Final Project: Advanced Methods In Medical Image Processing
RSNA Bone Age

מיכל רחימי 316614361 חננאל חדד 313369183

מבוא

הפרויקט עוסק בחיזוי גיל ילדים על בסיס תמונות רנטגן של כף היד. הדאטה שלנו מורכב מכ-15,000 תמונות רנטגן המחולקות ל2 datasets: train set and test set.
על כל התמונות שבdata ביצענו 2 פעולות- הבאנו את כל התמונות לגודל אחיד על מנת שיהיה לרשת קל ללמוד אותן ובנוסף, מרכזנו את התמונות- לכל תמונה היו שוליים שחורים שלא קשורים לתמונת הרנטגן ולכן הורדנו אותם.
אנו מבצעים בפרויקט 2 שיטות ללמידה עמוקה. הראשונה היא רגרסיה, כלומר, חיזוי גילוי עצמות על הtest set תוך כדי שימוש בMLP Regressor של ספריית sklearn. הinput לתוכנית הם תמונות הרנטגן שאותם אנו מכניסים לתוך רשת Encoder. ראשית, אנו מייצרים רשת Auto-Encoder ומאמנים אותה על הtrain set, לבסוף לביצוע הרגרסיה אנו נשתמש רק בחלק של הEncoder. אנו מקבלים כoutput את גיל העצמות של test set וכדי לבדוק האם תוצאות הרגרסיה שלנו מדויקות אנו מייצרים scatter plot, כלומר, אנו מראים קו מגמה בין תוצאות החיזוי לבין התוצאות האמיתיות שאנו מקבלים מהדאטא.
השיטה השנייה היא שימוש ברשת CNN לצורך ביצוע classification multi-class. הרשת כוללת שתי שכבות קונבולוציה כאשר על כל אחת מהן אנו מבצעים max pooling ושתי שכבות fully connected, כאשר השכבה האחרונה כוללת 20 נוירונים כאשר כל נוירון הוא בעצם גיל העצמות המוצג בתמונה- כלומר מחלקה. הdata מורכב מתמונות רנטגן של עצמות ידיים של ילדים בגילאי 0-238 חודשים כלומר בגילאי 0-19 שנים. לכן לדוגמה המחלקה הראשונה היא מי שגילו בין 0-12 חודשים כלומר, עד גיל שנה. המחלקה השנייה היא שגילו בין 12-24 חודשים כלומר, בין גיל שנה לגיל שנתיים. וכך הלאה עד שלמעשה המחלקה ה20 היא מי שגילו בדיוק 238 חודשים כלומר בדיוק בן 19 שנה. לכן הoutput יהיה גיל השנה שהוא הנבדק בתמונה נמצא, כלומר טווח של 12 חודשים שבו הוא נמצא. הinput למודל הוא הtrain set, עליו המודל מתאמן ולבסוף יציג את תוצאותיו על הtest set.
תרומת הפרויקט וחשיבותו הוא שבאמצעות שיטות של למידת מכונה ניתן לחזות את גיל הנבדק בתמונה ובכך להקל בניתוח תוצאות בדיקות רפואיות. הפרויקט משלב שתי טכניקות של למידה עמוקה: רשת CNN ורגרסיה, ושילובן בתוך עולם הרפואה ובכך מראה את היכולות של עולם מדעי המחשב שיכול להשתלב ולעזור בכל תחום בחיים.

Related Work

ביצענו את הסקירה ספרותית על המאמרים הבאים:

1. Automated Bone Age Classification with Deep Neural Networks:
[1] http://cs231n.stanford.edu/reports/2016/pdfs/310_Report.pdf

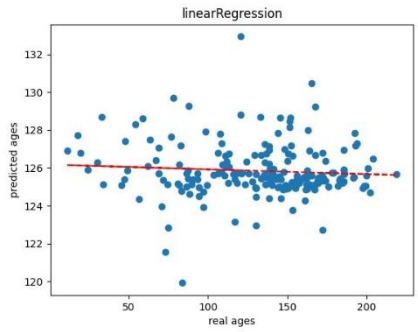
המאמר מתעסק בCNN על מנת לאמן מודל לצורך סיווג גיל עצמות של פציינט בהינתן תמונת x-ray. זו העבודה הראשונה שמשתמשת בשיטות למידה עמוקה, כל העבודות הקודמות התעסקו בסגמנטציה וfeature extraction. המאמר מסביר את המעבר מהשיטה הקודמת לשיטה של למידה עמוקה בעקבות שינויים מתקדמים באפקטיביות של שיטות למידה עמוקה לצורך סיווג תמונות.
המאמר מציג את התוצאות הבאות בשימוש CNN לצורך סיווג תמונות של: המודל הצליח להגיע ל46% ב top one accuracy כלומר, המודל סיווג ב46% מהמקרים את התמונה נכון, בtop two accuracy המודל הגיע ל70% דיוק כלומר, במקרים שבהם המודל לא הצליח לסווג נכון את התמונה אך הסיווג שהתקבל מהמודל היה קרוב מאוד לסיווג האמיתי.
סה"כ המודל במאמר מציג RMSE=1.1 שנים. תוצאות אלו מוצגות עבור הvalidation set.

2. RSNA Bone-age Detection using Transfer Learning and Attention Mapping:
http://noiselab.ucsd.edu/ECE228_2018/Reports/Report6.pdf [2]

המאמר מציג מספר מודלים ללמידת מכונה: מודלים המתאמנים על רגרסיה, מודלים של CNN עם transfer learning, image processing and extraction methods.
לבסוף המאמר מציג מודל יעיל ומדויק ללמידת מכונה עבור הערכת גיל עצמות. המודל המוצע הוא VGG16 עם attention mapping. המודל הגיע לmean absolute error של 9.82/10.75 חודשים לזכר ונקבה. תוצאות אלו תואמות למטרת הפרויקט להוריד את MAE בשנה אחת.

<u>שיטות</u>

א. בשיטת הלמידה לא עמוקה ביצענו רגרסיה לינארית על הdata set. אימנו את המודל לרגרסיה לינארית על הtrain set ולבסוף הרצנו לצורך פרידיקציה על הtest set. נראה בעת את התוצאות: כפי שניתן לראות מקו המגמה (האדום שבתמונה) הפרידיקציה לא נכונה כלל. ניתן לראות כי הגיל האמיתי מתפרס על כל החודשים (0-238) ואילו הגיל שהמודל נותן הוא בין 120-132. כתוצאה מכך אנו מבינים שלמודל היה קשה לבצע רגרסיה לינארית על הdata set של התמונות הנ"ל. אנו מסיקים זאת כיוון שגם כאשר שינינו את ההיפר פרמטרים של המודל (learning rate, optimizer, latent dim, image size) עדיין התוצאות היו לא טובות.

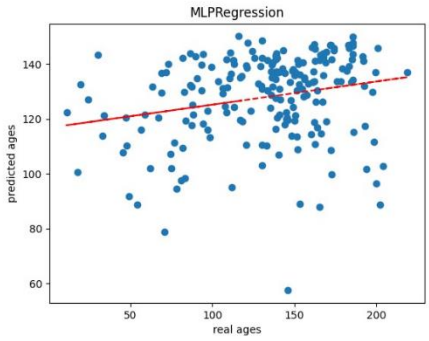


ב. ביצענו שתי גישות ללמידה עמוקה: רגרסיה תוך שימוש בMLP Regressor וmulti class classification תוך שימוש ברשת CNN. MLP Regressor הוא Multi-layer Perceptron regressor הנמצא בספריית sklearn של python. למודל הנ"ל אין שום היפטר פרמטרים- שורת הקריאה שלו היא:

```
<<<regr = MLPRegressor(random_state=1, max_iter=500).fit(X_train, y_train)

<<<regr.predict(X_test)
```

*נלקח מהאתר של sklearn.neural_network.MLPRegressor.
המודל מקבל את הtrain set יחד עם הlabel המתאים לכל תמונה, לומד מנתונים אלו ומבצע רגרסיה על הtest set.
פונקציית הloss במודל זו היא mean square error בין הגיל האמיתי לכל נבדק בתמונה לבין הגיל החזוי שקיבלנו מהמודל.
גם כאן תוצאות הרגרסיה היו לא טובות, למרות שביצענו למידה עמוקה ולא לינארית.
גם כאן אנו רואים שברוב המקרים המודל לא חזה את הגיל הנכון של הנבדק בתמונה (ניתן לראות זאת לפי קו המגמה שהוא איננו אלכסון). אמנם, בשונה מהרגרסיה הלינארית שבה טווח הגילאים החזוי היה בין 120-132 חודשים, כאן אנו רואים כי טווח הגילאים החזוי גדל והוא בין 60-150 חודשים. כלומר, אמנם עדיין לא הגענו לתוצאות מושלמות אך כן אנו רואים שחל שיפור כאשר עברנו מרגרסיה לינארית לרגרסיה רב-שכבתית. למרות זאת אחוזי הדיוק במודל זה לא היו גבוהים במיוחד- 0.12.




```
MLP Regressor accuracy:   0.12
```

מכיוון שהמודל הנ"ל לא היה מספיק יצרנו מודל נוסף שכעת נציג אותו.

המודל השני הוא רשת CNN לצורך ביצוע קלסיפיקציה של 20 מחלקות (כפי שתיארנו במבוא כל מחלקה מתארת טווח גילאים של 12 חודשים). הרשת מורכבת משתי שכבות קונבולוציה, שתי שכבות fully connected ובין כל שכבת קונבולוציה מבצעים max pooling. ההיפר פרמטרים שבחרנו למודל הם: learning rate = 0.01, opitimizer = Adam והרצנו את המודל עם Batch Size = 100 למשך 5 epochs. פונקציית הlossהפעם היא Cross Entropy אשר בקלסיפיציה multi class מחשבת את הLoss עבור כל מחלקה בנפרד וסוכמת את התוצאה. במהלך בחירת ההיפר פרמטרים ניסינו את הoptimizer SGD ואת Adam כאשר גם מקורס למידת מכונה שעשינו וגם עפ"י מידע מהאינטרנט ידענו שה optimizer Adam הוא יותר טוב ואכן במודל שלנו הוא הניב תוצאות טובות יותר מSGD כמצופה. בנוסף, בדקנו לאורך מספר ריצות בדקנו מספר סוגי learning rate כאשר 0.01 נתן במודל שלנו את התוצאות הטובות ביותר. מכיוון שהdata במקרה שלנו הוא מודל גדול וזמן הריצה מאוד ארוך לא רצינו להריץ את המודל למספר אפוקים רב על מנת שזמן הריצה לא יהיה ארוך מדי ולכן בחרנו ב5 אפוקים שאמנם לא הגענו לתוצאות הכי טובות שהמודל יכול להגיע להן אך בצורה כזו המודל הציג תוצאות קרובות יותר לפרויקטים קודמים שנעשו על הdata הנ"ל. כעת נציג ונסביר את התוצאות שקיבלנו בפרויקט שלנו: מכיוון שהaccuracy שלנו גם במודל הנ"ל יצאו נמוכות השתמשנו בשיטה שראינו במאמר הראשון המוצג בסקירה הספרותית שנקרא top-k accuracy, כלומר נציג את הaccuracy לא רק כאשר המודל נתן למחלקה הנכונה של טווח גיל העצמות שתמונה את ההסתברות הכי גבוהה, אלא גם אם המודל החזיר כי המחלקה הנכונה קיבלה את ההסתברות השנייה והשלישית הכי גבוהה, כלומר top-2 accuracy, top-3 accuracy. לצערנו, גם לאחר בדיקת top 3 accuracies עדיין התוצאות היו לא טובות- נראה דוגמה לפלט:

```
test accuracies:   accuracy1=0.115   accuracy2=0.25   accuracy3=0.4
```

הסיבה לאחוזי דיוק כאלו נמוכים הוא שהמודל שלנו מסווג בכל ריצה את כל הפרידיקציות כמחלקה אחת ספציפית בלבד. כלומר, בריצה מסוימת כולם יקבלו פרידיקציה של 13, בריצה אחרת פרידיקציה של 15 וכו'. אנו יודעים זאת מ confusion matrix:

```
confusion matrix:
[[ 0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  2  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  3  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  7  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 10  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 13  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  8  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 15  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 15  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 27  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 30  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 23  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 13  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 14  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 10  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0]]
```

השורות בconfusion matrix מייצגות את התיוגים האמיתיים של הtest set ואילו העמודות מייצגות את הפרידיקציות שהמודל נותן. מכן אנו מבינים שמחלקה הופיעה מספר שונה בdata set שלו אך כולם קיבלו את הפרידיקציה 15. כלומר, מחלקה 0 הופיעה פעם אחת בtest set והיא קיבלה את התיוג 15 לאחר המודל שיצרנו. מחלקה 14 הופיעה 30 פעמים בtest set אך גם היא קיבלה תמיד את התיוג 15 לאחר ריצת המודל שלנו.

ניתוח ואנליזה

א. עפ"י התוצאות שהצגנו בחלק הקודם ניתן להבין שהמודלים של למידה עמוקה הניבו תוצאות נכונות יותר מאשר הרגרסיה הלינארית שביצענו בגישות ללמידה לא עמוקה. כבר בהשוואה מול הMLP Regressor ראינו שהתוצאות היו טובות יותר כיוון שטווח הגילאים החזויים היה רחב יותר מהרגרסיה הלינארית שזהו כבר שיפור משמעותי. למרות שהשיטות ללמידה עמוקה הצליחו יותר מהרגרסיה הלינארית עדיין שני המודלים שהצגנו לא היו טובים במיוחד ולא הציגו אחוזי דיוק גבוהים במיוחד, אנו מניחים (לפי עבודות קודמות והפרויקט הנ"ל) שהדאטא קשה ללמידה. בנוסף, בין השיטות ללמידה עמוקה אנו מסיקים כי הMLP Regressor עבד יותר טוב מכיוון שבניגוד לCNN שחזה תמיד עבור כל הtest set תמיד את אותה מחלקה הMLP Regressor ניבא תוצאות בטווח מסוים של גילאים ולא תמיד את אותו גיל (כפי שציינו הוא לא חזה לאורך כל טווח הגילאים שיש בtest set אלא בטווח קטן יותר). בנוסף הaccuracy של MLP היה מעט יותר גבוה מהCNN 0.12 לעומת 0.115.

ב. המודל שעבד הכי טוב הוא הMLP Regressor כפי שצוין מעלה. קל להסיק כי בין שני סוגי הרגרסיה שביצענו הMLP עבד יותר טוב כיוון שטווח החיזויים שהוציא היה רחב יותר משל הרגרסיה הלינארית- 60-150 לעומת 120-132 ברגרסיה הלינארית. גם בין השיטות של למידה עמוקה הMLP הוציא תוצאות טובות יותר מCNN כיוון שהCNN סיווג את כל ה test set במחלקה אחת ולעומתו הMLP הוציא טווח רחב יותר של גילאים חזויים- הטווח הוא 60-150. למרות שהMLP הוציא את התוצאות הטובות ביותר מבין שלושת המודלים שהצגנו בתוכנית, גם הוא לא הניב אחוזי דיוק גבוהים וניתן לראות זאת מקו המגמה שמתואר בחלק של השיטות. מכיוון שהצגנו כאן מספר שיטות גם ללמידה עמוקה וגם ללמידה לא עמוקה אנו מסיקים כי הדאטא קשה ללמידה וחיזוי, דבר שתואם גם את התוצאות שראינו כי מחקרים ופרויקטים קודמים בוצעו על הדאטא הנ"ל.

את confusion matrix הצגנו בחלק הקודם כאשר ביססנו את טענתנו כי המודל שלנו מסווג את כל הtest set במחלקה אחת. כתוצאה מכך שהסיווג הוא תמיד מחלקה אחת בלבד לא יכולנו ליצור את roc curve כנדרש בהוראות. נצרף את השגיאה שקיבלנו ואת הקוד שבנינו לצורך הרצת הroc curve:

זה הקוד שבנינו ליצירת הroc curve אך אנו לא יכולים לצרף אותו לתוכנית שאנו מגישים כיוון שהוא מוציא שגיאה ועוצר את התוכנית, לכן אנו מצרפים אותו כאן מכיוון שאנו מבינים כי הוא חלק מדרישות הפרויקט.

```
target = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14'
          '15', '16', '17', '18', '19']
# set plot figure size
fig, c_ax = plt.subplots(1, 1, figsize=(12, 8))


# function for scoring roc auc score for multi-class
def multiclass_roc_auc_score(y_test, y_pred, average="macro"):
    lb = LabelBinarizer()
    lb.fit(target)
    y_test = lb.transform(y_test)
    y_pred = lb.transform(y_pred)

    for (idx, c_label) in enumerate(target):
        fpr, tpr, thresholds = roc_curve(y_test[:, idx].astype(int), y_pred[:, idx])
        c_ax.plot(fpr, tpr, label='%s (AUC:%0.2f)' % (c_label, auc(fpr, tpr)))
    c_ax.plot(fpr, fpr, 'b-', label='Random Guessing')
    z = roc_auc_score(y_test, y_test, average=average)
    print('ROC AUC score:', z)

    c_ax.legend()
    c_ax.set_xlabel('False Positive Rate')
    c_ax.set_ylabel('True Positive Rate')
    plt.show()
```

השגיאה שאנו מקבלים:

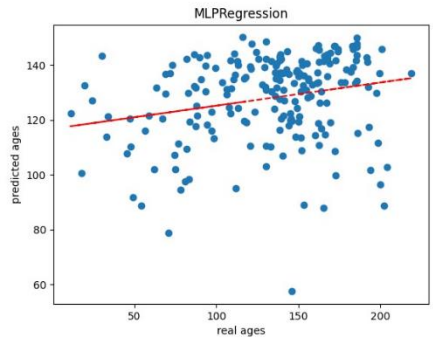
```
ValueError: Only one class present in y_true. ROC AUC score is not defined in that case.
```

כתוצאה מלמידה לא טובה של המודל החיזוי הוא תמיד מחלקה אחת וכפי שמצוין בשגיאה למעלה roc curve לא מוגדר במצב כזה.

נראה את הפלט לשלושת המודלים שיצרנו:

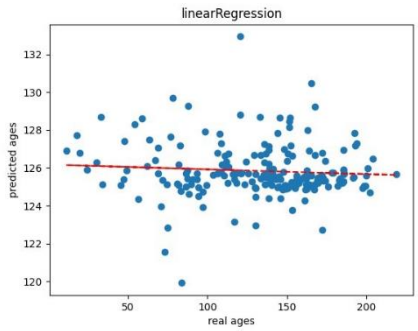
-MLP Regressor

```
MLP regressor MSE:     11.845465804002975
MLP Regressor accuracy:   0.12
```



ניתן לראות את אחוז הדיוק של המודל הנ"ל, הloss בסוף הריצה ואת קו המגמה שיוצרת ההשוואה בין הגיל החזוי לבין הגיל האמיתי של הנבדקים בtest set.

-Linear Regression



מוצג כאן קו המגמה שיוצרת ההשוואה בין הגיל החזוי לבין הגיל האמיתי של הנבדקים בtest set.

-CNN

```
test accuracies:   accuracy1=0.115   accuracy2=0.25   accuracy3=0.4
```

```
confusion matrix:
[[ 0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  2  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  3  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  7  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 10  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 13  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  8  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 15  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 15  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 27  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 30  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 23  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 13  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 14  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0 10  0  0  0  0]
 [ 0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0]]
```

מוצגות כאן top-3 accuracies שלמדנו מהסקירה הספרותית והרחבנו למעלה ובנוסף, הconfusion matrix שמעיד על כך שהמודל שלנו חוזה תמיד לכל הtest set מחלקה אחת לכולם.

כפי שתיארנו, לדעתנו הסיבה העיקרית שהובילה לכישלון שלושת המודלים שלנו היא אופן טעינת התמונות. בשל מגבלות זיכרון הRAM של המחשבים האישיים שאנו משתמשים בהם נאצלנו להקטין משמעותית את התמונות בדאטא הנ"ל ולכן הדבר גורם לאיבוד מידע משמעותי והמודלים שהצגנו לא מצליחים ללמוד כמעט את הדאטא. מידע קודם שיש לנו אנו יודעים כי למידה עמוקה מניבה בדר"כ תוצאות מדויקות יותר ואחוזי ניבוי גבוהות יותר- התוצאות שמוזכרות כאן תואמות זאת כיוון שהMLP שהוא רגרסיה בעלת שכבות הציג תוצאות טובות יותר מהרגרסיה הלינארית. עם זאת, השתמשנו ברשת CNN כיוון שראינו במאמרים שמוצגים בסקירה הספרותית כי רשת CNN הצליחה להניב תוצאות טובות (בהרבה ממה שאנו הצלחנו להראות), אך אנו לא הצלחנו להתקרב לתוצאות שמוצגות במאמרים.

דיון

עפ"י המאמרים שקראנו לצורך הסקירה הספרותית אנו מבינים שלא היה אף מודל עד כה שהגיע לתוצאות דיוק מאוד גבוהות, לכן אנו מסיקים שהדאטא קשה מאוד ללמידה וחיזוי. בנוסף, אנו מניחים שבפרויקט שלנו היו מספר גורמים נוספים שהקשו עלינו כמו: הגדלים הלא אחידים שבהם קיבלנו את התמונות ונאלצנו לצמצם את כולם לאותו גודל שיכול להוביל לאיבוד מידע. בנוסף, כל תמונה הכילה מסגרת שחורה שניסינו ככל האפשר להוריד אותנו על מנת שהמודל יוכל ללמוד יותר טוב את תמונות הרנטגן וסיבה אחרונה היא שהתמונות לא מגיעות בצבע אחיד- חלקן מגיעות בצבע כהה יותר וחלק בצבע כהה פחות, מה שמוסיף למודל דברים נוספים ללמוד שאינם רלוונטיים למה שאנו רוצים שהמודל ילמד.
על מנת להגיע לתוצאות שהגענו אליהם עד כמה ניסינו להוסיף עוד שכבות למודל, לשנות את הlearning rate אך הדבר לא עזר. לדעתנו הדבר העיקרי שמשפיע על התוצאות הן צורת טעינת התמונות שכפי שציינו התמונות אינן באותו גודל ולכן אנו משנים את התמונות לגודל אחיד- ואת כל התמונות זה מקטין משמעותית. במידה ולמחשבים שאנו משתמשים בהם היה יותר RAM היינו מנסים לקלוט את התמונות בגדלים גדולים יותר לראות אם זה ישפיע על אחוזי הדיוק (כאשר ניסינו לקלוט את התמונות בגדלים גדולים יותר קיבלנו שגיאות כי הזיכרון בRAM איננו מספיק), אך כפי שכבר ציינו גם במאמרים אחוזי הדיוק לא היו גבוהים מאוד. לסיכום, לפי הסקירה הספרותית ותוצאותינו אנו מסיקים כי הדאטא קשה ללמידה וחיזוי.

ביבליוגרפיה

[1] Automated Bone Age Classification with Deep Neural Networks. Matthew Cohen, Stanford University.

[2] RSNA Bone-age Detection using Transfer Learning and Attention Mapping. Juan Camilo Castillo, Yitian Tong, Jiyang Zhao, Fengcan Zhu.

קישורים לdata

[1] https://www.kaggle.com/kmader/rsna-bone-age

[2] https://stanfordmedicine.app.box.com/s/4r1zwio6z6lrzk7zw3fro7ql5mnoupcv