הטכניון מכון טכנולוגי לישראל

הפקולטה להנדסת תעשייה וניהול

למידה חישובית 1 (096411)

חורף תשפ"ב 2021/22

תרגיל בית 3

תאריך אחרון להגשה: 30/12/2021 בשעה 23:55

Instructions - Read before you start the exercise

- **Submission is in pairs -** exceptions must be specifically approved by the course staff. Submission not in pairs, which hasn't been approved, **will be graded 0 automatically**.
- You are to submit a **zip file** with the name **HW3_ID1_ID2.zip,** where ID1 and ID2 are your student ids.
- The zip should contain **one** file:
    - A **single** *HW3_ID1_ID2.pdf* with a (detailed) solution of all of the written exercises (Include **all code and graphs** for every question).
- Submissions not in pairs please follow this convention:
    - *HW3_ID1.zip*
- Replace ID1, ID2 with your own student ids in **all** the files (**zip**, **py** and **pdf**)
- One submission per team – only one team member should submit.
- For questions with plotting, you can submit a **static** jupyter / google colab notebook (static == pdf format **only**). Written answers can be written inside text cells. You must **combine all the written questions to a single PDF file** (e.g., *question_1.pdf*, *question_3.pdf,* etc is **not allowed**).
- All code (inside notebooks) must be clear and concise (documented, using meaningful variable names, etc.)
- Every plot must contain at least the following: axis labels, units and legend.
- We will run your code using:
    - python 3.8.8
    - numpy 1.20.2
    - pandas 1.2.4
- Using other versions of python or python libraries is not recommended. It is at your own risk if the code fails to run due to version compatibility issues. We recommend using a clean anaconda virtual environment with the exact python and python modules + versions above installed.
- **No cheating** - if you are to copy your answers from other students and/or online references, you risk getting 0 for the submission and a disciplinary board. You may consult each other, but you are expected to write your own answers.
- Please use the HW forum for questions. Your questions could be helpful for other classmates. Generally, we will not answer questions sent by email to the course staff (unless there is a good reason to).

Good luck

שאלה 1

בשאלה זאת עליכם לממש אלגוריתם **gradient descent (GD)** למציאת נקודת מינימום של פונקציה גזירה וקמורה במשתנה יחיד.

א. בחרו פונקציה מהצורה $\boldsymbol{f(x) = a + bx + cx^2}$ כאשר $\boldsymbol{a, b, c > 0}$ פרמטרים לבחירתכם. ממשו בפייתון את הפונקציה $\boldsymbol{f}$ והציגו את גרף הפונקציה $[\boldsymbol{f(x)}]$ עבור $\boldsymbol{-10 \le x \le 10}$.

ב. הציגו ביטוי לנגזרת של הפונקציה אותה בחרתם וממשו אותה בפייתון עם הפונקציה $\boldsymbol{grad\_f(x)}$.

ג. כתבו את נקודת הקיצון של הפונקציה שבחרתם.

כעת תממשו אלגוריתם **GD** למציאת נק' המינימום של הפונקציה $\boldsymbol{f}$ שבחרתם. אלגוריתם GD מבוסס על אתחול של ערך $\boldsymbol{x}$ וחזרה על עדכון ערך $\boldsymbol{x}$ עד להתכנסות ע"י צעד העדכון הבא:

$$\boldsymbol{x_{t+1} = x_t - \eta \nabla f(x_t)}$$

כלומר, בכל צעד נקדם את $\boldsymbol{x}$ בכיוון הפוך לכיוון הגרדיאנט של הפונקציה. הפרמטר $\boldsymbol{\eta}$ נקרא קצב הלמידה (learning rate).

ד. ממשו פונקציה $\boldsymbol{grad\_update(grad, x, \eta)}$ אשר מקבלת פונקציית גרדיאנט, ערך $\boldsymbol{x}$ נוכחי ופרמטר $\boldsymbol{\eta}$ ומחזירה את ערך $\boldsymbol{x}$ המעודכן לפי הנוסחא לעיל.

ה. השתמשו בפונקציות שמימשתם על מנת למצוא את נקודת המינימום של הפונקציה שבחרתם. כלומר, אתחלו את ערך $\boldsymbol{x}$ למספר כלשהו וחזרו על הקריאה לפונקציית העדכון עד להתכנסות. מהו ערך $\boldsymbol{x}$ שאליו התכנס האלגוריתם שלכם? האם הוא זהה לערך מסעיף ג'? הסבירו.

<u>הערות לסעיף:</u>

- עליכם לקבוע מבחן להתכנסות. נהוג להשתמש במרחק בערך מוחלט בין הערך לפני העדכון ולזה שאחריו, כלומר על האלגוריתם לעצור כאשר ההבדל בין 2 ערכים עוקבים בערך מוחלט קטן מסף $\boldsymbol{\epsilon}$ כלשהו (הנתון לבחירתכם).
- עליכם לבחור את קצב הלמידה. נהוג לבחור בערכים נמוכים עבורו על מנת למנוע את התבדרות האלגוריתם.
- במידה והאלגוריתם לא מתכנס, נסו לבחור בערך אתחול שונה, קצב למידה נמוך יותר או סף נמוך יותר.

ו. חזרו על סעיף ה' כאשר כעת אתם שומרים את ערכו של $\boldsymbol{x}$ בכל איטרציה. כלומר, עליכם לשמור רשימה המכילה את $[\boldsymbol{x_0, x_1, \dots, x_T}]$ כאשר $\boldsymbol{T}$ הוא מספר האיטרציות בריצת האלגוריתם שלכם. עבור אילו ערכים (Hyperparameters) האלגוריתם מתכנס ומתקבל ערך $\boldsymbol{T}$ נמוך? (אין צורך למצוא את ערך $\boldsymbol{T}$ הנמוך ביותר האפשרי; מספיק לנסות מספר פעמים עם ערכי התחלה, סף התכנסות וקצב למידה שונים. <u>לא לתת את האופטימום כערך התחלה</u>).

ז. עבור ה-$\boldsymbol{T}$ הנמוך ביותר שמצאתם, הציגו ב<u>תרשים אחד</u> את גרף הפונקציה $\boldsymbol{f}$ שלכם (כמו בסעיף א') ואת ערכי הפונקציה עבור $[\boldsymbol{x_0, x_1, \dots, x_T}]$. כלומר, הציגו מעל גרף הפונקציה $\boldsymbol{f}$ תרשים scatter plot כשציר ה-X הוא $[\boldsymbol{x_0, x_1, \dots, x_T}]$ וציר ה-Y הוא $[\boldsymbol{f(x_0), f(x_1), \dots, f(x_T)}]$. יש להקפיד לשרטט את 2 הגרפים (קו הפונקציה ותרשים הנקודות) בצבעים שונים.

שאלה 2

בשאלה זו תממשו את אלגוריתם ה-(SGD) Stochastic Gradient Descent לפתרון בעיית ה-soft-SVM.

תזכורת: בהינתן מדגם אימון המכיל m תצפיות $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ כך שלכל אינדקס i, $x_i \in \mathbb{R}^d$ ו-$y_i \in \{+1, -1\}$ ובהינתן פרמטר רגולריזציה $\lambda > 0$, בעיית ה-soft-SVM מוגדרת באופן הבא

$$(w^*, b^*) = \arg min_{\substack{w \in \mathbb{R}^d \\ b \in \mathbb{R}}} \frac{1}{m} \sum_{i=1}^{m} \max\{0, 1 - y_i(\langle w, x_i \rangle + b)\} + \lambda ||w||^2$$

כאשר $w \in \mathbb{R}^d$ הוא וקטור הפרמטרים המוכר ו-$b \in \mathbb{R}$ הוא פרמטר ה-bias שהוזכר בהרצאה ובתרגול.

א. הסבירו מדוע פונקציית המטרה קמורה (אין צורך בהוכחה פורמלית). היעזרו בתכונות ובטענות שנלמדו בתרגול. בנוסף, השתמשו בכך ש $f(x)^2$ היא פונקציה קמורה אם f(x) היא פונקציה קמורה.

ב. הוכיחו כי במקרה ההומוגני (b=0), פונקציית ה-Hinge Loss

$$l(w, x_i, y_i) \coloneqq \max\{0, 1 - y_i \cdot \langle w, x_i \rangle\}$$

בכל נקודה (x_i, y_i), היא $R - Lipschitz$ ביחס ל-w, כאשר $R \coloneqq \max_k ||x_k||$.

רמז: חלקו למקרים והשתמשו באי-שוויון קושי שוורץ.

ג. כזכור, ב-SGD כיוון העדכון בכל איטרציה נקבע לפי תצפית כלשהי (x_i, y_i) שנדגמה באקראי ממדגם האימון. הציגו sub-gradient של פונקציית המטרה לפי אותה התצפית.
וודאו כי אתם מתייחסים גם ל-subgradient לפי המשתנה w וגם ל-subgradient לפי המשתנה b.

ד. כתבו פונקציה בשם svm_with_sgd המקבלת את הפרמטרים הבאים (אין צורך לבדוק את תקינות הקלט):
- X – מטריצת נקודות האימון בה השורה ה-i היא הנקודה x_i.
- y – וקטור הלייבלים בו כל כניסה i היא הלייבל של הנקודה x_i.
- lam – פרמטר רגולריזציה אי-שלילי. ערך ברירת המחדל הוא 0.
- epochs – מספר הפעמים בהם האלגוריתם יעבור על **כל** מדגם האימון. ערך ברירת המחדל הוא 1000.
- l_rate – גודל צעד העדכון. ערך ברירת המחדל הוא 0.01.
- sgd_type – משתנה מחרוזת דגל. (יכול לקבל 'practical' או 'theory') ברירת מחדל הוא 'practical'.

<u>על הפונקציה לבצע את השלבים הבאים:</u>
אם sgd_type = 'practical' אז בצע את הגישה הפרקטית של SGD דרך השלבים הבאים:
1. חילוץ ממדי המטריצה X בכדי לגלות מהו m ומהו d.
2. אתחול הווקטור w וסקלר b בערכים המגיעים מהתפלגות אחידה רציפה $U(0,1)$.
3. בכל epoch תדגם פרמוטציה המתארת את הסדר בו תבחר כל תצפית מתוך המטריצה X והווקטור ה-y. לפי הפרמוטציה יתבצע מעבר על תצפיות האימון ועבור כל אחת מהן:
   3.1. יחושב sub-gradient של פונקציית המטרה לפי התצפית הנוכחית.
   3.2. יעודכנו הווקטור w והסקלר b לפי ה-subgradient של כל אחד מהם ולפי גודל צעד העדכון.
4. החזרת w ו-b. (הוקטור משקולות ואיבר ההטייה האחרון)

אם sgd_type = 'theory' אז בצע את אלגוריתם SGD התיאורטי שנלמד בהרצאה דרך השלבים הבאים:
1. חילוץ ממדי המטריצה X בכדי לגלות מהו m ומהו d.
2. אתחול הווקטור w וסקלר b בערכים המגיעים מהתפלגות אחידה רציפה $U(0,1)$.
3. בצע m*epoch איטרציות כאשר בכל איטרציה:
   3.1. תודגם תצפית בודדת באקראי מ X.
   3.2. יחושב sub-gradient של פונקציית המטרה לפי התצפית הנוכחית.

3.3. יעודכנו הווקטור w והסקלר b לפי ה-subgradient של כל אחד מהם ולפי גודל צעד העדכון.

4. החזרת $\bar{w}$ ו-$\bar{b}$. (ממוצע וקטורי המשקלים וממוצע איברי ההטייה) כלומר:

$$\bar{w} = \frac{1}{epoch * m} \sum_{t=1}^{epoch*m} w_t \qquad \bar{b} = \frac{1}{epoch * m} \sum_{t=1}^{epoch*m} b_t$$

כאשר w_t ו b_t הם וקטור המשקולות ואיבר ההטייה בצעד t.

ה. כתבו פונקציה בשם calculate_error המקבלת וקטור משקולות w, פרמטר bias, מטריצת נקודות X ווקטור הלייבלים המתאים לה y. על הפונקציה לחשב ולהחזיר את השגיאה של המסווג הלינארי המוגדר על ידי w ופרמטר ה- bias.

ו. בסעיף זה תבחנו את אלגוריתם ה-SGD שכתבתם כאשר **sgd_type = ‘practical’**

1. הוסיפו בראש הפונקציה אותה כתבתם בסעיף ד’, את שורת הקוד הבאה:

```
np.random.seed(2)
```

המקבעת את המנגנון הרנדומי ומאפשרת לכם להשוות בין הרצות שונות.

2. תטענו את iris dataset –

```
from sklearn.datasets import load_iris
X, y = load_iris(return_X_y=True)
X = X[y != 0]
y = y[y != 0]
y[y==2] = -1
X = X[:, 2:4]
```

3. הפרידו את הדאטה למדגם אימון ומדגם וולידציה באופן הבא –

```
from sklearn.model_selection import train_test_split
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.3, random_state=0)
```

4. לכל $\lambda \in \{0, 0.05, 0.1, 0.2, 0.5\}$, אמנו מודל SVM בעזרת הפונקציה אותה כתבתם בסעיף ד' (בסה"כ 5 מודלים). לכל מודל חשבו את שגיאת האימון, שגיאת המבחן ורוחב ה-margin (החד צדדי).

5. הציגו 2 גרפי bar-plot:
   a. גרף המציג את שגיאת האימון ושגיאת המבחן לכל מודל (סה"כ 5 **זוגות** של עמודות), כאשר השגיאה מוגדרת כ: $Error = 1 - Accuracy$**.**
   b. גרף המציג את רוחב ה margin (החד צדדי) כפונקציה של λ. כלומר עליכם להציג את רוחב ה margin כפונקציה של כל אחד מחמשת המודלים (סה"כ 5 עמודות).
   ע"פ הגרפים בסעיף זה – איזה מודל מחמשת המודלים נראה כטוב ביותר? כיצד אתם מסבירים זאת? התייחסו ל λ בתשובתכם.

ז. עבור ה λ שבחרתם בסעיף ו.5 הציגו 2 גרפים:
   a. גרף המציג את שגיאת <u>האימון</u> כפונקציה של משתנה ה- *epochs* באלגוריתם ה *SGD* שכתבתם.
   כאשר ערכי ה- *epochs* נעים בין 10 ל- 1000 (כולל) בקפיצות של 10.
   הגרף יכיל שתי עקומות כאשר עקומה אחת עבור הרצת האלגוריתם עם **sgd_type = ‘practical’** , ועקומה שנייה את **sgd_type = ‘theory’**.
   b. חזרו על a אבל כאשר מציגים את שגיאת <u>המבחן</u> כפונקציה של מס' ה- epochs.
   הסבירו את התוצאות שקיבלתם.

שאלה 3

כחלק מתהליך בחירת מודל, בחירת היפר-פרמטרים ובחירת משתנים מסבירים (פיצ'רים), למדנו על Cross Validation (CV).
בתרגיל זה נממש תהליך של בחירת *קונפיגורציה* של מודל (כלומר בחירת סוג המודל ובחירת היפר-פרמטרים עבורו) באמצעות k-Fold CV.

א. ממשו פונקציה בשם **cross_validation_error(X, y, model, folds)** כאשר:

- $X \in \mathbb{R}^{m \times d}$ – מטריצת הנתונים (מטיפוס numpy nd-array)
- $y \in \mathbb{R}^{m}$ – וקטור הלייבלים (מטיפוס numpy nd-array)
- $model$ – אובייקט מודל התומך בפונקציות fit, predict (לדוגמא אובייקט SVC של sklearn)
- $folds$ – מספר ה"קיפולים" עבור k-Fold CV (מספר שלם)

על הפונקציה להחזיר tuple המכיל את האיברים הבאים: (average_train_error, average_val_error) כאשר:

- average_train_error – שגיאת ה**אימון** הממוצעת על גבי כל ה-folds
- average_val_error – שגיאת ה**ולידציה** הממוצעת על גבי כל ה-folds

הערה: השגיאה במקרה הזה מוגדרת כ: $Error = 1 - Accuracy$.

<u>הערה חשובה לסעיף</u>: <u>**אסור**</u> לכם להשתמש בפונקציות עזר מהספריה sklearn עבור סעיף זה. בפרט אסור לכם להשתמש בפונקציה cross_val_score מתוך sklearn.

ב. ממשו פונקציה בשם **svm_results(X_train, y_train, X_test, y_test)** כאשר:

- $X_{train} \in \mathbb{R}^{m_{train} \times d}$ – מטריצת הנתונים עבור סט האימון (מטיפוס numpy nd-array)
- $y_{train} \in \mathbb{R}^{m_{train}}$ – וקטור הלייבלים עבור סט האימון (מטיפוס numpy nd-array)
- $X_{test} \in \mathbb{R}^{m_{test} \times d}$ – מטריצת הנתונים עבור סט המבחן (מטיפוס numpy nd-array)
- $y_{test} \in \mathbb{R}^{m_{test}}$ – וקטור הלייבלים עבור סט המבחן (מטיפוס numpy nd-array)

על הפונקציה להשתמש בפונ' **cross_validation_error** מסעיף א' עם folds=5 בכדי לחשב את שגיאות האימון והולידציה הממוצעת של אלגוריתם SVM, **לכל** פרמטר $C = 1/\lambda$ כאשר

$\lambda \in \{10^{-4}, 10^{-2}, 1, 10^{2}, 10^{4}\}$. כלומר הפונקציה צריכה להריץ 5-fold CV **חמש** פעמים. בנוסף, לכל פרמטר λ, הפונקציה צריכה להתאים מודל *SVM* עבור **כל** מדגם האימון ולחשב את שגיאת ה**מבחן**.

הפונקציה צריכה להחזיר מילון (dictionary) כאשר המפתחות (keys) הם שמות המודל (לדוגמא: 'SVM_lambda_*100*' והערכים (values) הינם tuple מהצורה הבאה:

(average_train_error, average_validation_error, test_error)

כאשר 2 האלמנטים הראשונים מחושבים ע"י 5-fold CV והאלמנט האחרון מחושב ע"י מודל בודד שמתאמן על כל מדגם האימון.

ג. טענו את סט הנתונים iris באמצעות הפקודות הבאות:

```
from sklearn.datasets import load_iris
iris_data = load_iris()
X, y = iris_data['data'], iris_data['target']
```

חלקו את סט הנתונים לסט אימון וסט מבחן באמצעות הפקודה הבאה:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=7)
```

הריצו את הפונקציה מסעיף ב' על הנתונים שטענתם בסעיף ג'. ציירו גרף עמודות (bar plot) המציג את התוצאות של כל ניסוי. כלומר, ציר ה-x יתאר את ערכי λ וציר ה-y יתאר את שגיאת האימון הממוצעת, שגיאת הולידציה הממוצעת ושגיאת המבחן (סה"כ 5 **שלשות** של עמודות). יש להקפיד על צבע שונה לכל סוג של עמודה (אימון / ולידציה / מבחן).

מיהו המודל הטוב ביותר לפי שיטת CV? מיהו המודל הטוב ביותר על מדגם המבחן? האם מדובר באותו המודלל? הסבירו מדוע.

שאלה 4

תהי $g(w) = \max_{i\in[r]} g_i(w)$ עבור r פונקציות $g_1, g_2, \dots, g_r$ כאשר לכל i , $g_i: R^d \to R$
היא פונקציה קמורה וגזרה בכל R^d.
עבור w מסוים, ניקח $j \in \arg\max_i g_i(w)$, הוכיחו ש - $\nabla g_j(w)$ הוא סאב־גרדיאנט של הפונקציה g בנקודה w.
כלומר הראו **שלכל** $\boldsymbol{u} \in \boldsymbol{R^d}$ מתקיים האי שוויון הבא:

$$g(u) \geq g(w) + \langle u - w, \nabla g_j(w) \rangle$$