**מערכת סיווג תמונות ECG - הסבר כללי ופירוט טכני**

המערכת שפיתחנו נועדה לסווג תמונות **ECG** (אלקטרוקרדיוגרם) לפי קטגוריות מוגדרות, תוך שימוש בטכנולוגיות של **למידת מכונה (בינה מלאכותית)** ומודלים של **רשתות נוירונים עמוקות (CNN)**. המטרה היא ליצור מערכת שתוכל לסווג תמונות בצורה מדויקת, להסביר את ההחלטות שלה בצורה שקופה, ולשפר את הדיוק ככל שנוסיף עוד נתונים לאימון.

**מה יש במערכת?**

המערכת מבוססת על מודל רשת נוירונים בשם **ResNet18** שמומש באמצעות ספריית **PyTorch** (עוד על כך בהמשך). המודל אומן על סט נתונים קטן יחסית, אך למרות זאת הצלחנו להשיג דיוק ממוצע של **60%** – תוצאה טובה בתנאים של מעט נתונים. המערכת גם כוללת מנגנון להסברים ויזואליים על ההחלטות באמצעות **Grad-CAM** (מפת חום), שמספקת תובנות לאילו חלקים בתמונה המודל התייחס כשהוא ביצע את הסיווג.

**מדוע הספריות שנבחרו?**

1. **PyTorch**:
   * ספרייה פופולרית ללמידת מכונה, שמאפשרת בניית מודלים עמוקים בקלות וגמישות.
   * היא מספקת כלים לאימון מודלים גדולים כמו **ResNet18** על נתונים מורכבים כמו תמונות.
2. **Torchvision**:
   * חלק מ-PyTorch, היא מספקת גישה למודלים מבוססי תמונות כמו **ResNet18** וגם טרנספורמציות מוכנות מראש שמאפשרות לעבד תמונות לפני האימון, כמו שינויי גודל, סיבוב וכו'.
3. **Grad-CAM**:
   * כלי שעוזר להסביר את המודל בצורה ויזואלית, בכך שהוא מציג מפת חום שמראה אילו חלקים בתמונה היו חשובים למודל בקבלת ההחלטות. כך המשתמשים יכולים להבין את ההיגיון מאחורי הסיווגים.
4. **Matplotlib**:
   * ספרייה שמאפשרת יצירת גרפים להצגת התוצאות והביצועים של המודל במהלך האימון והוולידציה.

**איך נמנענו מ-Overfitting?**

**Overfitting** הוא מצב שבו המודל לומד את הנתונים באימון "בצורה יותר מדי טובה", כלומר הוא מצליח להתאים בדיוק רב לסט האימון, אך לא מצליח להכליל את הידע לנתונים חדשים. כדי להימנע מבעיה זו, השתמשנו בכמה טכניקות:

1. **Dropout**:
   * זוהי שיטה שבה "מכבים" חלק מהנוירונים במודל במהלך האימון, מה שמכריח את המודל ללמוד על הנתונים בצורה כללית יותר ולא להסתמך על נוירונים ספציפיים.
2. **L2 Regularization**:
   * רגולריזציה L2 מוסיפה עונש קטן לערכים גדולים מדי של משקלי המודל, מה שמונע מהמודל ללמוד רק תבניות ספציפיות מהנתונים ומשפר את היכולת שלו להכליל לנתונים חדשים.
3. **העשרת הנתונים (Data Augmentation)**:
   * השתמשנו בשינויים אקראיים בתמונות כמו סיבוב, שינוי גודל, ושינוי צבע, כדי להעשיר את סט הנתונים שלנו. כך המודל ראה גרסאות שונות של התמונות הקיימות, וזה עזר לו ללמוד להכליל טוב יותר גם על תמונות שלא נחשף אליהן באימון.

**שיפור המודל בעתיד**

המודל שלנו אומן על **סט נתונים קטן יחסית** – מדובר בקבוצה קטנה של תמונות שמוגדרות מראש. המטרה העתידית היא להגדיל את **סט הנתונים** ולספק למודל יותר תמונות לאימון. ככל שנוסיף עוד תמונות לסט האימון, נוכל לשפר את ביצועי המודל, ולהעלות את הדיוק לרמות גבוהות הרבה יותר.

המערכת שלנו בנויה כך שניתן לעדכן אותה בקלות עם נתונים נוספים, ובכך לאמן את המודל שוב על בסיס המידע החדש ולהשיג תוצאות מדויקות יותר.

**הסבר הגרף ששלחת**

הגרף שהצגת לנו משקף את **הביצועים של המודל במהלך האימון והוולידציה**:

1. **Loss (הפסד)**:
   * גרף זה מציג את ההפסד, כלומר את המדד שבו המודל שלנו מודד כמה רחוקות התחזיות שלו מהערכים האמיתיים. ההפסד של האימון (Train Loss) ושל הוולידציה (Validation Loss) יורד ככל שהמודל לומד. בהתחלה, ההפסד גבוה מאוד, אך בהמשך הוא יורד משמעותית, מה שמעיד על כך שהמודל לומד את הנתונים בהדרגה ומשתפר.
2. **Accuracy (דיוק)**:
   * גרף הדיוק מציג את הביצועים של המודל במונחי אחוזי דיוק. ניתן לראות שכבר אחרי מספר אפוקים (סיבובים של אימון על כל הנתונים) המודל הגיע לדיוק גבוה מאוד בוולידציה, סביב 90% ומעלה. הדבר מראה שהמודל למד היטב את הדפוסים המרכזיים בתמונות, ומסוגל לבצע סיווגים מדויקים.

**שימוש באימון ובוולידציה**

כדי להבטיח שהמודל לא לומד רק על הנתונים שהוא ראה באימון, חילקנו את הנתונים לשתי קבוצות:

1. **סט האימון (Training Set)**:
   * זוהי קבוצת הנתונים שעליה המודל לומד. המודל רואה את התמונות בסט הזה שוב ושוב במהלך האימון.
2. **סט הוולידציה (Validation Set)**:
   * זוהי קבוצה נפרדת של נתונים שהמודל לא רואה במהלך האימון, והיא משמשת לבדוק עד כמה המודל מצליח להכליל את מה שהוא למד גם לנתונים חדשים.

הדיוק וההפסד שמתקבלים מהוולידציה הם המדדים האמיתיים לביצועים של המודל, כי הם מייצגים את היכולת של המודל לעבוד עם נתונים שהוא לא ראה קודם.

**סיכום**

המערכת שלנו נבנתה תוך שימוש בספריות מתקדמות כמו PyTorch ו-Torchvision, והצלחנו להימנע מ-overfitting בזכות טכניקות כמו Dropout ורגולריזציה L2. נכון לעכשיו, המודל שלנו הגיע לדיוק ממוצע של 60% על סט נתונים קטן יחסית, אך הוא מוכן לשיפור משמעותי ככל שנוסיף עוד נתונים ונאמנו אותו מחדש.

היכולת של המערכת להסביר את הסיווגים באמצעות **Grad-CAM** הופכת אותה לשקופה ולמובנת יותר, ומאפשרת למשתמשים להבין מהיכן נבעו ההחלטות של המודל. ככל שנמשיך להגדיל את סט הנתונים ולאמן את המודל מחדש, אנחנו מצפים לשפר את הדיוק לרמות גבוהות בהרבה.