

מטלת מנחה (ממ"ן) 11

הקורס: 22961 למידה עמוקה

חומר הלימוד למטלה: יחידה 1

מספר השאלות: 3

סמסטר: 2022ב

משקל המטלה: 5 נקודות

מועד אחרון להגשה: 01.04.2022

קיימות שתי חלופות להגשת מטלות:

- שליחת מטלות באמצעות מערכת המטלות המקוונת באתר הבית של הקורס
 - שליחת מטלות באמצעות הדואר או הגשה ישירה למנחה במפגשי ההנחיה
- הסבר מפורט ב"נוהל הגשת מטלות מנחה"

שאלה 1 (30 נקודות)

בשאלה זו עליכם לכתוב בנאי לטנזור אשר דוגם מספרים אקראיים מהתפלגות בדידה אשר הוא מקבל כקלט. חתימת הפונקציה תהיה

```
my_sampler(size, dist, requires_grad=False)
```

הפונקציה תחזיר טנזור בגודל `size` של מספרים אשר נדגמו מהתפלגות `dist`: `dist[i]` היא ההסתברות לדגום את המספר `i`.

הנחיות:

- יש לבדוק את תקינות הקלט `dist`: יש לבדוק שסכום ההסתברויות הוא אחד, וכן שכל איבר בוקטור הוא חיובי.
- אם בפרמטר `requires_grad` מתקבל ערך אמת יש להחזיר כפלט טנזור אשר מערכת הגזירה האוטומטית עוקבת אחריו.
 - חיוני שהמערכת לא תעקוב אחרי החישובים שמבצע הבנאי עצמו.
- בתוך הבנאי מותר להשתמש בלולאה אחת בלבד אשר מתחילה כך:

```
for i in range(dist.numel()-1):
```

- פונקציות ומתודות שימושיות:

```
assert, torch.rand, torch.cumsum, tensor.sum(), tensor.all()
```

- אתם יכולים להשתמש באלגוריתם לדגימת מ"מ בדיד הבא:

קלט: וקטור הסתברויות $dist = (p_0, p_1, \dots, p_n)$, דגימה של מ"מ אחיד סטנדרטי

$$U \sim U[0,1]$$

פלט: דגימה אקראית של מספר $I \in \{0, 1, \dots, n\}$ כך ש- $P(I=k) = p_k$

האלגוריתם:

1. אם $U < p_0$, החזירו $I = 0$.

$$2. \text{ אם } U \geq \sum_{i=0}^{n-1} p_i \text{ החזירו } I = n.$$

אחרת:

$$3. \text{ מצאו את הערך היחיד } k \in \{1, \dots, n-1\} \text{ עבורו מתקיים } U < \sum_{i=0}^k p_i \text{ וכן } U \geq \sum_{i=0}^{k-1} p_i.$$

(אין צורך לבדוק שקיים כזה ערך ושהוא יחיד).

$$4. \text{ החזירו } I = k.$$

- דגמו 10,000 מ"מ מההתפלגות [0.1, 0.2, 0.7] וציירו היסטוגרמה של הדגימות

בעזרת הפונקציה `matplotlib.pyplot.hist`.

בונוס (10 נק'): השתמשו באלגוריתם Alias Method במקום האלגוריתם הנ"ל. קראו עליו כאן:

https://en.wikipedia.org/wiki/Alias_method

דוגמאות לקלט ופלט:

```
>>>my_sampler(10,[0.5,0.5])

tensor([0., 0., 0., 0., 1., 0., 1., 0., 1., 0.])

>>>A=my_sampler((2,8),[0.1,0.2,0.7],requires_grad=True)
>>>print(A,A.grad,sep='\n')

tensor([[2., 0., 2., 0., 2., 2., 2., 0.],
        [2., 2., 2., 2., 0., 2., 2., 1.]], requires_grad=True)
None
```

שאלה 2 (30 נקודות)

בשאלה זו עליכם לממש בעצמכם את פונקציונליות השידור (broadcast) של PyTorch. בכל סעיפי

השאלה אין להשתמש בפונקציות/ מתודות מובנות של PyTorch המבצעות בעצמן שידור.

א. כתבו פונקציה אשר תקבל כקלט שני טנזורים, A ו-B, ותחזיר כפלט טנזור שלישי, C, שהוא

השידור של A למימדיו של B. למעשה, בסעיף זה עליכם לממש את הפונקציונליות של

המתודה `A.expand_as(B)`.

הנחיות:

- ראשית יש לבדוק ש-A ניתן לשידור למימדו של B ואחרת להחזיר שגיאה.

- מותר להקצות מקום חדש בזיכרון עבור C.

ב. כתבו פונקציה המקבלת כקלט שני טנזורים, ובודקת אם הם ניתנים לשידור יחדיו, לפי

חוקי השידור. הפונקציה תחזיר ערך אמת/שקר וכן פלט נוסף אם הם ברי שידור יחדיו:

הגודל שאליו הטנזורים ישודרו.

ג. כתבו פונקציה המקבלת כקלט שני טנזורים, ומשדרת אותם יחדיו. הפונקציה תחזיר כפלט שני טנזורים חדשים – תוצאות השידור של שני טנזורי הקלט.

הנחיות:

- ניתן להשתמש בסעיפי השאלה הקודמים.
 - מותר להקצות מקום חדש בזיכרון עבור טנזורי הפלט.
- ד. הריצו את הפונקציות אשר כתבתם על קלט לדוגמה והשוו את התוצאה לזו המתקבלת מהפונקציות המובנות ב-PyTorch. הפונקציות הרלוונטיות הן:
- `broadcast_tensors, expand_as.`

שאלה 3 (40 נקודות)

בשאלה זו יהיה עליכם לממש מערכת גזירה אוטומטית בעצמכם. על המערכת לעקוב אחרי פעולות החשבון המתבצעות על המשתנים ולחשב יחד עם כל פעולה את הנגזרת "המיידיית" שלה: הנגזרת של הפלט לפי הקלט. לבסוף יהיה על המערכת לחשב נגזרות לפי כלל השרשרת.

למערכת יהיו שלושה רכיבים:

1. מחלקה בשם `MyScalar` אשר כל אובייקט שלה מכיל שלושה מאפיינים:
 - ערך הסקלר (מספר).
 - הנגזרת המיידיית שהתקבלה בעת חישוב אובייקט זה (מספר).
 - ה"אב" של אובייקט זה – הקלט לפונקציה אשר בעזרתה אובייקט זה חושב (אובייקט `MyScalar` קיים, אם קיים "אב" כזה).
2. ספריית פונקציות מתמטיות אשר מקבלות כקלט אובייקט של המחלקה `MyScalar`, ומחזירות אובייקט נוסף של מחלקה זו. תוצאת חישוב הפונקציה תהיה שמורה במאפיין הערך והנגזרת של הפונקציה לפי משתנה הקלט תהיה שמורה במאפיין הנגזרת המיידיית. יש לממש את הפונקציות $e^a, \ln a, \sin a, \cos a, a^n, n \cdot a, a + n$. שימו לב שיש לממש יחד עם הפונקציות גם את חישוב הנגזרת (לפי המשתנה a בלבד).
3. פונקציה בשם `get_gradient` אשר מקבלת כקלט אובייקט `MyScalar` ומחזירה כפלט מילון המכיל את הנגזרת שלו לפי כל אחד מהמשתנים אשר הוא תלוי בהם. למשל, לאחר הרצת קטע הקוד הבא,

```
a=MyScalar(2)
b=power(a,2) #a^2
c=exp(b)
d=get_gradient(c)
```

הפלט אשר יתקבל ב-`d` הוא מילון המכיל ערכים עבור a, b, c , כך ש-`d[x]` היא הנגזרת של c לפי x .

שימו לב שעבור המשתנים המוגדרים בקטע הקוד הנ"ל מתקיים

$$\frac{dc}{dc} = 1,$$

$$\frac{dc}{db} = \frac{de^b}{db} = e^b,$$

$$\frac{dc}{da} = \frac{de^b}{da} = \frac{de^b}{db} \cdot \frac{db}{da} = \frac{de^b}{db} \cdot \frac{da^2}{da} = e^b \cdot 2a$$

ובעזרת חוקיות זו על המערכת לחשב את כל הנגזרות. כמו כן שימו לב שבעת חישוב

$b = \text{power}(a, 2)$ על המערכת לחשב גם כן את $\frac{db}{da} = 2a$ ולשמור ערך זה במאפיין הנגזרת

המיידית של b . באופן דומה גם $\frac{dc}{db} = e^b$ יחושב ויישמר במאפיין של c . לבסוף, בעת הקריאה

לפונקציה `get_gradient` ערכי הנגזרות המיידיות ישולבו לחישוב כל הנגזרות לפי כלל השרשרת.

הנחיות:

- ניתן להשתמש בפונקציות המובנות של PyTorch לצורך חישוב הפונקציות המתמטיות.
 - מומלץ לממש את `get_gradient` באופן רקורסיבי (הגדירו את מילון הפלט כמשתנה גלובלי).
 - השוו תוצאותיכם למערכת הגזירה האוטומטית של PyTorch. בצעו חישובים זהים בעזרת שתי המערכות ובדקו שהתוצאות המתקבלות זהות.
- בונוס (10 נק'):** הרחיבו את מערכת הגזירה האוטומטית שכתבתם כך שתוכלו לחשב נגזרות גם דרך פונקציות בעלות שני משתני קלט. ממשו את הפונקציות $a \cdot b$, $a + b$, a / b כאשר כעת יש לגזור גם לפי a וגם לפי b . שימו לב שבמקרה זה כלל השרשרת הוא

$$\frac{df(b, c)}{da} = \frac{\partial f(b, c)}{\partial b} \cdot \frac{db}{da} + \frac{\partial f(b, c)}{\partial c} \cdot \frac{dc}{da}$$

ובהתאם לאובייקט `MyScalar` הנוצר על ידי אחת הפונקציות האלו יהיו שני "אבות".

יש להגיש בקובץ ZIP אחד:

1. מחברת ג'ופיטר המכילה את הקוד והפלט הדרוש לכל השאלות.
 2. קישור לסרטון קצר בו אתם מסבירים את הקוד, מריצים אותו ומנתחים/מסבירים את הפלט (את הסרטון יש להכין בעזרת תוכנת Zoom ושימוש בפונקציה `Share screen`).
 3. תשובות בכתב ככל שנדרש.
- מומלץ להשתמש במערכת הגשה האינטרנטית.