

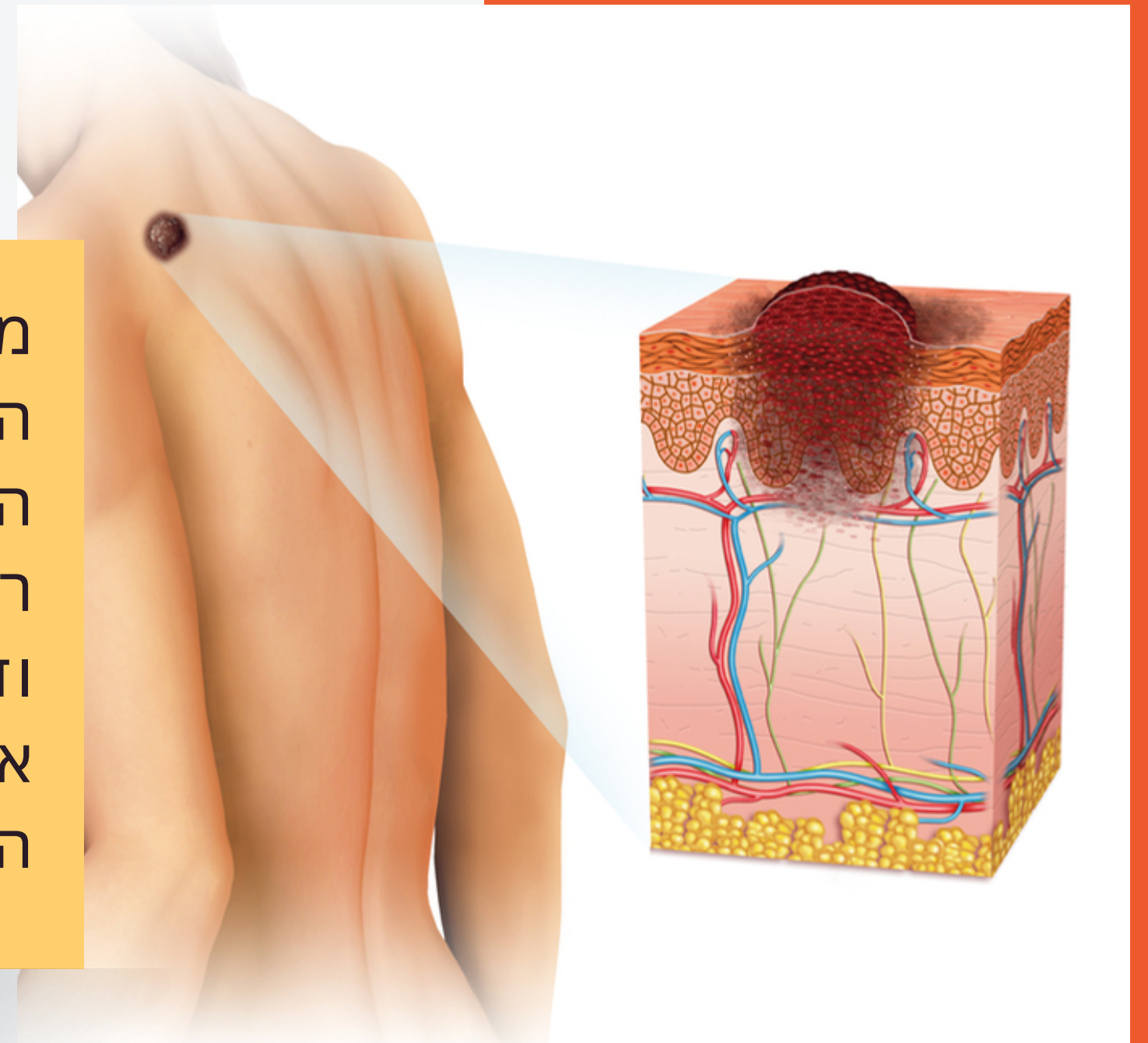


# **CNN MELANOMA CLASSIFICATION**

הילה רבקה הולנד  
אוראל כרמיאל

# המשימה

מלנומה היא סרטן עור קטלני  
הדורש גילוי מוקדם לסיכויי  
הישרדות גבוהים יותר.  
רופאי עור יכולים לשפר את מהירות  
ודיוק האבחנה על ידי מערכת  
אוטומטית שתשתמש בתמונות  
האזורים החשודים ופרטי המטופל.



# האתגרים

## דאטה לא מאורגנת

את הדאטה קיבלנו ב2 תיקיות: train וtest. אך בתוך כל תיקיה לא הייתה חלוקה בין המקריים השפירים (לא סרטניים) לבין הממאירים (סרטניים).

## נתונים בעלי הטייה

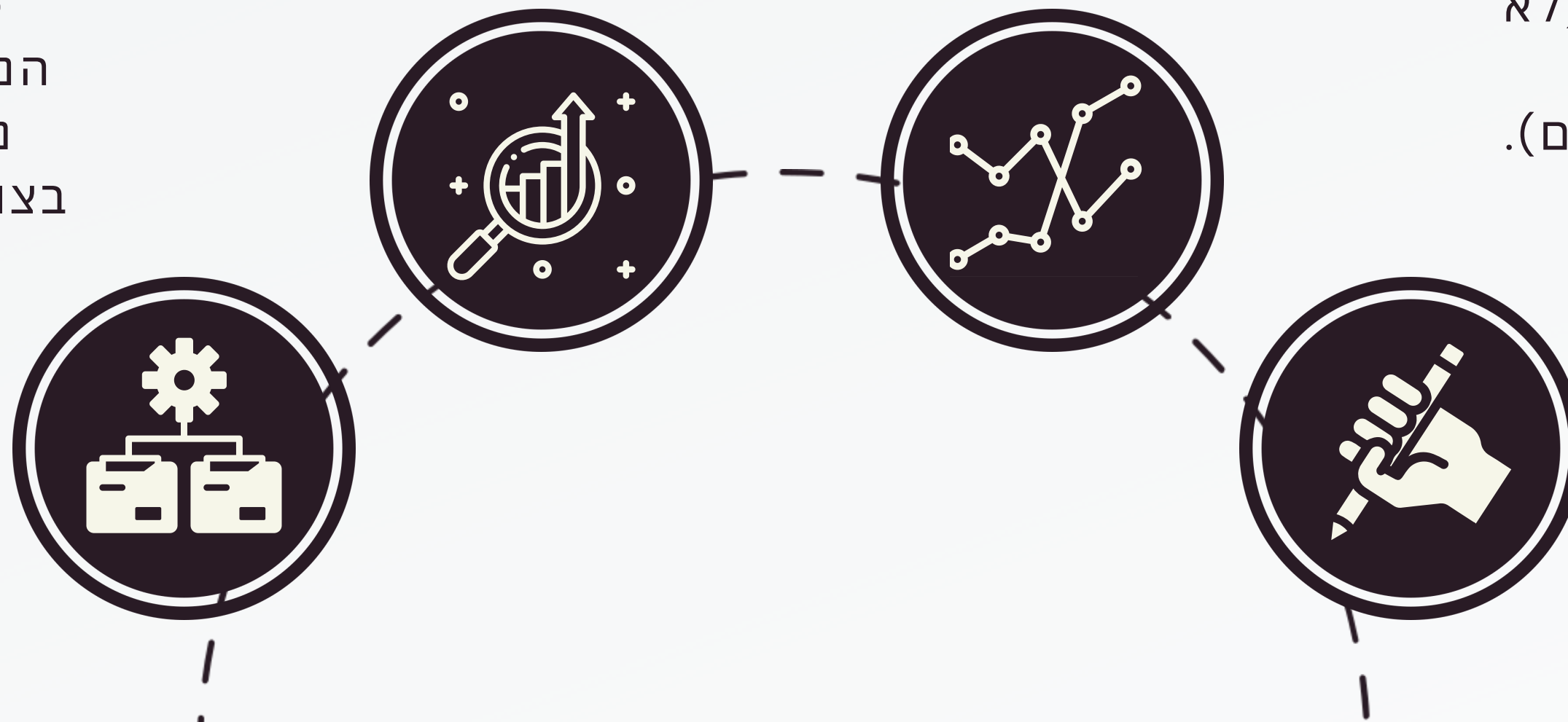
ביחס בין המקרים השפירים (98%) לבין המקרים הממאירים (2%) היה גדול מאוד.

## ריבוי נתונים

כמות התמונות הייתה גדולה (train: 33124) ואיכותם הייתה גבוהה מאוד.

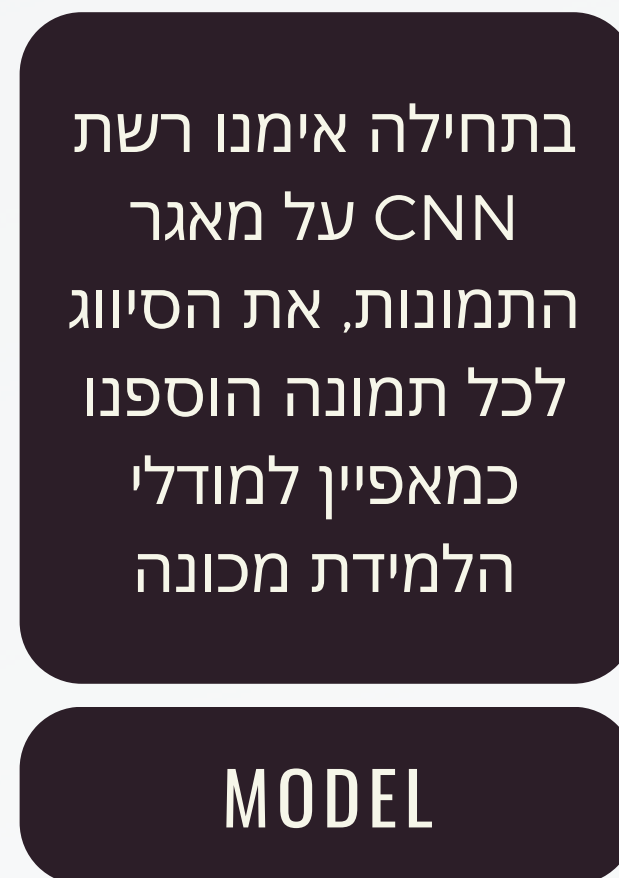
## שילוב סוגי נתונים

הדאטה שלנו הכילה תמונות נגעים בעור מסוג jpg. וכן נתונים רפואיים כדוגמת: גיל ומין הנבדק ואזור הגידול. נתונים אלו הגיעו בצורה טבלאית בקובץ csv.



# השיטה

אימון 2 מודלים שונים במקביל ושילוב ביניהם



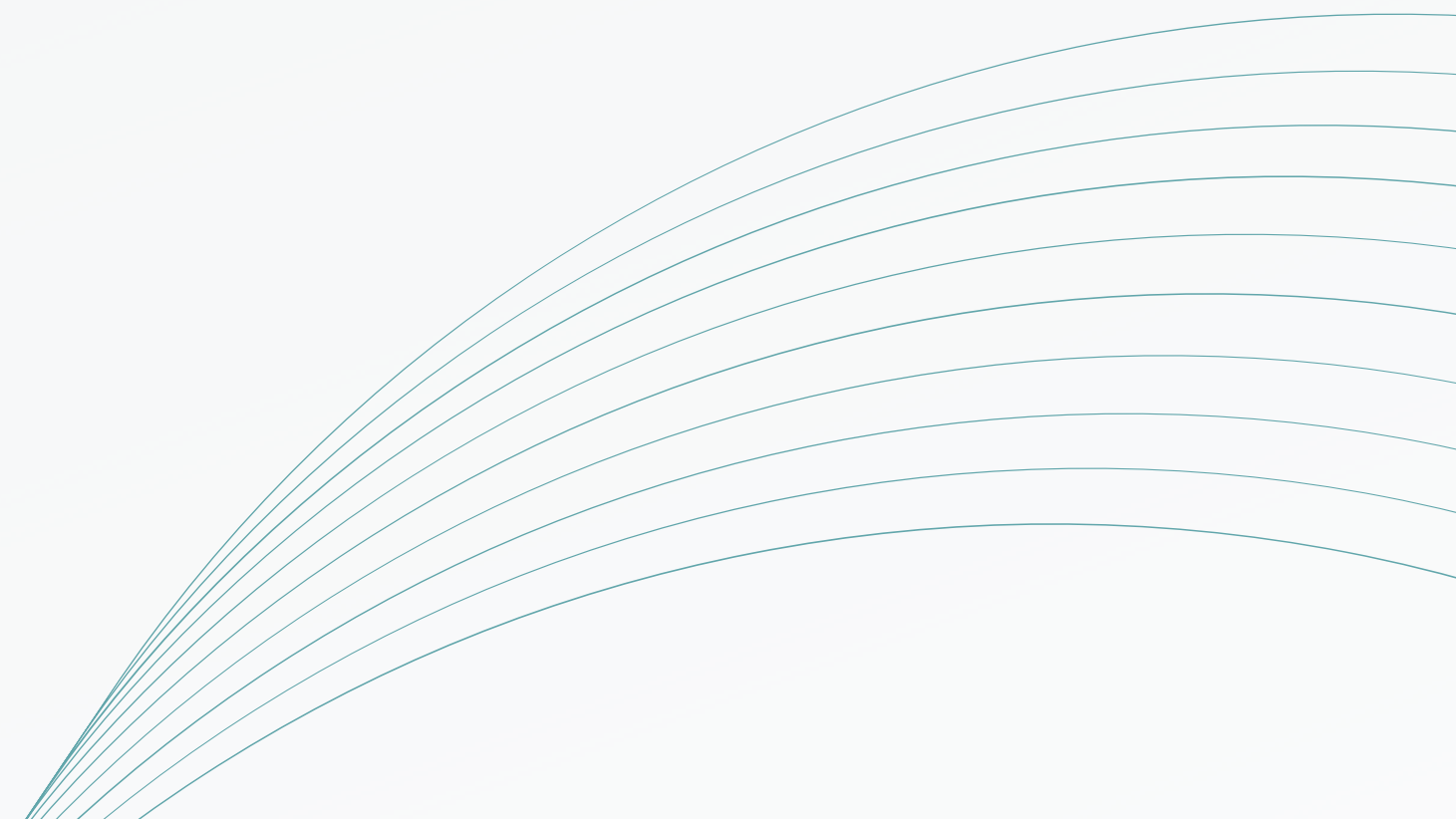


01

---

**עיבוד מקדים**

בעזרת קוד מצאנו את התיוג  
(0/1) של כל תמונה בcsv  
ויצרנו רשימות מתאימות.  
בעזרת הספריות osi shutil  
יצרנו את התיקיות benign  
malignant אליהן העברנו  
את התמונות בכל קטגוריה.  
ע"מ לפתור את בעיית גודל  
התמונות השתמשנו  
בספריה PIL והקטנו את כל  
התמונות.



01

02

## עיבוד מקדים

בעזרת קוד מצאנו את התיוג (0/1) של כל תמונה בcsv ויצרנו רשימות מתאימות. בעזרת הספריות osi shutil יצרנו את התיקיות benign malignant וליהן העברנו את התמונות בכל קטגוריה. ע"מ לפתור את בעיית גודל התמונות השתמשנו בספריה PIL והקטנו את כל התמונות.

## חלוקת הדאטה

ע"מ להתמודד עם ההטיה בין המחלקות נתנו משקל שונה לכל מחלקה בגנרטור התמונות. לקבוצה האימון הכנסנו 30,076 תמונות. ובקבוצת התיקוף שמנו 3050 תמונות (ביחס של 1:6).



01

02

03

## עיבוד מקדים

בעזרת קוד מצאנו את התיוג (0/1) של כל תמונה בcsv ויצרנו רשימות מתאימות. בעזרת הספריות osi shutil יצרנו את התיקיות benign וmalignant אליהן העברנו את התמונות בכל קטגוריה. ע"מ לפתור את בעיית גודל התמונות השתמשנו בספריה PIL והקטנו את כל התמונות.

## חלוקת הדאטה

ע"מ להתמודד עם ההטיה בין המחלקות נתנו משקל שונה לכל מחלקה בגנרטור התמונות. לקבוצה האימון הכנסנו 30,076 תמונות. ובקבוצת התיקוף שמנו 3050 תמונות (ביחס של 1:6).

## בניית מודל

בנינו מודל convolutional neural network המורכב מרשת בסיס מאומנת ומשכבות נוספות בסופו.

# מבנה המודל

Model: "model\_1"

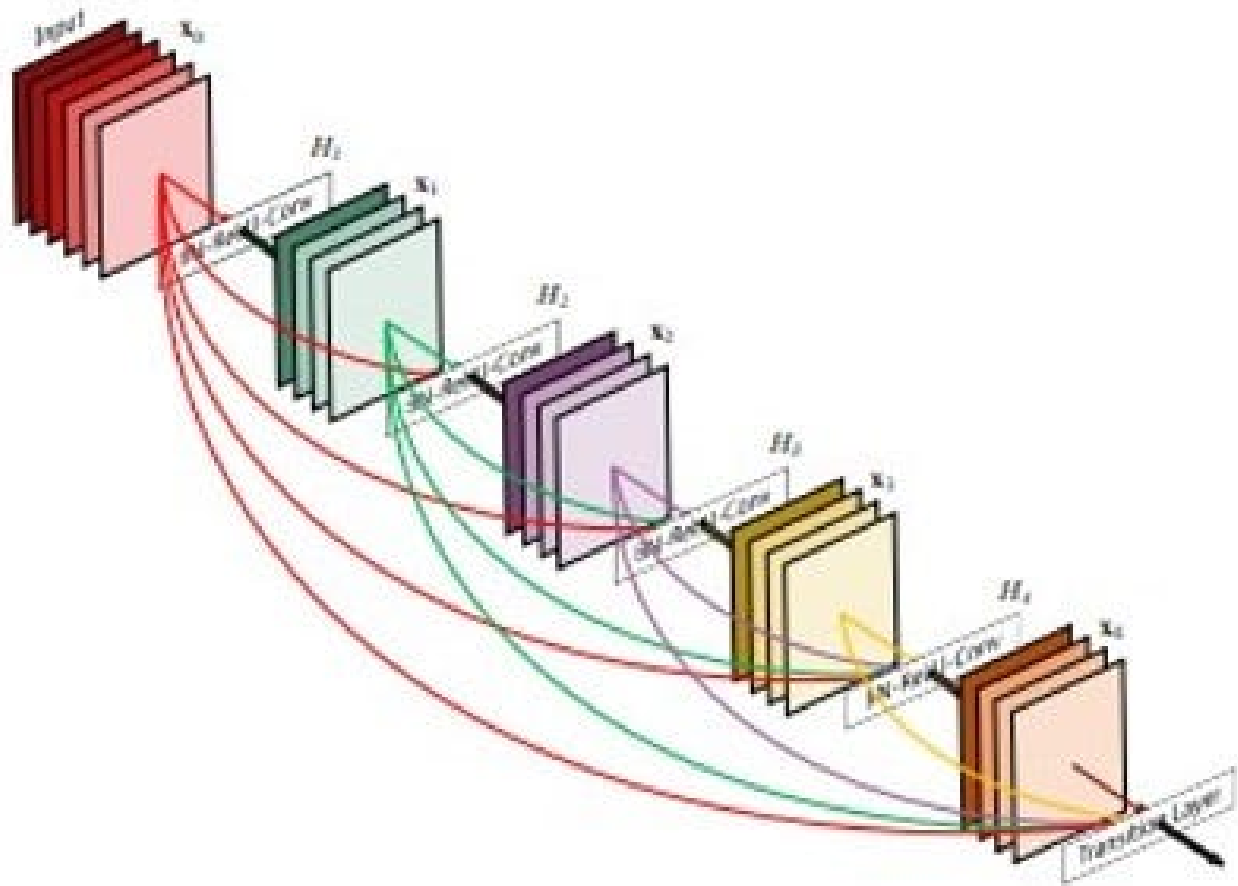
Layer (type)	Output Shape	Param #
=====		
input_4 (InputLayer)	[(None, 150, 150, 3)]	0
densenet121 (Functional)	(None, None, None, 1024)	7037504
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 1024)	0
dense_3 (Dense)	(None, 10)	10250
dense_4 (Dense)	(None, 10)	110
dense_5 (Dense)	(None, 2)	22

=====

Total params: 7,047,886  
Trainable params: 10,382  
Non-trainable params: 7,037,504

01  
02  
03  
04  
05  
OPT  
PERFOR

## BASE MODEL – DENSENET121





# מבנה המודל

Model: "model\_1"

Layer (type)	Output Shape	Param #
=====		
input_4 (InputLayer)	[(None, 150, 150, 3)]	0
densenet121 (Functional)	(None, None, None, 1024)	7037504
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 1024)	0
dense_3 (Dense)	(None, 10)	10250
dense_4 (Dense)	(None, 10)	110
dense_5 (Dense)	(None, 2)	22

=====

Total params: 7,047,886

Trainable params: 10,382

Non-trainable params: 7,037,504

**01**

BASE MODEL – DENSENET121

**02**

GLOBALAVERAGEPOOLING2D

**03**

DENSE (10,ACTIVATION="RELU")

**04**

DENSE (10,ACTIVATION="RELU")

**05**

OUTPUTS – DENSE(2, ACTIVATION='SOFTMAX')

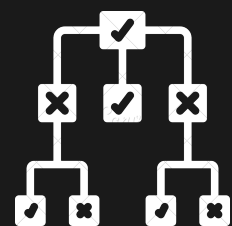
**OPT**

ADAM(0.0001)

**PERFOR**

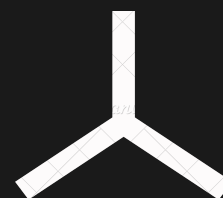
ACCURACY, LOSS, AUC

# ML



AdaBoost הוא טריק למידת מכונה המשלב מודלים פשוטים רבים כדי ליצור חיזוי חזק. הוא מתאים את משקלות הדגימות באימון על סמך הביצועים באיטרציות קודמות כך שהוא לומד ומשתפר עם הזמן.

## ADABOOST



אלגוריתם למידה מפקחת המשמש לניתוח נתונים לסיווג, כשיש יותר משני ממדים של תכונות. מציב את הדוגמאות במרחב וקובע וקטורים שיפרידו אותם לקבוצות.

## SVM



אלגוריתם למידה המשמש לסיווג, הבונה מספר רב של עצי החלטה במהלך האימון, עם שינויים קטנים בסדר התכונות על פיהן נעשות הבחירות בצמתים, ובוחר לסיווג את העץ עם התוצאות הטובות ביותר.

## RF

01

02

03

04

## עיבוד מקדים

בעזרת קוד מצאנו את התיוג (0/1) של כל תמונה בcsv ויצרנו רשימות מתאימות. בעזרת הספריות osi shutil יצרנו את התיקיות benign וmalignant אליהן העברנו את התמונות בכל קטגוריה. ע"מ לפתור את בעיית גודל התמונות השתמשנו בספריה PIL והקטנו את כל התמונות.

## חלוקת הדאטה

ע"מ להתמודד עם ההטיה בין המחלקות נתנו משקל שונה לכל מחלקה בגנרטור התמונות. לקבוצה האימון הכנסנו 30,076 תמונות. ובקבוצת התיקוף שמנו 3050 תמונות (ביחס של 1:6).

## בניית מודל

בניית מודל convolutional neural network המורכב מרשת בסיס מאומנת ומשכבות נוספות בסופו. בניית מודל Random Forest - Ada Boost.

## אימון

התמונות חולקו לBATCH בגודל 16. המודל התאמן במשך 10 EPOCHS כך שבסוף כל EPOCH אם הושג שיפור במדד val\_accuracy המודל נשמר בעזרת פונקציית Checkpoint.

# PERFORMANCE

## LOSS

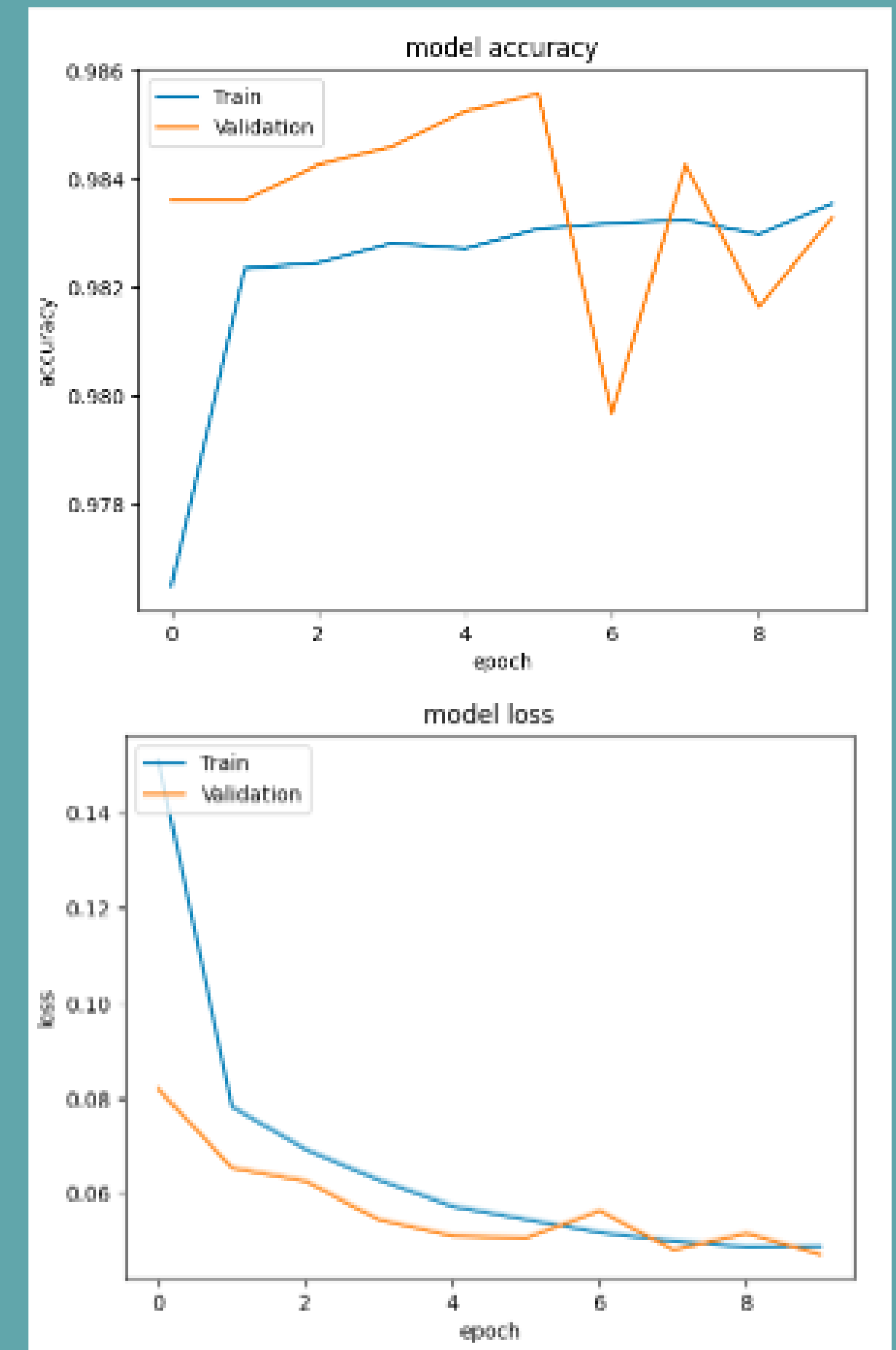
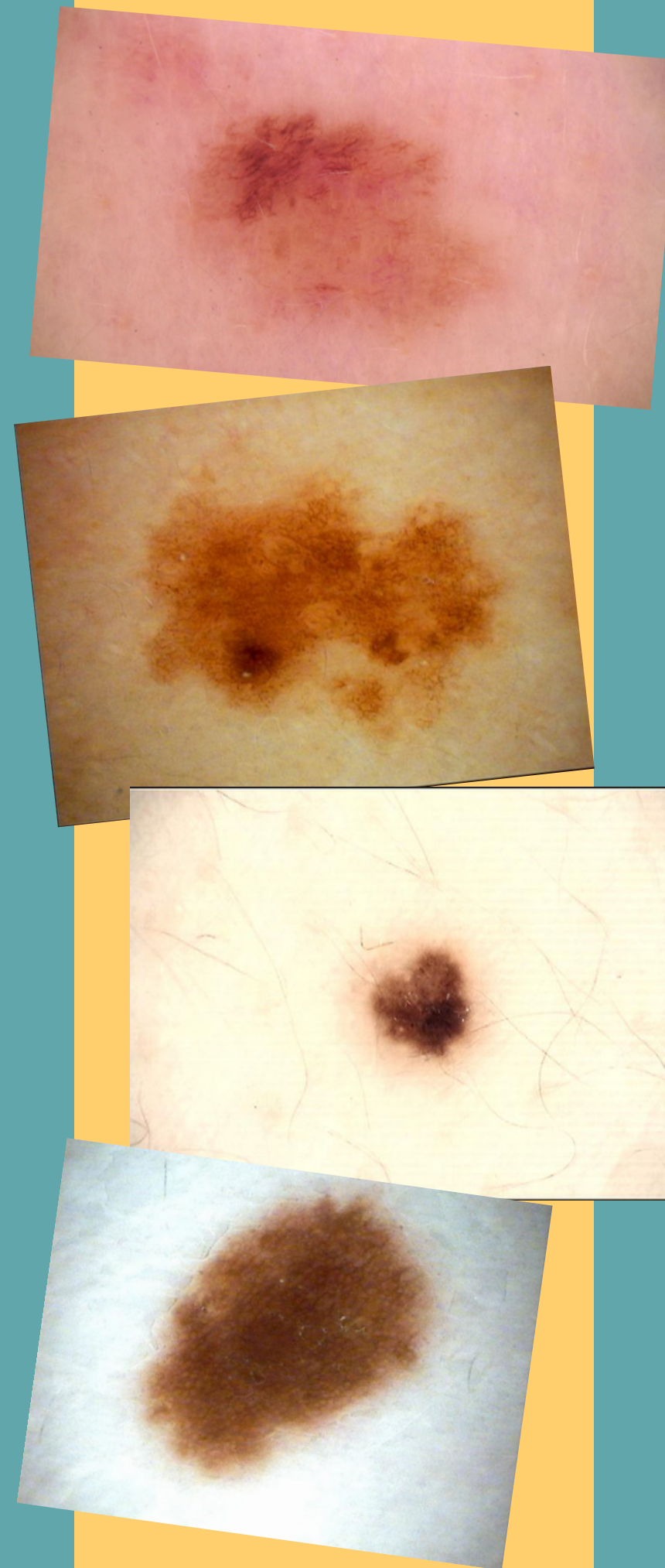
**0.049** train

**0.047** validation

## AUC

**0.998** train

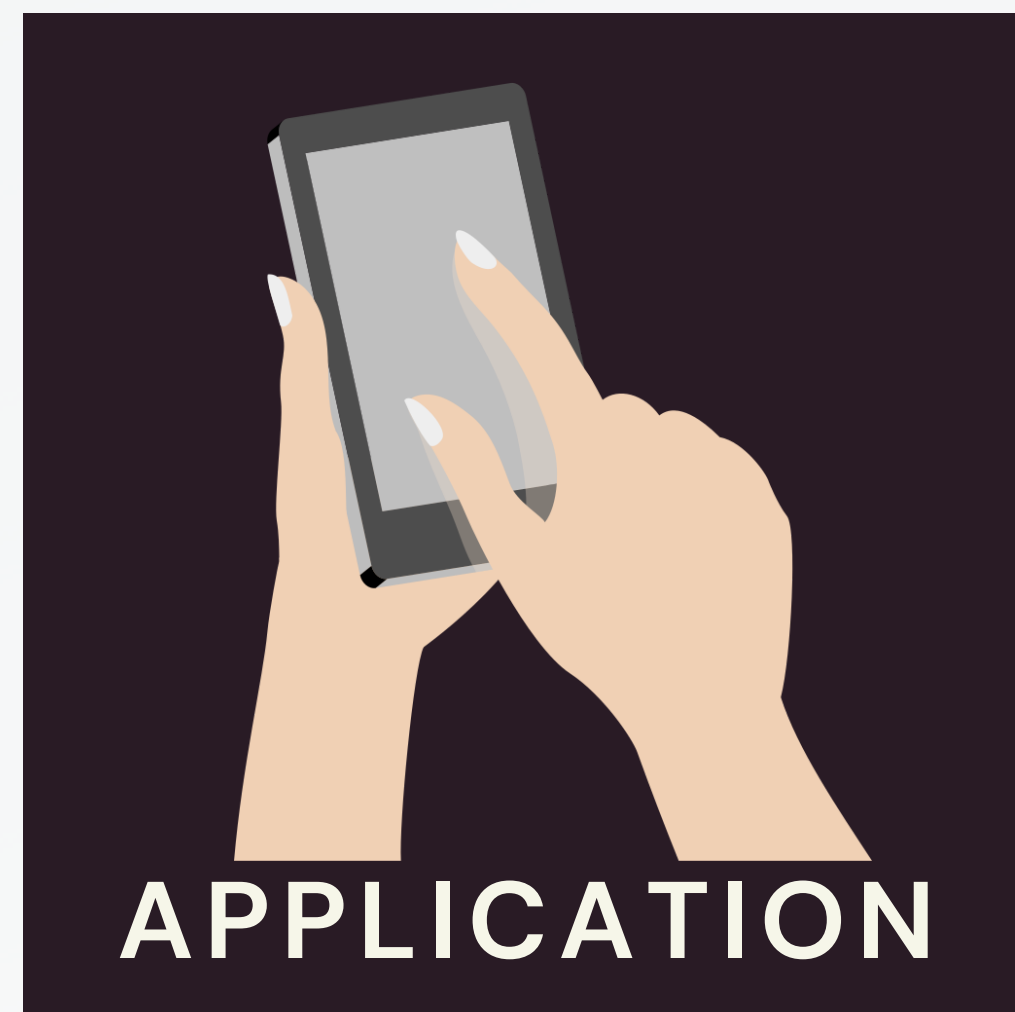
**0.998** validation





# יושום אפשרי

אפליקציה: אדם  
מצלם תמונה בבית  
ואחרי כמה דקות  
מקבל הערכה על  
התמונה שלו



# תודה על ההקשבה

שאלות? נשמח לענות!

