

## תזמון קצב הלמידה

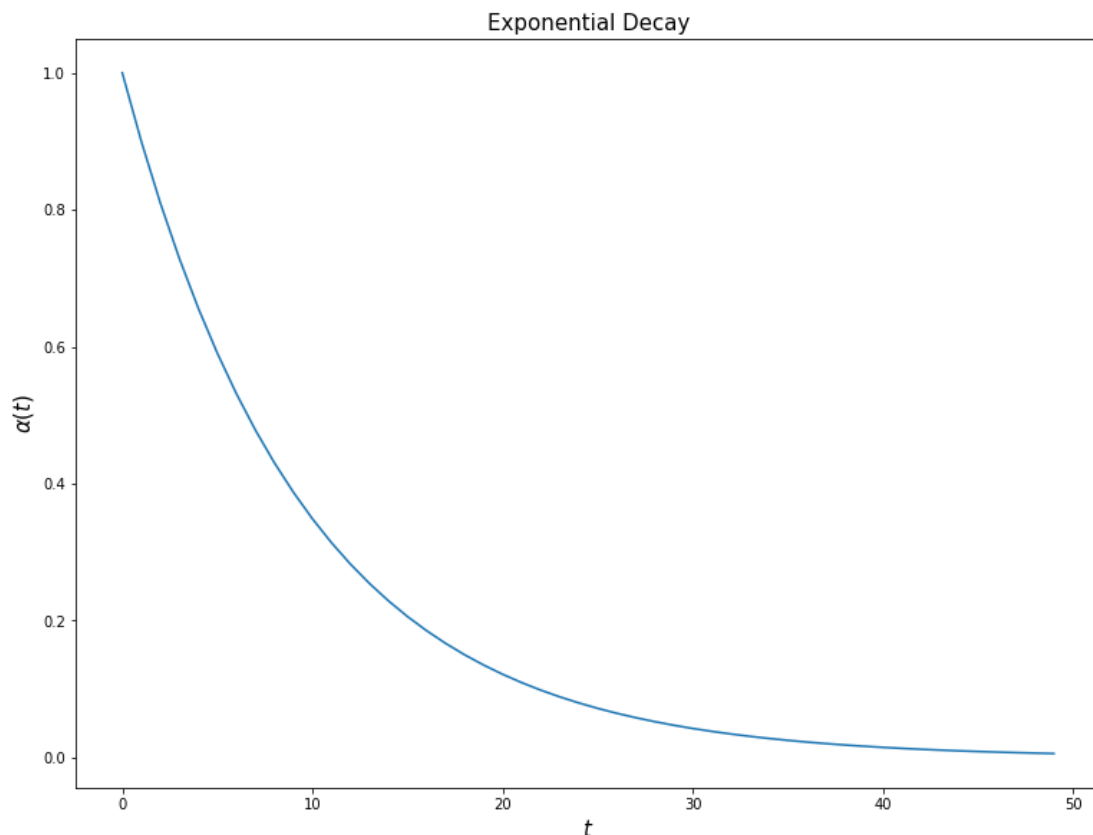
השיפור הראשון אשר נשלב באלגוריתם האופטימיזציה SGD הוא **דעיכת קצב הלמידה** – בכל איטרציה נקטין את קצב הלמידה לפי כלל קבוע, כך שלאחר מספר רב של איטרציות, גם אם הגרדיאנט עוד לא התאפס, קרי האלגוריתם עוד לא התכנס לערכי פרמטרים אשר מהווים נקודת מינימום של פונקציית המחיר, קצב הלימוד יהיה אפסי בפני עצמו, ותהליך עדכון הפרמטרים יתייבב. נוסחת העדכון של פרמטרי המודל בכל איטרציה תהיה

$$Parameters = Parameters - \alpha(t) \nabla C$$

כאשר  $t$  הוא מספר ה-**epoch** הנוכחי, ו- $\alpha(t)$  היא פונקציה יורדת כלשהי. דוגמה פשוטה ושימושית היא הקטנת את קצב הלמידה פי קבוע הנבחר מראש, כלומר

$$\alpha(t) = \gamma \alpha(t-1) = \gamma^t \alpha(0)$$

כאשר  $0 < \gamma < 1$  כמובן. הגרף של פונקציית דעיכה זו, עבור  $\gamma = 0.9$  מצוירת להלן.



השימוש בקצב למידה משתנה בספריית PyTorch הוא פשוט ביותר, ומתאפשר בעזרת אובייקט תזמון, `lr_scheduler`. לאחר הגדרת **אובייקט האופטימיזציה**, מעבירים אותו כפרמטר **לאובייקט תזמון קצב הלמידה**, יחד עם פונקציה הקובעת את הדעיכה **בכל צעד**, כלהלן.

```
optimizer = torch.optim.SGD(model, lr=1)

decay = lambda previous_lr: 0.9 ** previous_lr
scheduler = torch.optim.lr_scheduler.LambdaLR(
    optimizer, lr_lambda=decay)
```

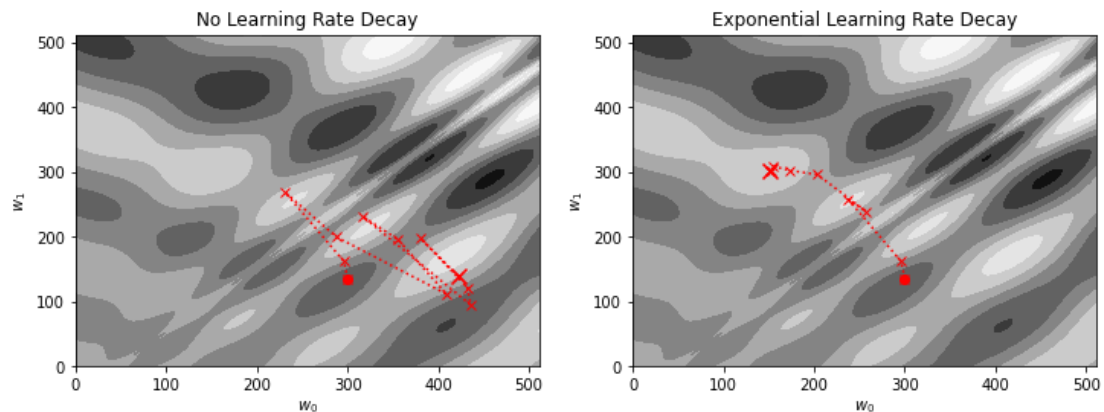
אחרי כן, ללולאת האימון יש להוסיף פקודה המעדכנת את קצב הלמידה **בסוף כל epoch**, זו הפקודה `scheduler.step()` בסוף קטע הקוד הבא.

```

for epoch in range(num_epochs):
    for batch in range(num_minibatches):
        .
        .
        .
        optimizer.step()
        scheduler.step()

```

ראו באיור הבא את ההשפעה של דעיכת קצב הלמידה על ריצת אלגוריתם מורד הגרדיאנט (המלא), כפי שהפעלנו אותו על פונקציית קרטון הביצים בפרק הקודם.



הפונקציה המשמשת לחישוב המסלול מופיעה בקטע הקוד הבא, ויש לשים לב שאנו מפעילים את דעיכת קצב הלמידה לפי ערכו של `decay_flag`, שאת המסלול אנו שומרים במשתנה `history` וזכרו שהמתודה `detach()` מנתקת את המשתנה ממערכת הגזירה האוטומטית.

```

def GD_egg(alpha=5, start=[300., 135.], decay_flag=False):
    w = torch.tensor(start, requires_grad=True)
    model=[w]
    num_epochs=10
    optimizer = torch.optim.SGD(model, lr=alpha)
    if decay_flag:
        decay = lambda previous_lr: 0.9 ** previous_lr
        scheduler = torch.optim.lr_scheduler.LambdaLR(
            optimizer, lr_lambda=decay)

    history = torch.zeros(1+num_epochs, 2)
    history[0,:] = w.detach()

    for epoch in range(num_epochs):
        optimizer.zero_grad()
        cost = eggholder(w[0], w[1])
        cost.backward()
        optimizer.step()
        if decay_flag:
            scheduler.step()
        history[1+epoch,:] = w.detach()

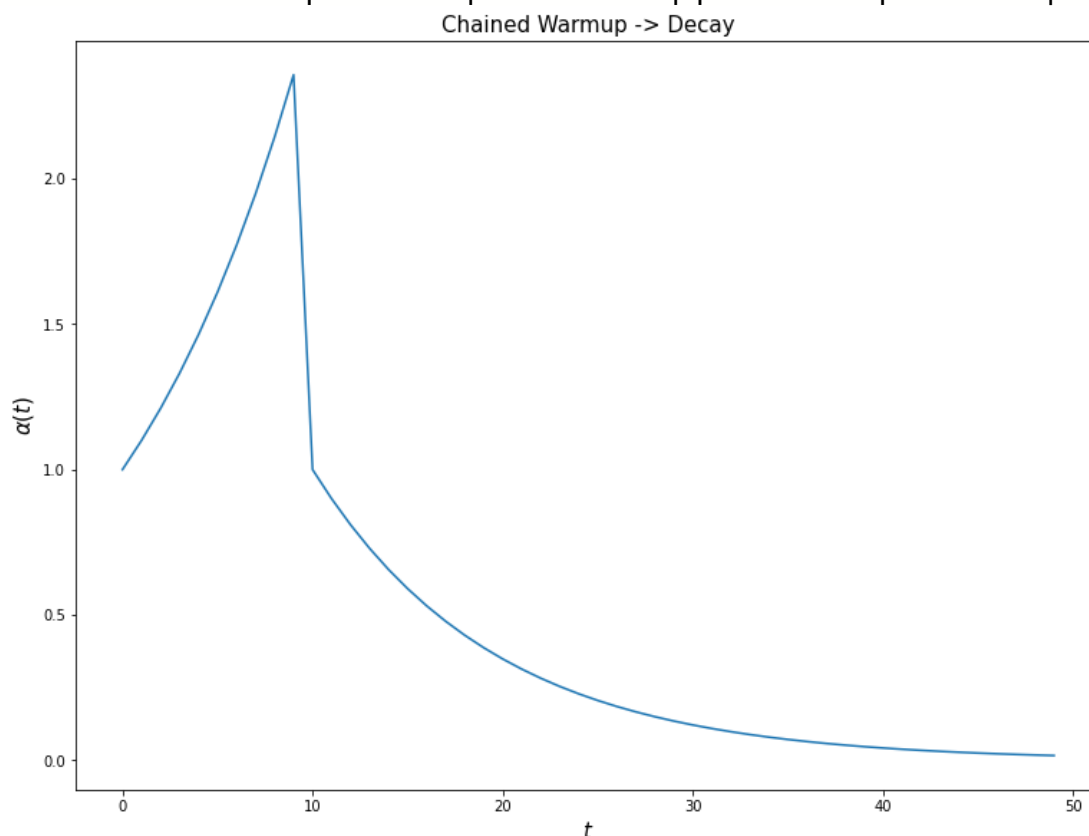
    return history

```

תזמון קצב הלמידה הוא תחום פעיל בחקר רשתות נוירונים עמוקות, שכן לקצב הלמידה השפעה קריטית על ערכי הפרמטרים המתקבלים. הספרייה `torch.optim.lr_scheduler` מספקת מספר רב של אובייקטים אשר בעזרתם ניתן לממש אסטרטגיות שונות, כפי שהתפרסמו בספרות המדעית. היוריסטיקה מעניינת לדוגמה נקראת "חימום" (warmup), והרעיון מאחוריה הוא שבתחילת ריצת האלגוריתם כדאי דווקא להגדיל את קצב הלמידה אחרי כל epoch, בכדי לאפשר לאלגוריתם לחקור אזור גדול יותר במרחב הפרמטרים. אחרי תקופת החימום עוברים לשלב של קצב למידה דועך, לטובת התכנסות האלגוריתם. בקטע הקוד הבא אנו ממשים רעיון זה בעזרת אובייקט המחבר מתזמנים שונים, בזה אחר זה.

```
warmup = lambda previous_lr: 1.1 ** previous_lr
decay = lambda previous_lr: 0.9 ** previous_lr
scheduler1 = torch.optim.lr_scheduler.LambdaLR(optimizer,
        lr_lambda=warmup)
scheduler2 = torch.optim.lr_scheduler.LambdaLR(optimizer,
        lr_lambda=decay)
scheduler = torch.optim.lr_scheduler.SequentialLR(optimizer,
        schedulers=[scheduler1, scheduler2],
        milestones=[10])
```

`SequentialLR` קושר את שני מתזמני קצבי הלמידה, כאשר בפרמטר `milestones` מופיע ה-epoch בו יש להחליף ביניהם. מתזמן קצב הלמידה המתקבל מאויר להלן.



### שאלות לתרגול

- אתרו את נקודות המינימום של פונקציית קרטון הביצים בקרבת הנקודות  $(w_0, w_1) = (400, 130)$  ו-  $(w_0, w_1) = (450, 420)$  בעזרת אלגוריתם מורד הגרדיאנט עם קצב למידה דועך. השוו את התוצאות לאלגוריתם עם קצב למידה קבוע.

2. הוסיפו מתזמן קצב למידה דועך למודל הסיווג של התמונות מאוסף הנתונים Fashion-MNIST והשוו את התוצאות עם ובלי דעיכת קצב הלמידה. שימו לב שדעיכת קצב הלמידה מתבצעת בסוף כל epoch בלבד.