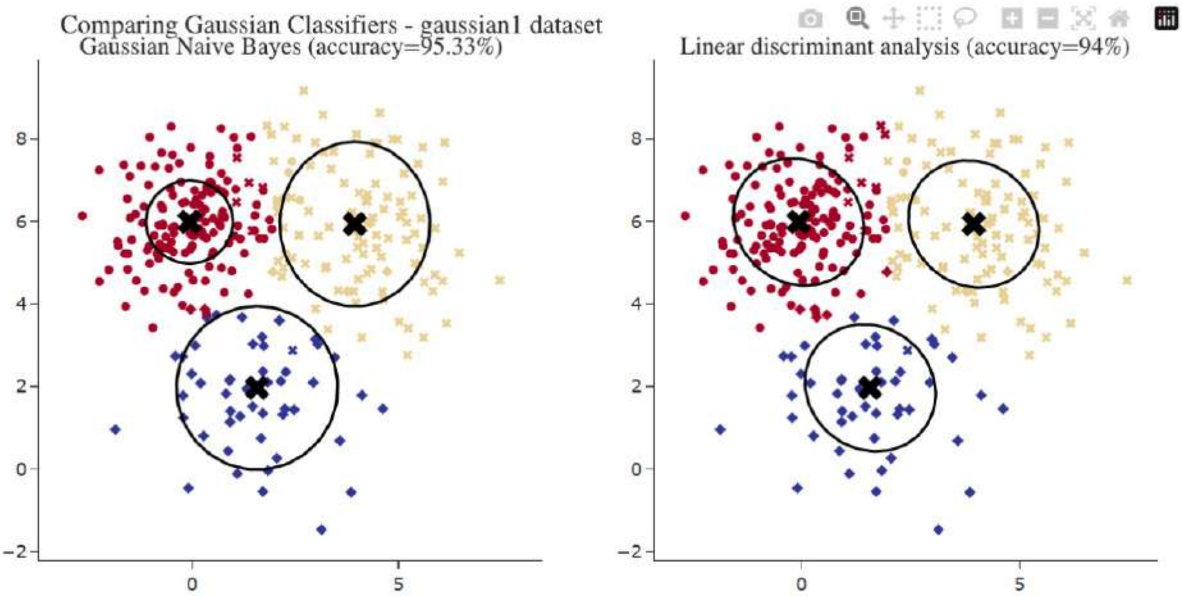


perceptron algorithm's training loss when data is Linearly Inseparable
y
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
200
400
600
800
1000

302

סעיף 1

תחילה ניתן להסיק כי הדגימות בקובץ הראשון אכן מתפלגות בקירוב כהתפלגות נורמליות כתלות בשיוך על המחלקה, כלומר ערך ממוצע כלשהו המשמש כמומנט לתוחלת של כל המחלקות. כמו כן, ניתן לראות כי עבור כל מחלקה בקובץ זה אין אינדיקציה לתלות כלשהי בין התפלגות ה-features השונים של הדגימות בכל מחלקה. לכן, מודל ה-GNB אשר מניח כי אין תלות בין פיצ'רים מפיק תחזית קבועה, טובה יותר מבחינת ה-accuracy מאשר מודל ה-LDA אשר מניח כי ישנה תלות בין התפלגות ה-features השונים, אך הדבר קיים והסך הכל בחשבון נלכן כמה הביקורת ~~[illegible]~~ שהם סך הכל דומה.



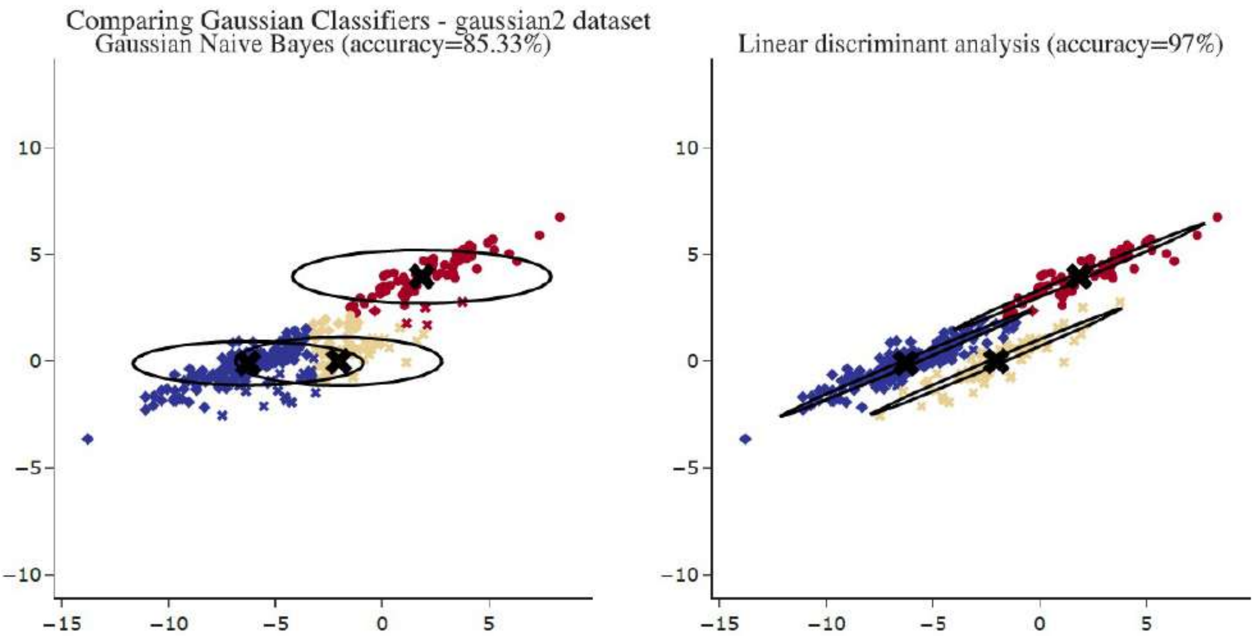
Comparing Gaussian Classifiers - gaussian1 dataset
Gaussian Naive Bayes (accuracy=95.33%)
Linear discriminant analysis (accuracy=94%)
8
6
4
2
0
−2
0
5

3.2 סעיף 2

במקרה הנ"ל ניתן לראות כי סביר מאוד שהנתונים נראים מהסתכלות כך שבכל מחלקה ישנה קורלציה בין שני ה-features, ולכן ישנה תלות ביניהם.

לכן, ניתן לראות כי במקרה זה רמת הדיוק של מודל ה-LDA מפגינה פרדיקציה מוצלחת יותר בצורה חכמה משמעותית על הנתונים לעומת מודל ה-GNB, מכיוון שהמודל מניח כי אין ישנה תלות כלשהי בין ההתפלגויות ה-features השונים.

לכן, ישנה עדיפות גבוהה להשתמש במודל ה-LDA במקרה בו אנו מאמינים כי אכן ישנה תלות כלשהי בין התפלגות ה-features של ההתפלגות ממנו נלקח המדגם.


Comparing Gaussian Classifiers - gaussian2 dataset
Gaussian Naive Bayes (accuracy=85.33%)
Linear discriminant analysis (accuracy=97%)
10
5
0
−5
−10
−15
−10
−5
0
5