

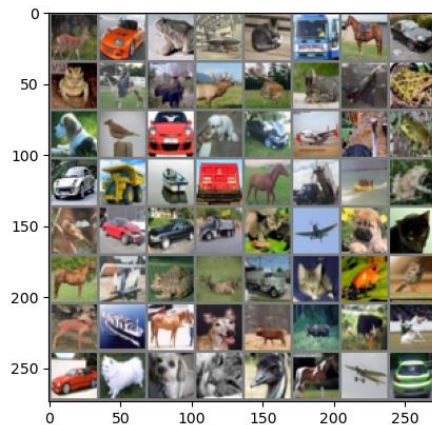
## מטלת בית – רשת CNN על CIFAR10 – איתי גילי

חלק א:

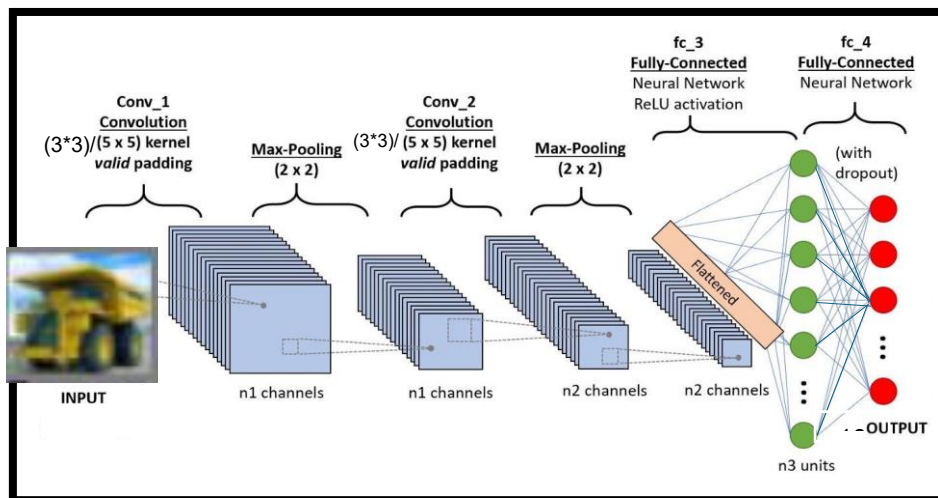
א. כללי - המשימה:

התבקשנו לייצר מודל מיטבי ללמידה עמוקה על מאגר המידע CIFAR10. המאגר מכיל מספר רב של תמונות () בגודל () השייכים ל-10 קבוצות:

```
# CIFAR-10 class names
classes = ['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck']
```

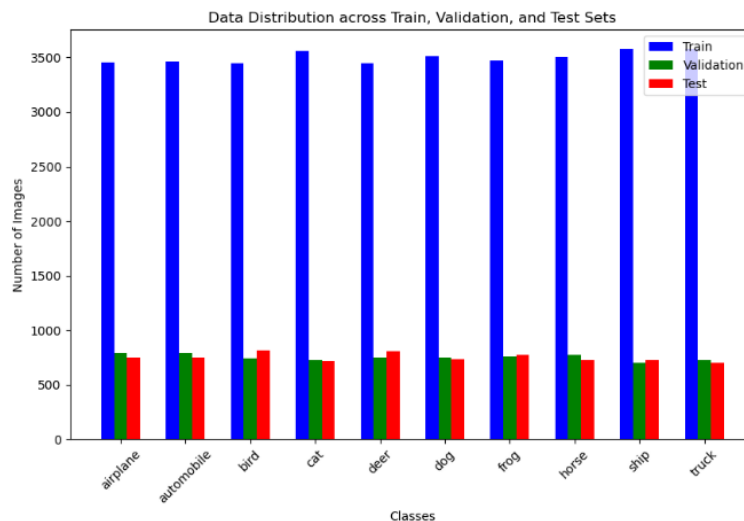


התבקשנו לבצע את המודל באמצעות CNN, המודל מבוסס על הצורה הבאה:



ב. חלוקת הדאטה:

50,000 תמונות, 5,000 לכל סוג, בוצע חילוק לtrain, val and test.



**ג. הנחות עבודה טרם האימון:**

1. עבודה נקייה – בכתובת הקוד לאפשר את מרבית הגמישות בהגדרת המשתנים, למטרת הרצה של hyperparameter בלבד בין מודל למודל. כך, נמנע מהעתקת קוד מיותרת והסדר ישמר בתהליך שיפור המודל.
2. Hyperparameter לאימון- נשנה אותם בצורת ניסויי וטעיה, נשמר לקחים שיעלו יפה.

Hyperparameter	
גודל הקרנל במהלך האימונים יהיה 3*3 או 5*5.	kernel_size
רשימה לfc שתחזיק את גודל השכבות ומס' הנירונים בכל שכבה (מלבד שכתבת ה out שתמיד תהא 10 נירונים).	fc_neurons_list
הסתברות להשתקת נירונים בCNN	dropout_cnn
הסתברות להשתקת נירונים בfc	dropout_fc
קצב למידת הגרדיאנט	lr
גודל הענשה על משקולות גדולות	l2
כמות האפוקים	epochs
מספר של כמה אפוקים נמתין לפני הפעלת Early Stopping.	patience

**3. מניעת OVERFIT :**

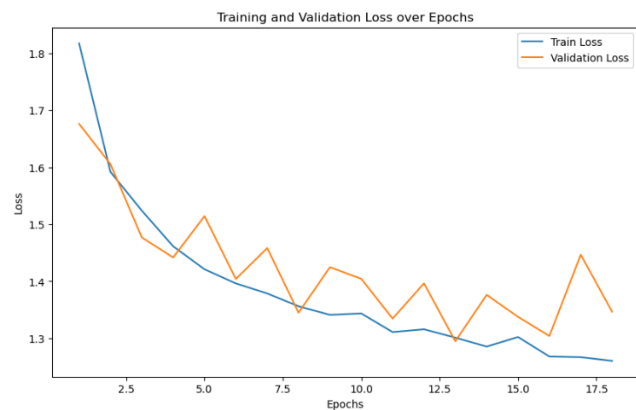
- i. Early Stopping כאשר קצב שיפור הloss ב val קטן ולא משמעותי (והמשקולות ממשיכות להשתנות בגלל הקטנת הloss train).
- ii. הענשת l2 למניעת הסתמכות יתר על משקולות גדולים.
- iii. שימוש בdropout בשכבות הCNN והFC – גורם להשתיק רנדומאלית נירונים ומצמצמת הסתמכות על נירונים דומיננטים.
- iv. שינויים במודל – שינוי גודל ומספר השכבות בfc, כני"ל שינוי בkernel.

## ד. האימון:

### נריץ מודלים עם KERNEL בגודל 5\*5:

#### 1. אימון מס' 1

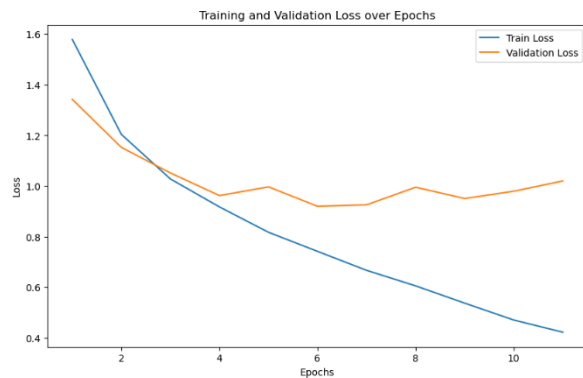
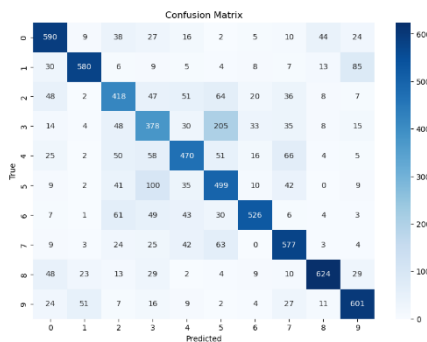
Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout_fc	dropout CNN
0.29	1.3	32	128,64	5	0.001	50	0.01	0	0



המודל סובל מ'רכבת הרים', יתכן שנובע מקצב למידה גדול שלא מאפשר לו להתכנס.

#### 2. אימון מס' 2

Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout_fc	dropout CNN
0.9197	0.7415	32	128,64	5	0.0001	50	0.001	0	0

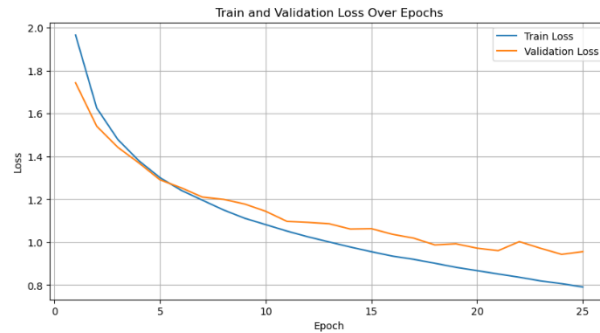


ניתן לראות שהקטנת קצב הלמידה אפשרה שיפור בתוצאה.

### נריץ בעת אימון שלהם מוגדרים 3\*3.

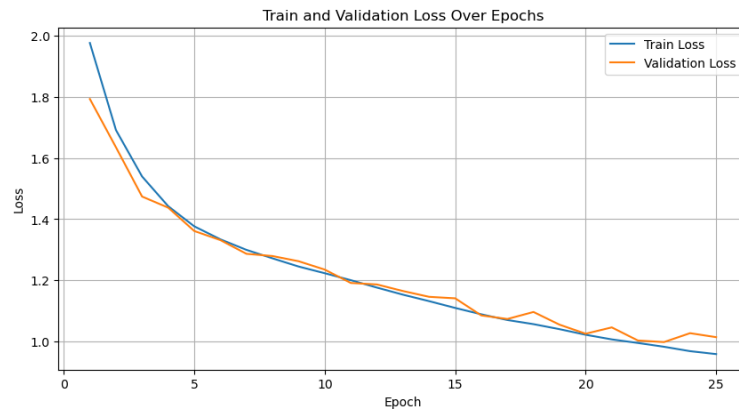
3. אימון מס' 1:

Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout fc	dropout CNN
0.9558	0.7908	64	ללא	3	0.0001	25	0.01	0	0



4. אימון מס' 2

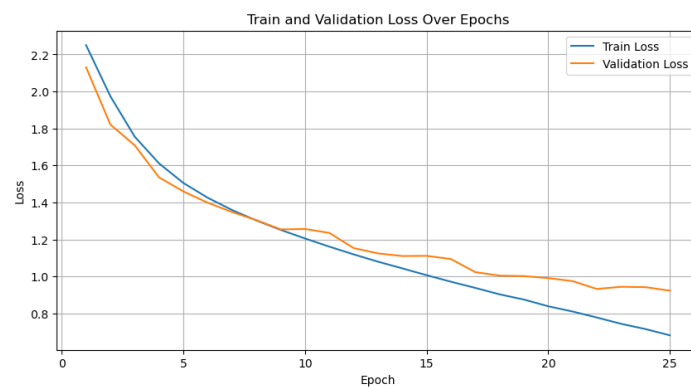
Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout fc	dropout CNN
1.0137	0.9584	64	ללא	3	0.0001	25	0.01	0	0.5



בתצורה הזאת, ה dropout אינו מטיב עם המודל.

5. אימון מס' 3

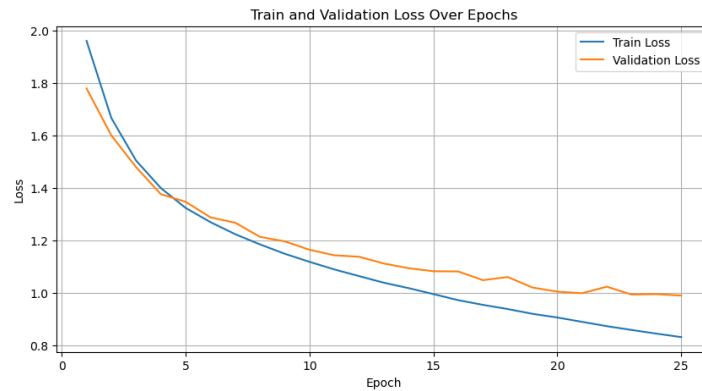
Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout fc	dropout CNN
0.9225	0.6812	64	128, 64	3	0.0001	25	0.01	0	0



הגדלת הfc משפרת את ביצועי המודל.

## 6. אימון מס' 4

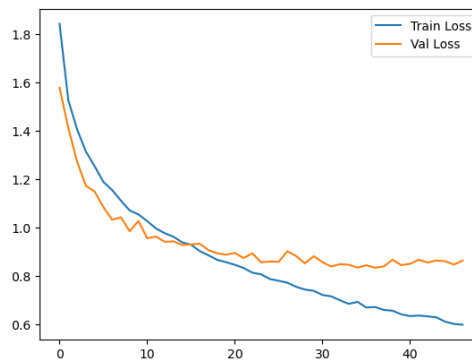
Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout_fc	dropout CNN
0.9891	0.8305	64	ללא	3	0.001	25	0.01	0	0



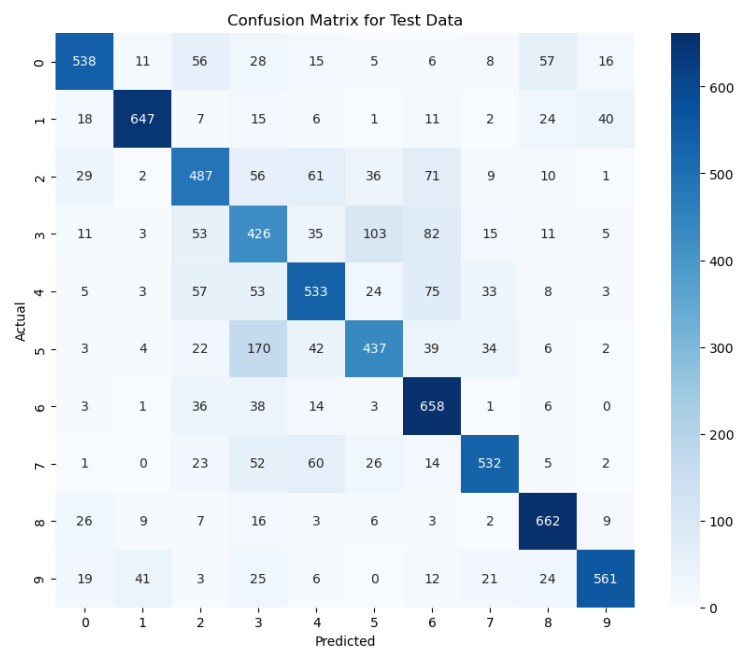
הגדלת 'הענשת' L2 אינה משפרת את המודל.

## 7. אימון מס' 5

Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout_fc	dropout CNN
0.8336	0.6919	64	128, 64	3	0.0001	45	0.001	0.5	0.2



לאחר שהגדלתי את כמות האפוקים למדידת תהליך מעט יותר ארוך, ביצעתי בדיקה על test :



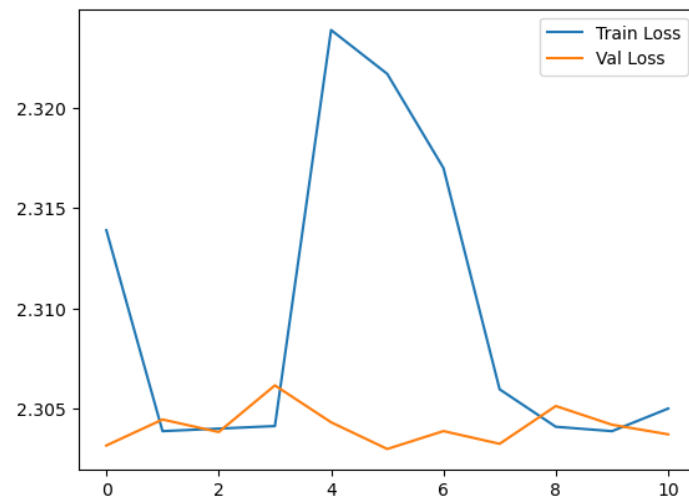
(הוספתי את dropout להימנע מoverfit), במחברת ניתן לראות את המודל הזה כאשר הוא ללא fc עמוקה, התוצאות שלו פחות טובות (0.8599 בולידצה) ואת אותו המודל בbatch\_size קטן יותר (32 לעומת 64 בניסויים ההתחלתיים).

(זהו המודל הטוב ביותר אליו הצלחתי להגיע).

באימון הבא ננסה להגדיל את קצב הלמידה.

## 8. אימון מס' 6

Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout fc	dropout CNN
2.3042	2.3039	32	לא	3	0.0001	50	0.01	0.5	0.2

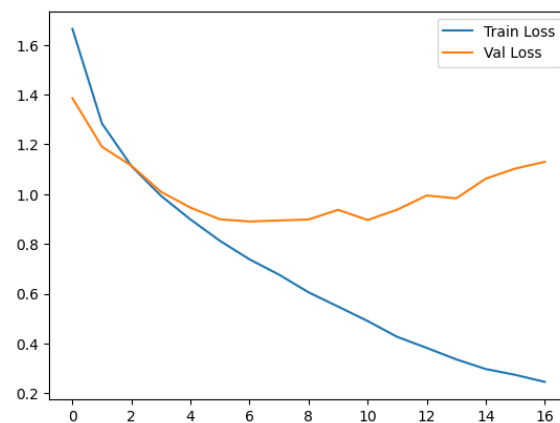


הגדלת קבוע הלמידה יוצר תנודות רחבות במשקולות ומקשה על הלמידה.

ננסה למטב את המודל ולהעמיק את שכבות הfc.

## 9. אימון מס' 7

Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout fc	dropout CNN
1.1541	1.2083	32	128,64,22	3	0.0001	30	0.001	0	0

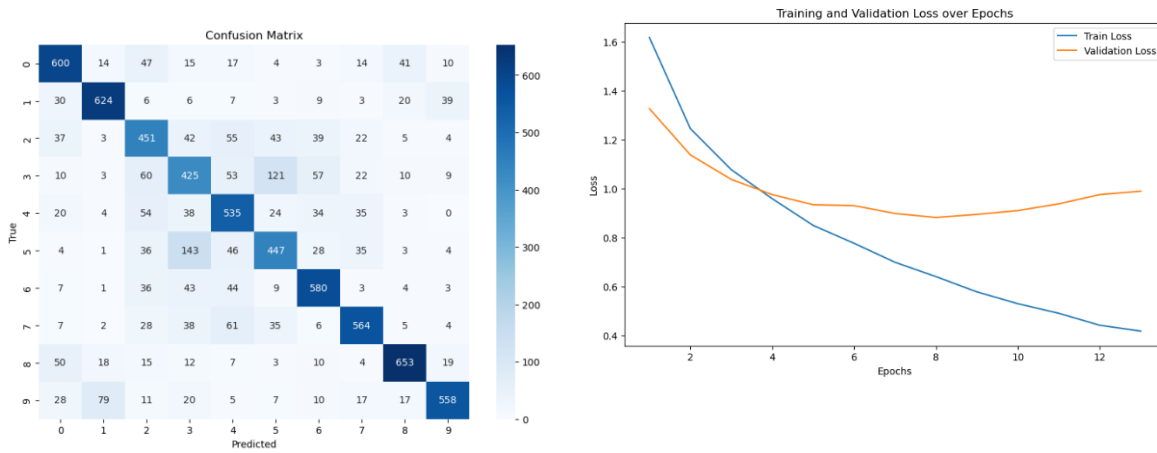


נראה כי הfc המיטבי הוא 128,64.

## 10. אימון מס' 8

## אימון עם מספר נירונים רב בשכבה.

test loss	Val Loss	Train Loss	batch size	fc layers	kernel size	weight decay	epochs	learning rate	dropout fc	dropout CNN
0.9113	0.6396	0.8816	64	512,256	3	0.0001	50	0.001	0.4	0.2



## ה. מסקנות:

- batch\_size גדול – מגדיל את זמן הרצת המודל אבל מאפשר למידה טובה יותר.
- מומלץ להשתמש עם early stopping למניעת ריצות מיותרות באפוקים גדולים.
- FC ממולץ – 128,64 (עם שכבת יציאה של 10 נירונים..)

חלק ב:

$$\frac{\partial g}{\partial a} = \begin{cases} 1 & a > 0 \\ 0.5e^a & a \leq 0 \end{cases} \quad \text{1. כ}$$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial w_2} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial w_2} = (\hat{y} - y)h \quad \text{2. ב}$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial h} \cdot \frac{\partial h}{\partial w_1} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial h} \cdot \frac{\partial h}{\partial w_1} \quad \text{3. ג}$$

$$(\hat{y} - y)w_2 \cdot \frac{\partial h}{\partial w_1} = \begin{cases} (\hat{y} - y)w_2 x & w_1 x + b_1 > 0 \\ \frac{1}{2}(\hat{y} - y)w_2 e^{w_1 x + b_1} x & w_1 x + b_1 \leq 0 \end{cases}$$

$$w_1 = w_1 - \eta \frac{\partial L}{\partial w_1} \quad \text{3. ד}$$

$$w_1 = w_1 - \alpha \sum_{i=1}^n \frac{\partial L}{\partial w_1}(x_i)$$

"hidden layer"  $w_1$  -  $b_1$   $\rightarrow$   $\hat{y}$   
 "output layer" " " -  $b_2$

אם  $w_1$  קטן -  $b_1$  קטן

חלק ג:



א. שימוש ב-CNN (על ידי מסכות הקרנלים רצות על המטריצה ומאגדות את ערכי הפיקסלים הסמוכים אחד לשני) מאפשר שימור של יחסי מיקום הפיקסלים אחד משכנו, ANN לא מבין את תפיסת המרחב וחשיבות סמיכות של פיקסלים אחד לשני (מכיוון שהווקטור לו שומר על הגדרות המיקום שלו לשכניו), הדבר החשוב מאוד בתמונה.

ב. נאמר שיש תמונה של חתול מציץ משמיכה, פעם מציד מצד ימין, פעם מציץ מצד שמאל. שכבות ה-POOLING מאפשרות למצות מידע בתמונה ללא קשר למיקום האובייקט בתמונה, ב-ANN – הרגישות לשינויים גדולה, מיקום האובייקט ישפיע על איך נראה הקטור. מפחית את היכולת לעשות הכללה בעת הלמידה.

ג. רשת CNN מאפשרת חיסכון מפרמטרים לאתחול ולשינוי (בתהליך ה-CNN יש הרבה פחות משקולות מאשר בתהליך ה-FC – שזקוק למשקולות שונה עבודה על נתון שזז מנוירון לנוירון).

2. א. אנחנו רוצים להימנע מoverfit לתמונות הדאטהסט שאנו מעלים לרשת. כאשר אנחנו משתמשים ב-Data augmentation אנחנו יוצרים עוד תמונות שמאפשרות גיוון לדאטה (על ידי שינוי התמונה המקורית – קנה מידה, הזזות וחיתוכים, סיבובים וכו')..

ב.

1. חיתוך אקראי. (Random cropping)

2. היפוך אופקי. (Horizontal flipping)

3. שיבוש צבעים. (Color jittering)