תרגיל בית 2

יש להגיש קובץ ZIP ובו שני קבצים נפרדים: קובץ PDF ובו פתרון התרגיל כולל הפלטים של החלק המעשי וקובץ נוסף ובו הקוד שכתבתם. יש להקפיד על תשובות ברורות ומסודרות ועל קוד מסודר ומתועד היטב. רק אחד מבין חברי הזוג צריך להגיש את הפתרון.
שאלות על התרגיל יש לכתוב בפורום תרגילי הבית באתר הקורס. התרגיל מנוסח בלשון נקבה אך מתייחס לשני המינים.

שאלה 1

בהרצאה ראינו כי כאשר הנתונים ניתנים להפרדה לינארית, אלגוריתם perceptron מובטח להתכנס לפתרון בזמן סופי. בשאלה זאת תוכיחי זאת. (הנך מוזמנת לפנות לספר Understanding Machine Learning לטובת רמזים. גרסתו האינטרנטית נמצאת כאן)

יהי $w^* = \operatorname{argmin}\{\|w\|: \forall i, y_i\langle w, x_i\rangle \geq 1\}$ ויהיו $B = \|w^*\|, R = \max_i \|x_i\|$

א. הראי כי לאחר ביצוע T איטרציות של האלגוריתם מתקיים:

$$\langle w^*, w^{(T+1)}\rangle \geq T$$

ב. הראי כי לאחר ביצוע T איטרציות של האלגוריתם מתקיים:

$$\left\|w^{(T+1)}\right\|^2 \leq TR^2$$

ג. כעת הראי כי לאחר ביצוע T איטרציות של האלגוריתם מתקיים:

$$1 \geq \frac{\langle w^*, w^{(T+1)}\rangle}{\|w^*\|\|w^{(T+1)}\|} \geq \frac{\sqrt{T}}{RB}$$

ד. הסיקי כי האלגוריתם רץ לכל היותר $T \leq (RB)^2$ איטרציות ומכאן שמובטחת התכנסות.

ה. נקודה למחשבה, לא להגשה: האם מהירות ההתכנסות תלויה באופן ישיר בגודל המדגם?

חלק רטוב

בקוד המצורף לתרגיל בקובץ main.py ישנן שתי חתימות של פונקציות לשאלות 2-4. אתם רשאים לכתוב פונקציות עזר, אך את חתימות פונקציות אלו אסור לשנות מכיוון שבדיקת התרגילים מתבצעת באופן אוטומטי ומסתמך על שמות חתימות אלו.

שאלה 2

בשאלה זאת תממשי אלגוריתם gradient descent (למעשה מדובר במשפחה של אלגוריתמים בעלי עקרון פעולה דומה) למציאת נקודת מינימום של פונקציה קמורה במשתנה יחיד.

א. בחרי פונקציה מהצורה $y = a + bx + cx^2$ כאשר $a, b, c > 0$ פרמטרים לבחירתך, ממשי ב-Python את הפונקציה func(x) שבחרת והציגי גרף של הפונקציה.

ב. כתבי את הנגזרת של הפונקציה שבחרת וממשי אותה ב-Python בפונקצייה grad(x)

ג. כתבי את נקודת הקיצון של הפונקציה שבחרת.

כעת תממשי אלגוריתם למציאת נקודת המינימום של הפונקציה שבחרת.

אלגוריתם gradient descent מבוסס על אתחול של ערך x וחזרה על עדכון ערך x עד להתכנסות ע"י צעד העדכון הבא:

$$x_{n+1} = x_n - \alpha \nabla f(x_n)$$

כלומר, בכל צעד נקדם את x בכיוון הפוך לכיוון הגרדיאנט של הפונקציה. הפרמטר α נקרא קצב הלמידה.

ד. ממשי פונקציה updt(grad,x,a) אשר מקבלת פונקציית גרדיאנט, ערך x נוכחי ופרמטר α ומחזירה את ערך x המעודכן לפי הנוסחה לעיל.

ה. השתמשי בפונקציות שמימשת על מנת למצוא את נקודת המינימום של הפונקציה שבחרת. כלומר, אתחלי את ערך x למספר כלשהי וחזרי על הקריאה לפונקציית העדכון עד להתכנסות. מהו ערך x שאליו התכנס האלגוריתם שלך? האם הוא זהה לערך מסעיף ג'? הסבירי.

הערות:

- עלייך לקבוע מבחן להתכנסות. נהוג להשתמש במרחק בערך מוחלט בין הערך לפני העדכון ולזה שאחריו.
- עלייך לבחור את קצב הלמידה. נהוג לבחור בערכים נמוכים עבורו על מנת למנוע את התבדרות האלגוריתם.
- במידה והאלגוריתם לא מתכנס, נסי לבחור בערך אתחול שונה או בקצב למידה נמוך יותר.

שאלה 3

בשאלה זו נסווג תמונות מDataset בשם MNIST באמצעות SVM.

שימי לב, האובייקט המתקבל הינו מסוג Dictionary, במפתח 'data' נמצאות התמונות הרלוונטיות כבר בצורת וקטור. בקוד המצורף ישנה דוגמה לקריאה והורדה אוטומטית של הDataset.

תיאור הData:

Dataset זה כולל בתוכו תמונות של מספרים אשר נכתבו בכתב יד ותויגו בהתאם (ספרות 0-9)

- התמונות הן ממימד 28x28
- יש ברשותכם 70000 דוגמאות (בקוד המצורף מבוצעת חלוקה לסט אימון וסט מבחן)

א. התאימי מודל SVM על סט האימון והשתמשי בערך ברירת המחדל עבור הפרמטר cost ו וב-kernel = "linear".

ב. הציגי confusion matrix על נתוני האימון ונתוני המבחן וחשבי את שגיאת האימון ושגיאת המבחן.

ג. האם אפשר להשיג תוצאות טובות יותר בעזרת שינוי הפרמטרים בהם השתמשת בסעיפים א'-ב'? אם כן – מה השינויים שביצעת ובכמה השתפרו התוצאות? אם לא – למה? בתשובתך התייחסי לפרמטרים הטובים ביותר עבור ערך פונקציית הloss בסט האימון ובסט המבחן.

__רמז__ - נסי להשתמש ב"kernel="polynomial וב"kernel="rbf

ד. נתחי בקצרה את הניסויים והתוצאות שקיבלת עד כה. כחלק מהניתוח הציגי גרפים רלוונטיים של התוצאות שקיבלת.

שאלה 4

בשאלה זו תממשי את אלגוריתם Perceptron.

1. $\boldsymbol{w}^{(\mathbf{1})} = (\mathbf{0}, \dots, \mathbf{0})^{\boldsymbol{t}}$
2. **For** $\boldsymbol{t} = \mathbf{1}$ **to** $\boldsymbol{T}$:
    2.1 If there exists $\boldsymbol{i}$ such that $\boldsymbol{y_i w^t x_i} \leq \mathbf{0}$:
        2.1.1 set $\boldsymbol{w}^{(\boldsymbol{t}+\mathbf{1})} = \boldsymbol{w}^{(\boldsymbol{t})} + \boldsymbol{y_i x_i}$
        2.1.2 continue
    2.2 Else return $\boldsymbol{w}^{(\boldsymbol{t})}$

ממשי את האלגוריתם הנ"ל בפונקציה Perceptron. על האלגוריתם להחזיר את w לאחר ההתכנסות. שימי לב שפונקציה זו צריכה להיות גנרית כך שתוכל לקבל data מכל מימד.

על מנת לבדוק את איכות הפונקציה שכתבת, תוכלי לבדוק את הפונקציה עם מידע סינטטי מכאן:
http://scikit-learn.org/stable/datasets/index.html