המחלקה למתמטיקה ולמדעי המחשב

האוניברסיטה הפתוחה

עבודה מסכמת זו הוגשה כחלק מהדרישות לקבלת תואר

"מוסמך למדעים" M.Sc. במדעי המחשב

האוניברסיטה הפתוחה

המחלקה למתמטיקה ומדעי המחשב

חיזוי ערך מניות באמצעות בינה מלאכותית

Stock value prediction using artificial intelligence

על-ידי

שיר בינשטוק (אובסטלר)

201354404

העבודה הוכנה בהדרכתו של ד"ר לאוניד ברנבוים

## תוכן עניינים

תקציר 3

1. רקע 4
2. תיאור הפרויקט וסביבת העבודה 6
3. איסוף ועיבוד המידע 8

3.1 סקירת מחקרים קודמים ואיסוף המידע 8

3.2 הוספת פרמטרים נגזרים 11

3.3 מדדים חיצוניים 17

3.4 סינון וארגון המידע 19

4. חלוקת המידע והכנת המידע למודל 23

4.1 טרנספורמציית נתונים בשיטת חלון הזזה 23

4.2 חלוקת הנתונים לאימון, ולידציה ובדיקה 24

5. מודל LSTM 26

5.1 בחירת המודל 26

5.2 מימוש המודל 27

5.3 כוונון היפר פרמטרים לשיפור המודל 30

6. הערכת ביצועים של המודל 34

7. סיכום והצעות למחקר עתידי 38

8. מקורות 39

## תקציר

בפרוייקט זה אני מתמקדת בחיזוי מחירי מניות על ידי שימוש ברשתות נוירונים נשנות מסוג LSTM, עם מטרה להבין טוב יותר את דינמיקות השוק ולבדוק אפשרויות לפיתוח מודלים איכותיים ככל הניתן לחיזויים פיננסיים. רשתות LSTM מוכרות ביכולתן לעבד ולחזות נתוני ציר זמן באופן אפקטיבי, הודות למבנה המיוחד שלהן שמאפשר לשמור מידע לטווחים זמניים ארוכים וקצרים.

הפרוייקט התחיל באיסוף ועיבוד מסד נתונים גדול של מחירי מניות, כולל מידע היסטורי על תנודות מחירים, נפחי מסחר וגורמים חיצוניים אשר יכולים להשפיע על המחירים. פרט לחיזוי מחירי המניות, המודל כלל גם ניתוח של מדדים כלכליים ופיננסיים נוספים כדי להבין אילו גורמים משפיעים באופן משמעותי על השווקים.

במהלך הפרוייקט, השתמשתי בטכניקות עיבוד מתקדמות לנתונים כגון סטנדרטיזציה, מפת חום לניתוח קשרים בין הנתונים, טיפול בנתונים חסרים ועוד. שלב העיבוד המקדים היה קריטי לצורך שיפור ביצועי המודל, מכיוון שנתונים נקיים ומעובדים נכון מאפשרים למודל ה-LSTM ללמוד בצורה יעילה יותר ולחזות את התנודות במחירי המניות בדיוק גבוה יותר.

לאחר מכן בוצע כוונון היפר פרמטרים על מנת לשפר את המודל ולבחור פרמטרים שיביאו את המודל לביצועים טובים יותר.

לסיכום, עבודה זו מציעה נסיון לפתח מודל חיזוי למחירי מניות שיכול להוות כלי עזר למשקיעים ולאנליסטים בשוק ההון, תוך שימוש בכלים טכנולוגיים מתקדמים וניתוח נתונים עמוק. המחקר מצביע על הפוטנציאל להמשך פיתוח ומחקר בתחום, על מנת להבין טוב יותר את המנגנונים המובילים לתנודות במחירי מניות ולשפר את דיוק החיזויים בעתיד.

## רקע

### **תהליך חיזוי בלמידת מכונה**

בלמידת מכונה, חיזוי מתבצע על ידי מודל מתמטי שנועד ללמוד מנתונים ולחזות את התוצאה עבור נתונים חדשים. התהליך מתחיל בבניית מודל, שבו משתמשים בנתוני אימון כדי להבין דפוסים ולחזות תוצאות. המרכיב המרכזי בתהליך זה הוא פונקציית העלות, שמטרתה למדוד את רמת הטעות של המודל ביחס לנתונים האמיתיים.

פונקציית העלות משקפת את ההבדל בין הערכים שהמודל חזה לבין הערכים האמיתיים. מטרת האימון היא למזער את ערך פונקציית העלות, כלומר להקטין את השגיאה בחיזויים. לדוגמה, במשימות רגרסיה נפוצה השימוש ב-Mean Squared Error, שהיא ממוצע הריבועים של ההפרשים בין הערכים הנחזים לאמיתיים.

השלב הבא הוא עדכון המשקולות וההטיות של המודל בהתאם לגרדיאנט (נגזרת) של פונקציית העלות. שיטה נפוצה לביצוע כך היא שיטת ירידה בגרדיאנט (Gradient Descent), שבה מחשבים את נגזרת פונקציית העלות ביחס לכל משקולת ומעדכנים את המשקולות בכיוון שמקטין את פונקציית העלות. ההטיות מתואמות באופן דומה לפי השגיאה שנמדדה.

התהליך אינו מסתיים במציאת פתרון אופטימלי אחד, אלא בבדיקה ושיפור מתמשכים של המודל. שיטות כמו חלוקת הנתונים לסט אימון וסט בדיקה, כיוון היפרפרמטרים ושימוש בטכניקות רגולריזציה עוזרות למנוע התאמה יתר ולהבטיח שהמודל יעיל גם עבור נתונים שלא נראו במהלך האימון.

**מהם רשתות נוירוניות**

עם התפתחות עולם הטכנולוגיה, כלים חדשניים לניתוח וחיזוי מידע נהפכים למרכיב חיוני בכלכלה המודרנית. במיוחד בתחום הפיננסי, יכולת לחזות מגמות ולזהות תבניות בשוק ההון היא קריטית. אחת מהטכנולוגיות המרכזיות שמשמשות למטרה זו היא רשתות נוירוניות.

רשתות נוירוניות מבוססות על רעיון הלמידה העמוקה, שהיא תת-תחום בתוך למידת מכונה. הן מכילות שכבות של נוירונים, כאשר כל נוירון בשכבה אחת מחובר לנוירונים בשכבה הבאה. המידע עובר בין השכבות, כאשר כל נוירון מעבד את הקלט שהוא מקבל תוך שימוש בפונקציה חישובית נתונה ומעביר את התוצאה לשכבה הבאה. המבנה ההיררכי של רשתות נוירוניות מאפשר להן ללמוד מגוון רחב של תכונות ודפוסים, מהפשוטים ביותר ועד למורכבים ביותר.

**שלוש שכבות הרשת**

1. **שכבת קלט** - מקבלת את הנתונים הראשוניים לעיבוד. זו השכבה שבה הנתונים מוזנים לראשונה לרשת.
2. **שכבות נסתרות** - מהוות את ליבת החישוב של הרשת. בשכבות אלו מתבצעים הפעולות המרכזיות של עיבוד הנתונים. מספר השכבות ומספר הנוירונים בכל שכבה יכול להשתנות והוא מותאם לסוג הבעיה ולמורכבות הנדרשת.
3. **שכבת הפלט** - מספקת את התוצאה הסופית לאחר כל החישובים. התוצאה יכולה להיות כל סוג של נתון, כגון קטגוריה, מספר, או כל תגובה אחרת שהמודל נדרש לייצר.

גם ברשתות נוירוניות המטרה של החיזוי היא לעדכן את משקלי המודל שמשמשים לחיזוי לפי פונקציית עלות , שיטה זאת ברשתות נוירוניות נקרא Backward Propagation, או התפשטות לאחור. שיטה זו משמשת לעדכון המשקולות וההטיות ברשת במטרה למזער את פונקציית העלות של הרשת. התהליך כולל מספר שלבים עיקריים:

1. **הפצת קדימה (Forward Propagation)**: בשלב זה, הנתונים מוזנים לרשת מהקצה הקדמי ועוברים דרך השכבות שלה עד לשכבה האחרונה שמוציאה את התחזית. בכל שכבה, נעשה שימוש בפונקציית הפעלה כדי להכריע על ערך הפלט של כל נוירון.
2. **חישוב פונקציית העלות**: לאחר שהרשת מבצעת תחזית, נערך חישוב של פונקציית העלות, המשווה בין התוצאה שהרשת חזתה לתוצאה האמיתית. הטעות שנמדדת היא הבסיס לעדכון המשקולות.
3. **התפשטות לאחור (Backward Propagation)**: בשלב זה, הטעות שחושבה מועברת בחזרה דרך הרשת, מהשכבה האחרונה לראשונה. בכל שכבה, מתבצע חישוב של הנגזרת של פונקציית העלות ביחס לכל משקולת והטיה. כל משקולת והטיה מתעדכנים בהתאם לשיעור הלמידה ולגרדיאנט שנמדד, במטרה להקטין את פונקציית העלות בהרצה הבאה.
4. **עדכון משקולות והטיות**: השלב הזה הוא המימוש הפיזי של התיקונים שהוצעו בשלב ההתפשטות לאחור. משקולות והטיות מעודכנות כדי לשפר את יכולת החיזוי של המודל.

בנושא העבודה, הרשתות מסוגלות לעבד ולחזות מגמות שוק מתוך נתונים היסטוריים עשירים, תוך למידת תכונות ודפוסים שקשה לזהות באמצעים אחרים. הן עושות זאת באמצעות יכולתן להבין תלות זמנית בין הנתונים, כלומר לחזות את ההשלכות של אירועים על מגמות עתידיות. זוהי יכולת חשובה במיוחד בשוק המניות, שם המחירים עשויים להשתנות בתדירות גבוהה על פי מגוון גורמים כלכליים וחברתיים.

## תיאור הפרויקט וסביבת העבודה

הפרויקט נועד לחקור חיזוי מחירי מניות, תוך שימוש במודלים של רשתות נוירוניות. פיתוח המחקר נעשה בסביבת Jupyter Notebook, שהיא סביבה אינטראקטיבית המאפשרת כתיבה נוחה ומהירה של קוד בשפת Python, יחד עם תיעוד וויזואליזציה של נתונים ותוצאות.

המחקר נעשה בשפת Python, שנבחרה בשל התמיכה הרחבה שלה בספריות עיבוד נתונים ולמידת מכונה. כחלק מהעבודה, נעשה שימוש בספריות כמו:

**NumPy** ו-**Pandas**, שהן הבסיס לכל פעולות העיבוד והניתוח של נתונים כמותיים, מאפשרות טיפול יעיל ומהיר במערכים גדולים של נתונים.

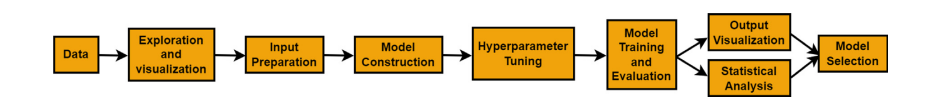
**Matplotlib** ו-**Seaborn**, שמספקות כלים חזקים לוויזואליזציה של נתונים, הכרחיות להצגה ברורה וממחישה של הממצאים.

**Scikit-learn**, שמציעה כלים מתקדמים למידול, כמו פונקציות למדידת ביצועים, קליטת נתונים ועיבוד מקדים, המאפשרים לכוונן את המודלים לביצועים טובים יותר.

**TensorFlow** ו-**Keras**, שמאפשרות בניית ואימון רשתות נוירוניות עמוקות בצורה פשוטה ויעילה, עם תמיכה במבנים מתקדמים כמו LSTM לחיזוי סדרות זמן.

**TA-Lib**, ספריה שמשמשת לניתוח טכני של נתוני שוק המניות, עם מגוון רחב של פונקציות לזיהוי דפוסים וחישוב מדדים.

העבודה ב-Jupyter Notebook מתאפיינת בכתיבה אינטראקטיבית, כאשר המשתמש יכול לראות בזמן אמת את תוצאות הקוד שהוא כותב. הדבר מאפשר בדיקה ותיקון מהירים של באגים, כמו גם יכולת לשנות ולהתאים את הקוד ללא צורך בהרצה מחדש של תסריטים ארוכים. שימוש ב-Jupyter מסייע גם בשיתוף פעולה ובהצגת העבודה לאחרים, שכן המסמכים ניתנים לשיתוף בקלות ומכילים את כל המידע הדרוש להבנת הניתוחים והתוצאות.

נתאר בתרשים את תהליך פיתוח מודל למידת מכונה, החל משלב איסוף הנתונים ועד בחירת המודל הסופי לשימוש. התהליך מורכב מתשעה שלבים עיקריים:

תמונה 1: LSTM-SDM: An integrated framework of LSTM implementation for sequential data modeling

1. **Data**: השלב הראשון בתהליך הוא איסוף הנתונים, שבו מתבצעת אגירה של נתונים רלוונטיים למחקר ממקורות שונים.
2. **Exploration and Visualization**: בשלב זה מתבצעת חקירה ראשונית של הנתונים כדי לזהות תכונות, טרנדים ודפוסים, תוך שימוש בטכניקות ויזואליזציה שונות להמחשת הממצאים.
3. **Input Preparation**: לאחר מכן, מתבצע עיבוד הנתונים לקראת יישום המודל, כולל ניקוי נתונים, טיפול בערכים חסרים, סטנדרטיזציה ועוד.
4. **Model Construction**: בשלב זה מתבצעת הגדרת המודל ובנייתו, כולל בחירת אלגוריתם מתאים והגדרת מבנה המודל עם שכבות, נוירונים וקשרים.
5. **Hyperparameter Tuning**: שלב זה כולל אופטימיזציה של המודל על ידי כיוון ערכי ההיפרפרמטרים למקסום ביצועי המודל על נתוני האימון.
6. **Model Training and Evaluation**: בשלב זה מתבצע אימון המודל על נתוני האימון והערכתו על נתוני הבדיקה, תוך שימוש במטריקות שונות כמו דיוק ו-F1 למדידת ביצועי המודל.
7. **Output Visualization**: ויזואליזציה של תוצאות המודל, שמספקת הבנה עמוקה של האופן בו המודל פועל וכיצד הוא מתמודד עם הנתונים.
8. **Statistical Analysis**: ניתוח סטטיסטי של התוצאות לקבלת תובנות נוספות על יעילות המודל והשפעת גורמים שונים על ביצועיו.
9. **Model Selection**: השלב הסופי, שבו מתבצעת בחירת המודל הסופי לשימוש על פי קריטריונים שהוגדרו מראש, לאחר שכל השלבים הקודמים הושלמו והמודל הוכח כיעיל ואמין.

התהליך מדגים את הדרך בה מפתחים מודל למידת מכונה, כשכל שלב מוביל ותומך בשלב הבא, מה שמבטיח שהמודל הסופי יהיה מדויק ויעיל ככל האפשר.

## איסוף ועיבוד המידע

### 3.1 סקירת מחקרים קודמים ואיסוף המידע

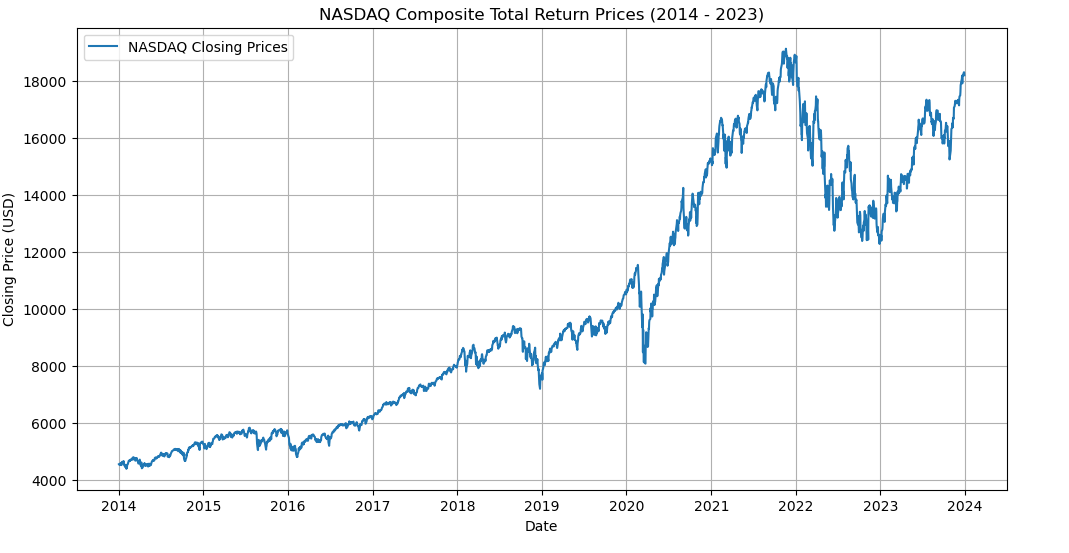
בשלב הראשון של הפרויקט שלי, אני מתמקדת באיסוף מאמרים רלוונטיים שעוסקים בחיזוי מניות באמצעות רשתות נוירונים. מטרת שלב זה היא לפתח הבנה עמוקה של המורכבות והאתגרים בתחום זה, תוך מתן דגש על מחקרים עדכניים ככל האפשר. מסקירת המאמרים עולה כי הפתרון הנפוץ ביותר לחיזוי מניות הוא שימוש במודל LSTM, שנבחר על בסיס תוצאותיו הטובות בחיזוי התנהגות שווקים פיננסיים.

בשלב איסוף המידע, הורדתי נתונים ממערכת בלומברג, אשר מספקת מידע פיננסי מדוייק ומוביל בתחומו. אספתי נתונים לעשור האחרון ממדד נפוץ מאוד NASDAQ Corporate כדי לאמן ולבחון את המודל. לאור מעמדה של בלומברג כמערכת מובילה, הנתונים שהושגו נחשבים לבעלי דיוק גבוה ואמינות מרבית. בהתחלה בחנתי גם את האפשרות להשתמש בנתונים מ-Yahoo Finance, מקור שהוזכר במאמרים אקדמאיים רבים, אך נמצא שהמידע שם עשוי להיות חסר ולא מדוייק כמו בבלומברג. לכן, החלטתי להשתמש בנתונים מבלומברג לצורך המחקר.

ה-NASDAQ Composite הוא אינדקס שוק המורכב מכל החברות המסחריות בבורסת NASDAQ, והוא משמש כאינדיקטור חשוב למצבו של טכנולוגיה ומגזרים חדשניים בשוק ההון. אינדקס זה כולל חברות גדולות וידועות כמו Apple, Amazon, ו-Google, אך גם חברות קטנות וחדשות יותר. מאפיינו המרכזי של ה-NASDAQ הוא התמקדותו בחברות טכנולוגיה, מה שהופך אותו למדד מוביל למעקב אחר ביצועי התעשייה הטכנולוגית.

בהקשר למחקר וחיזוי בתחום הפיננסי, ה-NASDAQ Composite מספק נתונים רבי ערך על תנודות השוק ומגמות הצמיחה או הירידה במגזר הטכנולוגי. חיזוי התנועות באינדקס זה יכול להיות קריטי למשקיעים, חברות, ואנליסטים כאחד, מאחר שהוא משקף טרנדים כלכליים רחבים והשפעות של אירועים גלובליים על ההיי-טק ועל הכלכלה בכלל.

אני מציגה גרף המתאר את המגמות במחירי הסגירה של ה-NASDAQ Composite לאורך תקופה של עשור (2014-2023). הגרף מדגים את התנודות בשוק ההון ומציג כיצד מגמות כלכליות וטכנולוגיות משפיעות על המדד.



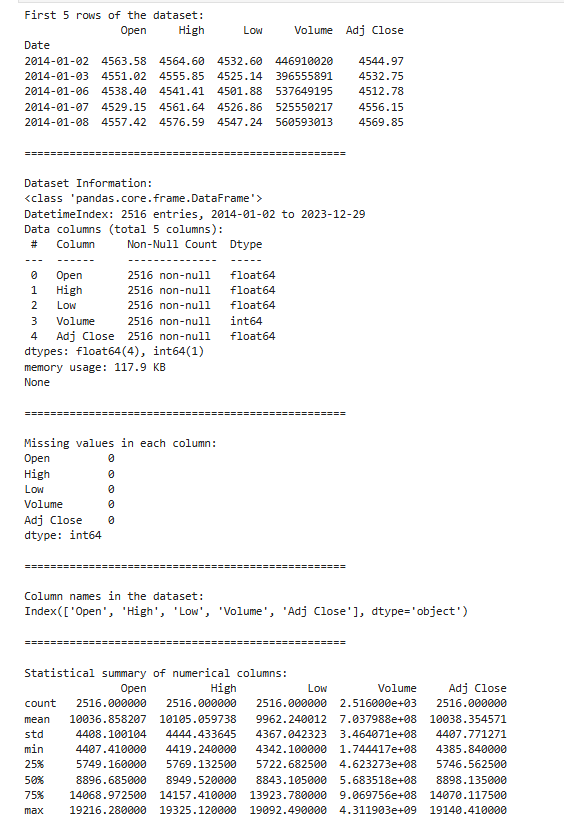
הנתונים שנאספו מבלומברג נלקחו מהאינדקס XCMP, שמייצג את ה-Total Return ,זהו מונח כלכלי המתאר את התשואה הכוללת שמשקיע מקבל מהשקעה בנייר ערך, כולל עליות וירידות במחיר הנייר, וכן דיבידנדים או הכנסות נוספות שהנייר מייצר. זהו מדד חשוב שמאפשר למשקיעים להעריך את הביצועים האמיתיים של השקעותיהם לאורך זמן.

הטבלה מכילה את העמודות הבאות, המספקות מידע על הביצועים היומיים של האינדקס:

* **Date**: מראה את תאריך המסחר של כל יום.
* **Open**: המחיר שבו נפתח המסחר לנייר ערך באותו יום.
* **High**: השיא, או המחיר הגבוה ביותר שהושג במהלך יום המסחר.
* **Low**: התחתית, או המחיר הנמוך ביותר אליו ירד הנייר באותו יום.
* **Volume**: כמות המניות שנסחרו באותו יום.
* **Adj Close**: מחיר הסגירה של הנייר, כשהוא מותאם לשינויים כמו דיבידנדים או פיצול מניות.

הערך 'Adj Close' חשוב במיוחד מכיוון שהוא משקף את התשואה הכוללת על ההשקעה לאחר התחשבות בדיבידנדים, מאפשר להשוות ביצועים בצורה עקבית ומדויקת. זהו גם הפרמטר שנמצא בשימוש במחקר אקדמי נוסף ועדכני [1], כדי להבטיח שהערכה של ביצועי ההשקעות תהיה מדויקת ועקבית.

בהמשך ביצעתי ניתוח ראשוני של הנתונים. במסגרת הניתוח, הוצגה תצוגה מקדימה של חמשת השורות הראשונות מהאקסל, מה שאיפשר להבחין במבנה הכללי ובערכים הספציפיים כגון Open, Adj Close, ועוד. כמו כן, נבדקו מאפייני המידע כגון סוגי הנתונים וכמות הערכים הלא חסרים בכל עמודה, שם לא נמצאו ערכים חסרים בשום עמודה. הדבר מציין כי הנתונים נקיים ואין צורך בתהליכי ניקוי או השלמה. סקירה סטטיסטית כללית חשפה את הממוצע, סטיית התקן, והערכים המקסימליים והמינימליים, דבר שמעיד על טווח התנודתיות ומאפיינים נוספים של הנתונים. מידע זה מהווה בסיס חיוני לקביעת שיטות ניתוח מתקדמות והערכת צורך בטרנספורמציות נתונים לפני יישום מודלים פרדיקטיביים.



### 3.2 הוספת פרמטרים נגזרים

בבנייה של מודלים לחיזוי מחירי מניות, כמו מודלי LSTM, בחירת התכונות (features) היא קריטית לביצועי המודל. אחת מהתכונות המרכזיות שיכולות לשפר את הדיוק הן תכונות נגזרות (Derived Features), אשר מחושבות מתוך הנתונים ההיסטוריים של המניה, כמו מחירים קודמים, נפח מסחר או תנודתיות.

במחברת ה-Jupyter שבה השתמשתי, שמתי דגש על גישה גנרית בהגדרת התכונות הנגזרות. בניגוד למאמרים שבהם מקובל להשתמש בפרמטרים קבועים, כמו חישוב מומנטום ליום, יומיים או חמישה, הקוד שלי מקבל את הערכים הללו כקלט במערך. בהתאם לכך, הפונקציות שנבנו יוצרות את העמודות הנוספות על פי הערכים שהוזנו, מה שמאפשר גמישות רבה יותר בניסויים ושימוש חוזר בקוד. גישה זו ננקטה גם עבור פרמטרים אחרים, כך שכל אחד מהם ניתן להגדרה בצורה דינמית לפי הצורך.

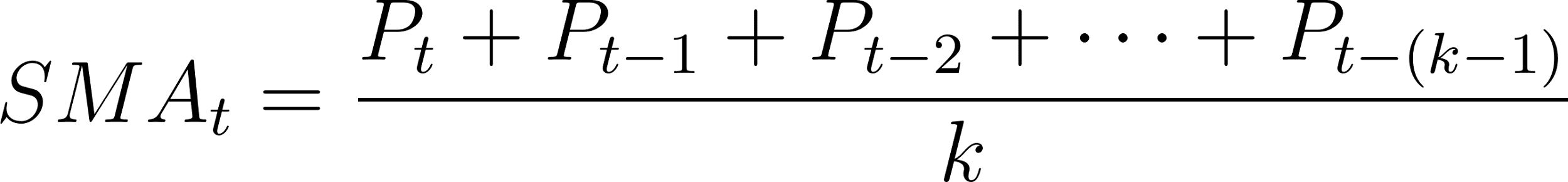
שיטה זו מאפשרת עבודה יעילה וגמישות מרבית בתהליך החיזוי, כך שניתן להתאים את המודל למגוון רחב של תרחישים בצורה פשוטה ומהירה.

בפרק זה נסקור את כל הפרמטרים הנגזרים שבהם השתמשתי, כמו מומנטום, ממוצעים נעים ומדדים נוספים, כאשר הבחירה בהם נעשתה בהשראת המאמר [1].

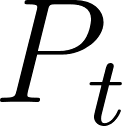
**ממוצע נע פשוט (Simple Moving Average (SMA**

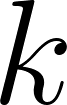
ממוצע נע פשוט (SMA) הוא כלי נפוץ בניתוח טכני המשמש להחלקת נתוני מחיר ולזיהוי מגמות בשוק. ה-SMA מחושב על ידי לקיחת ממוצע המחירים לאורך תקופה קבועה. הממוצע הנע עוזר להחליק תנודות קצרות טווח, כדי לאתר מגמות ארוכות יותר.

הנוסחה לממוצע נע פשוט עבור תקופה של n ימים היא:

[](https://www.codecogs.com/eqnedit.php?latex=SMA_t%20%3D%20%5Cfrac%7BP_%7Bt%7D%20%2B%20P_%7Bt-1%7D%20%2B%20P_%7Bt-2%7D%20%2B%20%5Cdots%20%2B%20P_%7Bt-(k-1)%7D%7D%7Bk%7D#0)

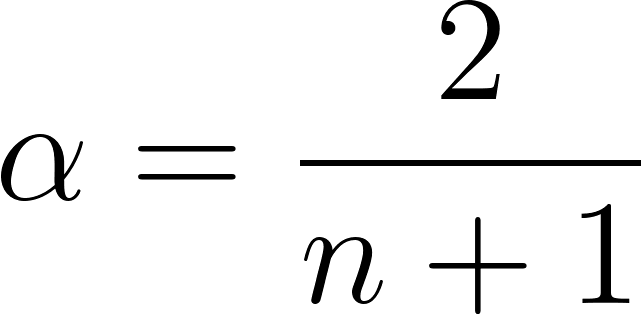
כאשר [](https://www.codecogs.com/eqnedit.php?latex=SMA_t#0) זה הממוצע הנע בזמן t

[](https://www.codecogs.com/eqnedit.php?latex=P_t#0) זה מחיר המניה בזמן t

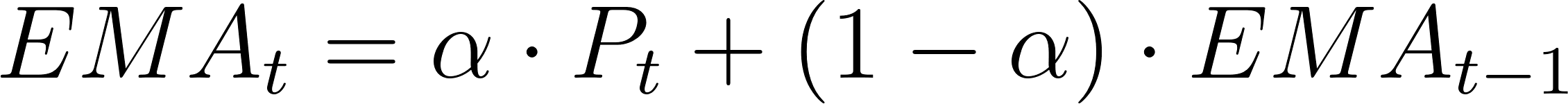
[](https://www.codecogs.com/eqnedit.php?latex=k#0) זה מספר הימים שעבורם מחושב הממוצע

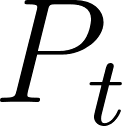
**ממוצע נע אקספוננציאלי** (**Exponential Moving Average ( EMA**

ממוצע נע אקספוננציאלי הוא כלי ניתוח טכני המעניק משקל גבוה יותר לנתונים האחרונים, בניגוד לממוצע נע פשוט (SMA) שמעניק משקל שווה לכל הנתונים. EMA נועד להגיב מהר יותר לשינויים האחרונים במחירי הנכס, ולכן הוא מתאים יותר למצבים שבהם נדרש זיהוי מהיר של מגמות חדשות.

בממוצע נע אקספוננציאלי, כל מחיר מקבל משקל יחסי, כאשר המחירים האחרונים מקבלים את המשקל הגבוה ביותר. המשקל של כל מחיר נקבע על ידי פרמטר ההחלקה [](https://www.codecogs.com/eqnedit.php?latex=%5Calpha#0), המחושב כיחס של הזמן הנבחר: 

כאשר n הוא מספר התקופות (למשל, ימים) בהם מתחשבים בחישוב ה-EMA

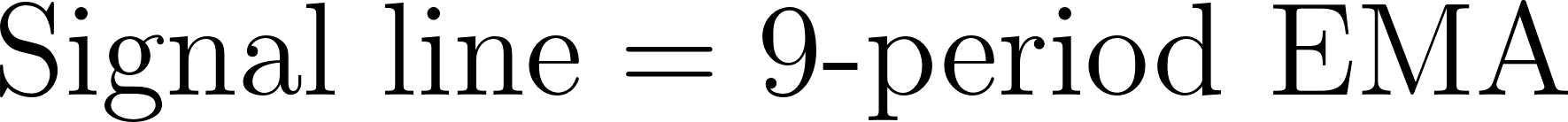
הנוסחה הבסיסית לחישוב הממוצע האקספוננציאלי ביום t היא:

[](https://www.codecogs.com/eqnedit.php?latex=P_t#0) זה מחיר המניה בזמן t

(**Moving Average Convergence Divergence (MACD**

MACD הוא אינדיקטור טכני פופולרי המשמש לזיהוי שינויים במומנטום ובמגמות בשווקי ההון, במיוחד במסחר מניות. ה-MACD מחושב כהפרש בין שני ממוצעים נעים אקספוננציאליים (EMA): ממוצע נע מהיר (בדרך כלל על פני 12 תקופות) וממוצע נע איטי (בדרך כלל על פני 26 תקופות). ההפרש ביניהם יוצר את קו ה-MACD, אשר מושווה לקו signal שהוא ממוצע נע אקספוננציאלי של ה-MACD עצמו (בדרך כלל על פני 9 תקופות). אחד השימושים הנפוצים ב-MACD הוא זיהוי סטיות (Divergences), כאשר המחיר נע בכיוון אחד וה-MACD בכיוון אחר, מה שיכול לרמוז על היפוך מגמה. היתרון המרכזי של ה-MACD הוא שילוב בין אלמנטים של מגמה ומומנטום, מה שהופך אותו לכלי חשוב עבור סוחרים המעוניינים בקבלת החלטות קנייה ומכירה מושכלות על בסיס מגמות השוק.

לכן יש 3 אינדיקטורים שאנו מחשבים:

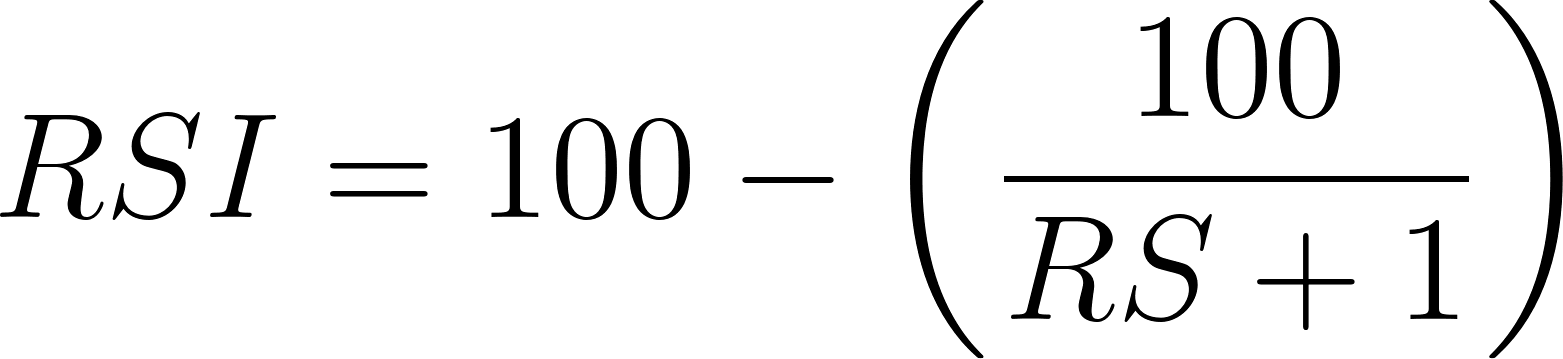
1. **MACD Line קו MACD**: מציג את הפער בין ממוצע נע לתקופה קצרה ולתקופה ארוכה, ומאפשר לזהות מגמות שינוי במחיר, כאשר עלייה או ירידה של הקו מעלה נקודות מפנה אפשריות בתנודתיות של המנייה.
2. **Signal Line (קו האיתות)**: מהווה ממוצע נע של ה-MACD Line ומשמש כאיתות לקניה או מכירה בהתבסס על חיתוך של קו ה-MACD עם קו זה, כאשר חיתוך מלמעלה למטה מצביע על מכירה ומלמטה למעלה מצביע על קניה.
3. **Histogram (היסטוגרמה)**: מראה את ההפרש בין ה-MACD Line ל-Signal Line, ומאפשר לזהות את התחזקות או החלשות של המגמה הנוכחית, כאשר ההפרשים הולכים וגדלים או מצטמצמים.

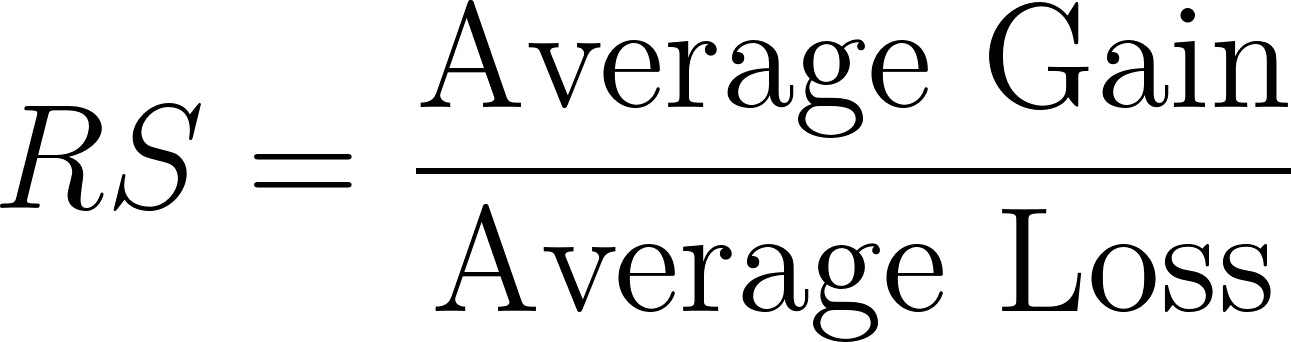
כל אחד מהם מספק תובנות שונות אך משלימות על דינמיקת המחירים, וביחד הם מהווים כלי חזק לניתוח טכני שמטרתו לחזות את התנהגות המחיר בעתיד.

**(Relative Strength Index (RSI**

ה-(Relative Strength Index (RSI הוא אינדיקטור טכני פופולרי המשמש להערכת עוצמת השינויים במחיר של מניה, במטרה לזהות אם היא במצב של קניית יתר (*Overbought*) או מכירת יתר (*Oversold*). ה-RSI נע בטווח של 0 עד 100 ונועד למדוד את מהירות ותנופת התנועות במחיר. ערך RSI מעל 70 נחשב למצב של קניית יתר, מה שעשוי להעיד על כך שהמניה עלולה לחוות ירידה במחיר. ערך RSI מתחת ל-30 נחשב למצב של מכירת יתר, מה שיכול להעיד על כך שהמניה נמצאת במצב של ירידת יתר ועלולה לעלות במחיר בקרוב.

הנוסחה לחישוב ה-RSI:

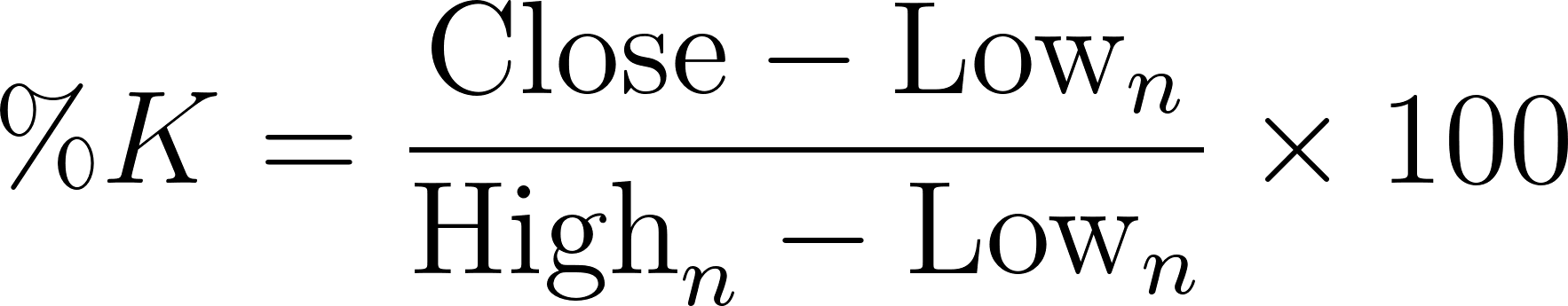




**Stochastic K**

ה-Stochastic K הוא אינדיקטור טכני פופולרי בתחום המסחר במניות ובניתוח טכני של שוקי הון, והוא נועד לזהות נקודות קנייה ומכירה אפשריות על ידי השוואת המחירים הנוכחיים של מניה לטווח המחירים שלה בפרק זמן מוגדר. אינדיקטור זה מבוסס על הרעיון שכשמחיר של מניה נמצא בעלייה, הוא נוטה להיסגר קרוב לקצה העליון של טווח המחירים שלה, וכאשר מחיר המניה יורד, הוא נוטה להיסגר קרוב לקצה התחתון.

הוא מחושב על בסיס מיקום המחיר הנוכחי של המניה ביחס לטווח המחירים שלה בתקופה האחרונה. הנוסחה לחישוב Stochastic K נראית כך:



[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BClose%7D#0)– המחיר הנוכחי של המניה (מחיר סגירה).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BLow%7D_n#0)– המחיר הנמוך ביותר בפרק הזמן שנבחר (n).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BHigh%7D_n#0)– המחיר הגבוה ביותר בפרק הזמן שנבחר (n).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7Bn%7D#0) – מספר הימים שנבחרו לחישוב (לרוב משתמשים ב-14 ימים)

תוצאת החישוב של ה-Stochastic K נותנת ערך בין 0 ל-100, כאשר ערכים קרובים ל-100 מצביעים על כך שהמחיר הנוכחי נמצא קרוב לקצה העליון של טווח המחירים שנבחן, וערכים קרובים ל-0 מצביעים על כך שהמחיר קרוב לקצה התחתון. ה-Stochastic K משמש לסימון מצבים של קניית יתר (Overbought) כאשר הערכים גבוהים מ-80, או מכירת יתר (Oversold) כאשר הערכים נמוכים מ-20. במצבים אלה, ייתכן שהמניה צפויה לחוות שינוי כיוון, כמו עלייה במצב של מכירת יתר או ירידה במצב של קניית יתר.

**מומנטום (Momentum)**

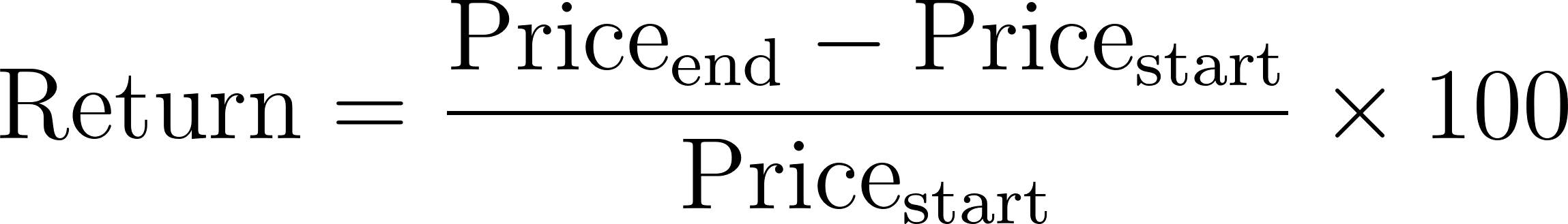
המומנטום (Momentum) הוא כלי בניתוח טכני המודד את קצב השינוי במחירי נייר ערך לאורך זמן, והוא משמש להערכת עוצמת המגמה הנוכחית של המניה. הנוסחה הבסיסית לחישוב מומנטום היא ההפרש בין המחיר הנוכחי לבין המחיר לפני מספר מסוים של ימים: ​

מומנטום חיובי מעיד על עלייה במחיר לעומת התקופה שנבחרה, מה שמצביע על מגמת עלייה, בעוד מומנטום שלילי מצביע על ירידה במחיר ועל מגמת ירידה.



**Return**

ה-Return Indicator הוא מדד המשמש להערכת התשואה שהניבה מניה או נכס פיננסי מסוים על פני תקופת זמן. התשואה מודדת את אחוז השינוי בערכו של הנכס, והיא משקפת את הרווח או ההפסד שנוצר למשקיע כתוצאה מהשקעה בנכס זה.

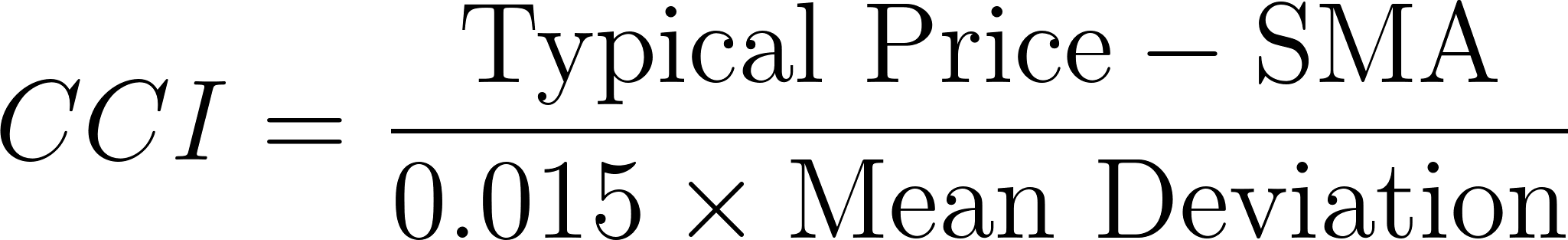


[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BPrice%7D_%7B%5Ctext%7Bend%7D#0) – המחיר הנוכחי או הסופי של הנכס.

[](http://www.sciweavers.org/tex2img.php?bc=Transparent&fc=Black&im=jpg&fs=100&ff=modern&edit=0&eq=%5Ctext%7BPrice%7D_%7B%5Ctext%7Bstart%7D%E2%80%8B#0) – המחיר ההתחלתי של הנכס בתחילת התקופה הנמדדת.

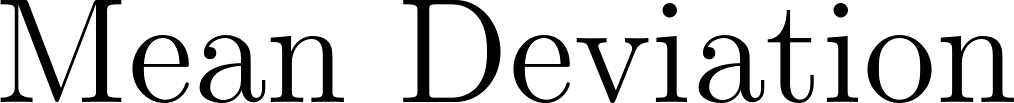
**Commodity Channel Index (CCI)**

ה-(Commodity Channel Index (CCI הוא אינדיקטור טכני שמשמש להערכת סטיות מהמחיר הממוצע של נייר ערך או סחורה לאורך זמן. למרות שמו, ה-CCI אינו משמש רק לסחורות אלא גם למניות וניירות ערך אחרים. הוא עוזר לסוחרים לזהות מצבים של קניית יתר (*Overbought*) או מכירת יתר (*Oversold*), וכך להחליט על נקודות כניסה ויציאה מהשוק. כאשר ה-CCI חוצה את ה-100 כלפי מעלה, ייתכן שזהו סימן לעלייה נוספת במחיר. לעומת זאת, כשה-CCI יורד מתחת ל-(-100), זה יכול להוות סימן לקנייה, מכיוון שהמחיר עשוי להתחיל לעלות. נוסחת ה CCI:



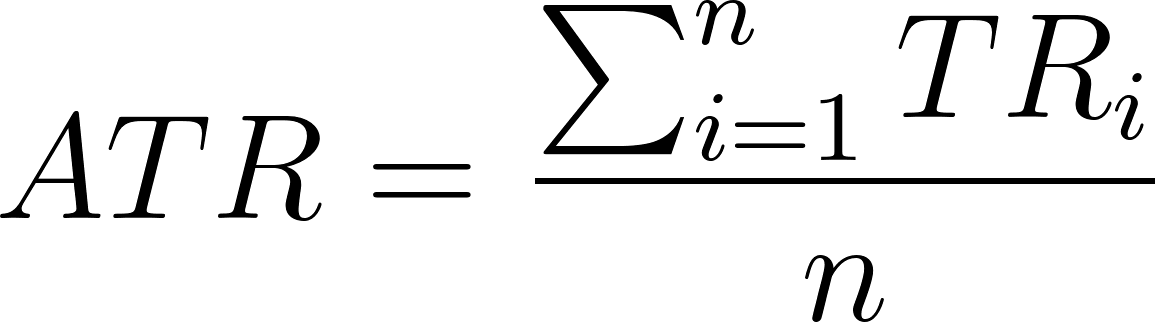


[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BSMA%7D#0): ממוצע נע פשוט (Simple Moving Average) של ה-Typical Price, המחושב על פני מספר ימים (לרוב 20).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BMean%20Deviation%7D#0): סטיית המחירים הממוצעת מהממוצע הנע.

0.015: מקדם שנבחר כדי להתאים את התוצאה לטווח המצוי לרוב בין 100+ ל-100-.

**Average True Range (ATR)**

ה-(Average True Range (ATR הוא אינדיקטור טכני המודד את התנודתיות של נייר ערך (כגון מניה או סחורה) על פני תקופת זמן. הוא פותח על ידי ויילדר כדי להעריך את התנודתיות בשוק, והערכים של ה-ATR גבוהים מעידים על תנודתיות גבוהה, בעוד שערכים נמוכים מעידים על תנודתיות נמוכה.



[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7Bn%7D#0) – מספר הימים לחישוב ה-ATR (לרוב משתמשים ב-14 ימים).

**Accumulation/Distribution Index (A/D Index)**

ה-(Accumulation/Distribution Index (A/D הוא אינדיקטור טכני שנועד להעריך האם מניה או נייר ערך נמצאים במגמת צבירה (Accumulation), כאשר קונים מצטברים ורוכשים אותו, או במגמת הפצה (Distribution), כאשר ישנם יותר מוכרים שמפיצים ומוכרים את הנייר. ה-A/D Index מחושב באמצעות שילוב של המחיר היומי (כמו מחיר הסגירה, המחיר הגבוה והנמוך) ונפח המסחר, וכך מאפשר לזהות את עוצמת הפעילות הקשורה למחירים הללו. כאשר מחיר הסגירה קרוב למחיר הגבוה של היום ויש נפח מסחר גבוה, זה מעיד על צבירה, כלומר רוב הפעילות היא של קונים. לעומת זאת, אם מחיר הסגירה קרוב למחיר הנמוך והנפח גבוה, זה עשוי להצביע על מגמת הפצה, כלומר רוב הפעילות היא של מוכרים. אינדיקטור זה מסייע לסוחרים לזהות לחצים שעשויים להצביע על המשך המגמה או על שינוי פוטנציאלי בשוק.

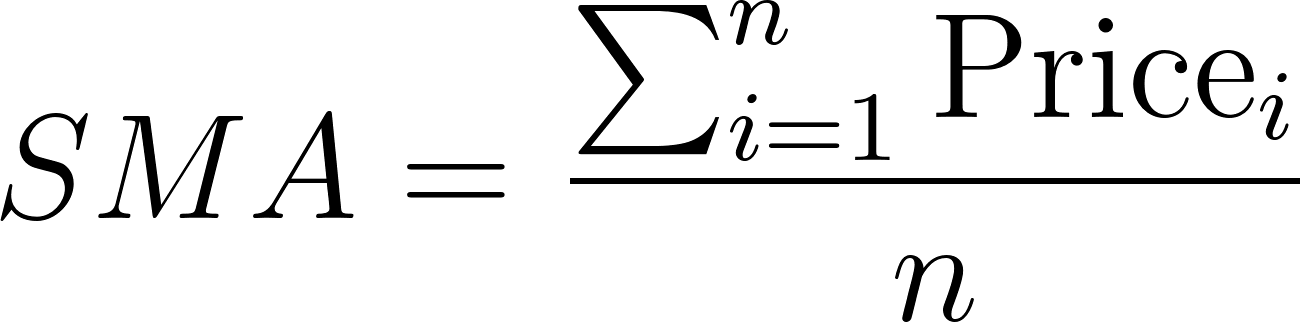
[](https://www.codecogs.com/eqnedit.php?latex=A%2FD%20%3D%20A%2FD_%7B%5Ctext%7Bprevious%7D%7D%20%2B%20%5Cfrac%7B((%5Ctext%7BClose%7D%20-%20%5Ctext%7BLow%7D)%20-%20(%5Ctext%7BHigh%7D%20-%20%5Ctext%7BClose%7D))%7D%7B%5Ctext%7BHigh%7D%20-%20%5Ctext%7BLow%7D%7D%20%5Ctimes%20%5Ctext%7BVolume%7D#0)

**Bollinger Bands**

Bollinger Bands הם אינדיקטור טכני המשמשים להערכת תנודתיות של נייר ערך ולזיהוי תנאים של קניית יתר (*Overbought*) או מכירת יתר (*Oversold*). Bollinger Bands מורכבים משלושה קווים: הממוצע הנע (*Simple Moving Average*, או SMA), קו עליון, וקו תחתון. הקווים העליון והתחתון ממוקמים במרחק של סטיות תקן מהממוצע הנע, ומשתנים בהתאם לתנודתיות השוק.

הנוסחאות הן:

הממוצע הנע הפשוט (SMA):



הפס העליון (Upper Band):



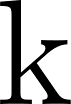
הפס התחתון (Lower Band):

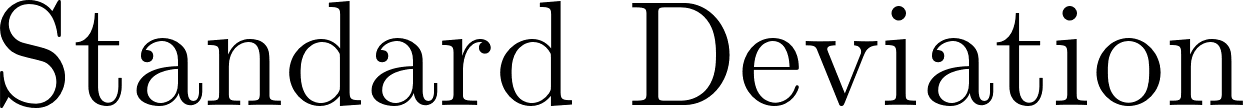


כאשר:

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BPrice%7D_i#0) – מחירי נייר הערך (לרוב מחירי סגירה).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7Bn%7D#0) – מספר התקופות עבור חישוב הממוצע הנע (בדרך כלל 20 תקופות).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7Bk%7D#0) – מספר הסטיות התקן (בדרך כלל 2 סטיות תקן).

[](https://www.codecogs.com/eqnedit.php?latex=%5Ctext%7BStandard%20Deviation%7D#0) – סטיית התקן של מחירי נייר הערך על פני התקופה הנמדדת.

### 3.3 הוספת מדדים חיצוניים

מעבר לתכונות שנגזרות ישירות מנתוני המניה או המדד,מודלים לחיזוי יכולים להפיק תועלת משילוב מדדים חיצוניים (External Indicators). מדדים אלו אינם מגיעים מנתוני המסחר עצמם, אלא מספקים מידע על מצב הכלכלה הרחב ועל גורמים חיצוניים שעשויים להשפיע על שוקי ההון. דוגמאות למדדים חיצוניים הם למשל שיעורי אבטלה ושיעורי ריבית.

שילוב מדדים חיצוניים במודל החיזוי מאפשר הבנה רחבה יותר של הכוחות הכלכליים המשפיעים על השוק. מדדים אלו יכולים לספק הקשר על מגמות כלכליות רחבות, כמו צמיחה כלכלית, לחץ אינפלציוני או שינויי ריבית, שיכולים לשנות את הביקוש למניות ואת תנועת המחירים בשוק. שילוב מידע זה במודל LSTM משפר את יכולת החיזוי על ידי קישור בין גורמים כלכליים כלליים למגמות מחירי מניות ספציפיות. מכיוון שהמדדים שאני מתמקדת בהם בעבודה הם של השוק האמריקאי כך גם המדדים שאציג.

**Effective Federal Funds Rate (EFFR)**

ה-(Effective Federal Funds Rate (EFFR הוא אחד המדדים המרכזיים בשוק הפיננסי האמריקאי, המייצג את הריבית הממוצעת בה בנקים בארה"ב מלווים זה לזה כספים לטווח קצר כדי לעמוד בדרישות הרזרבה של הבנק הפדרלי. ריבית זו נקבעת על בסיס היצע וביקוש בשוק ההלוואות הבין-בנקאי ומשפיעה ישירות על ריביות אחרות במשק, כמו ריביות על משכנתאות, הלוואות עסקיות וכרטיסי אשראי. ה-EFFR משמש ככלי מרכזי של הבנק הפדרלי לוויסות הצמיחה הכלכלית: כאשר הריבית נמוכה, היא מעודדת צריכה והשקעות, מה שמוביל לרוב לעליות בשוק המניות, בעוד ריבית גבוהה יכולה להאט את הפעילות הכלכלית ולגרום לירידות בשוק. מסיבה זו, ה-EFFR נחשב לאינדיקטור מרכזי לחיזוי מגמות בשוק ההון, שכן שינויים בו מאותתים למשקיעים על כיווני המדיניות המוניטרית הצפויה של הבנק הפדרלי ועל ההשלכות האפשריות עבור שוק המניות.

**US Dollar Index (USDX)**

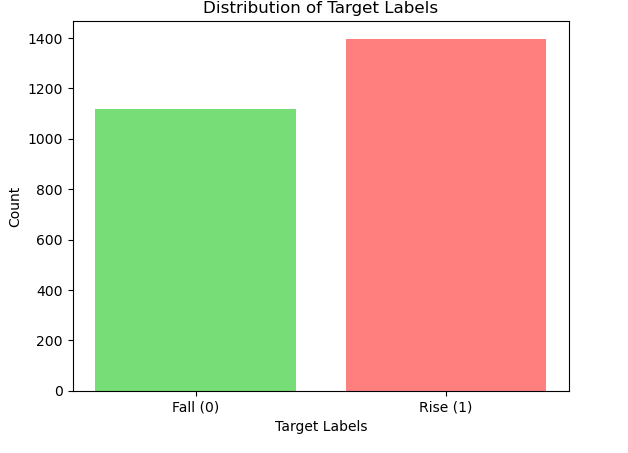
ה-(US Dollar Index (USDX הוא מדד חשוב שמודד את ערכו של הדולר האמריקאי ביחס לסל של שישה מטבעות עיקריים (כולל האירו, הין היפני והלירה שטרלינג), ומשמש אינדיקטור מרכזי להערכת חוזקו של הדולר בשוקי המט"ח. מאחר והדולר האמריקאי משפיע באופן ישיר על הסחר הבינלאומי, שינוי בערכו משפיע על מחירי סחורות, השקעות בינלאומיות ושווקי המניות. כאשר ה-USDX מתחזק, זה עשוי להעיד על מגמת התחזקות של הכלכלה האמריקאית, מה שמוביל בדרך כלל לעליות בשוק המניות המקומי, מכיוון שמשקיעים זרים נוטים להזרים כספים לשוק האמריקאי כשהדולר חזק. מנגד, ירידה ב-USDX יכולה לרמז על חולשה יחסית של הדולר ועליית מחירים של מוצרים המיובאים לארה"ב, מה שעשוי להשפיע על שוק המניות בצורה שלילית.

**Unemployment Rate - שיעור האבטלה בארה"ב**

שיעור האבטלה הוא אחד המדדים הכלכליים החשובים, המייצג את האחוז של כוח העבודה הכולל שאינו מועסק אך מחפש עבודה באופן פעיל. שיעור זה משקף את מצבה של הכלכלה: שיעור אבטלה גבוה עשוי להצביע על מיתון כלכלי, בעוד ששיעור נמוך יכול להצביע על צמיחה כלכלית והתפתחות. השפעת שיעור האבטלה על שוק המניות מורכבת: מצד אחד, שיעור אבטלה גבוה עשוי להוביל לירידות בשוק המניות מכיוון שהוא מעיד על חולשה כלכלית, צמצום בהוצאות הצרכנים ועלייה בחוסר הוודאות הכלכלית. מצד שני, שיעור אבטלה נמוך מאוד יכול להעלות חששות מלחץ שכר ועליית מחירים, שגם כן עלולים להשפיע לרעה על שוק המניות. בנוסף, פעולות המדיניות של הממשלה והבנק המרכזי בתגובה לשינויים בשיעור האבטלה יכולות להשפיע באופן משמעותי על שוק ההון.

### 3.4 סינון וארגון המידע

**הוספת שורת מטרה (Target) -** בפרויקט זה אני מגדירה את שורת המטרה 'Target' לחיזוי עליה או ירידה של המדד. השורה נוצרת על ידי חישוב השינוי במחיר הסגירה המתואם של המניה מהיום הקודם ליום הנוכחי. הערך בשורה זו מוגדר כ-'1' אם הייתה עלייה במחיר (שינוי חיובי) וכ-'0' אם הייתה ירידה או שאין שינוי. לאחר מכן, אנו מבצעים סכימה של התוצאות כדי להבין את התפלגות הנתונים ולאמוד את האיזון בין מקרי העלייה לירידה במחירי המדד. זיהוי זה חשוב כדי להבטיח שהמודל שאבנה יהיה מאוזן ויתן חיזויים מדויקים יותר.



**הסרת ערכים חסרים -** במסגרת ניתוח הנתונים, ניתן להבחין כי קיימות שורות עם מידע חסר שנובע מהשימוש בממוצעים נעים ובחישובי אינדיקטורים על פני מספר ימים, כמו מומנטום ומדדים אחרים המחושבים על בסיס נתוני עבר. התוצאה היא כי הערכים ההתחלתיים של מדדים אלו ריקים (NaN), כיוון שאין מספיק מידע מהעבר לחישוב הערכים הנדרשים. במצבים כאלו, אין טעם בהשלמת הערכים החסרים מכיוון שהם אינם תוצאה של נתונים חסרים אלא מאפיין של שיטת החישוב. לכן, במקרים אלו עדיף להסיר את השורות הראשונות עם המידע החסר מתוך מסגרת הנתונים, במיוחד כאשר מדובר במאגר נתונים גדול. הסרה של מספר קטן של ערכים התחלתיים הינה זניחה ואינה משפיעה על הניתוח הכללי.

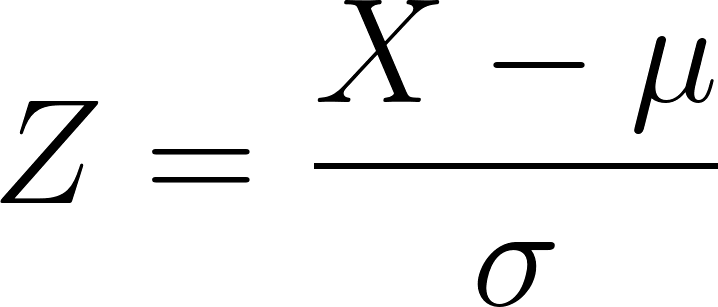
לאחר הסרת השורות המכילות מידע חסר, כמות השורות בטבלת הנתונים ירדה מ-2516 שורות ל-2467 שורות. בהתחשב בגודל הכולל של הנתונים, הירידה הזו נחשבת זניחה ואינה משפיעה באופן מהותי על הניתוח.

מלבד הערכים האלו אין ערכים ריקים להשלים.

**ביצוע סטנדרטיזציה של הנתונים -** אחת הטכניקות הנפוצות בעיבוד נתונים היא הסטנדרטיזציה, שבה מבצעים כיווץ של המאפיינים כך שהממוצע שלהם יהיה אפס וסטיית התקן שלהם תהיה אחת. תהליך זה חשוב במיוחד באלגוריתמים של למידת מכונה אשר מבוססים על אופטימיזציה של גרדיאנט ובאלגוריתמים המחשבים מרחקים בין נקודות נתונים, שכן הם עשויים להתקשות או להאט במקרה של מאפיינים בסדרי גודל שונים.



הנוסחה לסטנדרטיזציה של משתנה [](https://www.codecogs.com/eqnedit.php?latex=X#0) היא:

[](https://www.codecogs.com/eqnedit.php?latex=Z%20%3D%20%5Cfrac%7BX%20-%20%5Cmu%7D%7B%5Csigma%7D#0)

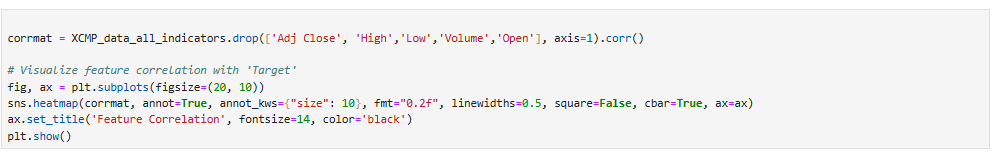
כאשר [](https://www.codecogs.com/eqnedit.php?latex=%5Cmu#0) הוא הממוצע של [](https://www.codecogs.com/eqnedit.php?latex=X#0) ו- [](https://www.codecogs.com/eqnedit.php?latex=%5Csigma#0) היא סטיית התקן של [](https://www.codecogs.com/eqnedit.php?latex=X#0). לאחר הסטנדרטיזציה, המשתנה המתוקן [](https://www.codecogs.com/eqnedit.php?latex=%20Z#0) יהיה עם ממוצע של אפס וסטיית תקן של אחת.

בחירה בסטנדרטיזציה על פני נרמול (שבו מבצעים כיווץ בדרך כלל לטווח [0,1]) היא מועילה במיוחד בחיזוי מניות מכיוון שהיא שומרת על היחס של המרחקים המקוריים בין הנתונים, דבר שיכול להיות קריטי כאשר המרחקים משקפים דינמיקות פיננסיות אמיתיות. בנוסף, הסטנדרטיזציה אינה מושפעת מנתונים קיצוניים כפי שהנרמול עלול להיות, דבר שחשוב מאוד כאשר מתמודדים עם נתוני מחירים של מניות, אשר לעיתים יכולים להציג תנודות חדות ולא צפויות. בפועל לביצוע הסטנדרטיזציה השתמשתי בספריה חיצונית שנקראת scikit learn וייבאתי מתוכה את האובייקט StandardScaler.

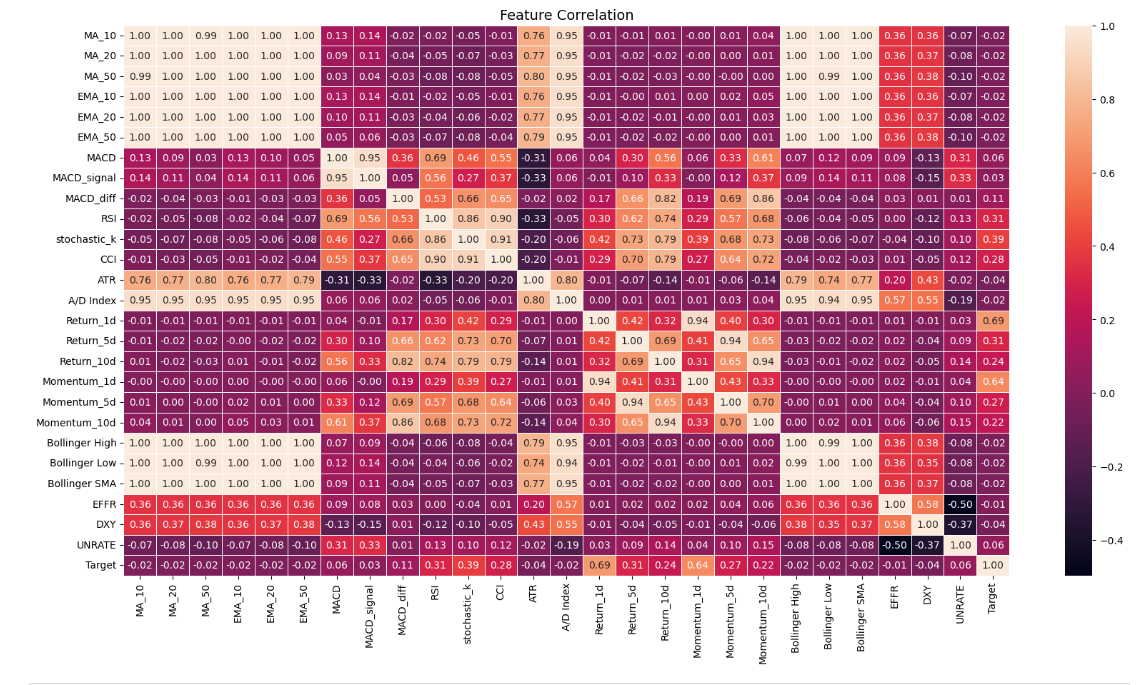
**מפת חום -** מפת החום, או "מטריצת קורלציה", היא כלי ניתוח נתונים שמשמש להצגת הקשרים הסטטיסטיים בין משתנים במערכת. הקורלציות מוצגות על גבי סקאלה של צבעים, כאשר כל תא במפה מייצג את עוצמת הקשר בין שני משתנים. מטריצת הקורלציה מאפשרת זיהוי ברור ומהיר של קשרים חיוביים ושליליים, ומסייעת לאנליזה מדויקת של הנתונים:

**ערך קורלציה גבוה וחיובי (קרוב ל+1)** מצביע על קשר חזק וישיר בין שני המשתנים. כלומר, כאשר אחד מהמשתנים גדל, השני גם כן נוטה לגדול.

* **ערך קורלציה גבוה ושלילי (קרוב ל-1)** מעיד על קשר חזק אך הפוך בין שני המשתנים. זאת אומרת, כאשר אחד מהמשתנים גדל, השני נוטה לקטון.
* **ערך קורלציה נמוך (קרוב ל-0)** מרמז שאין קשר משמעותי בין המשתנים. כלומר, שינויים באחד מהם אינם מנבאים שינויים בשני.



להצגת מפת החום השתמשתי בספריות seaborn ו matplotlib שמספקות כלים לוויזואליזציה של נתונים.



במהלך העיבוד המקדים של הנתונים לבניית מודל חיזוי מניות, נעשה סינון של משתנים בהתבסס על ערכי הקורלציה שלהם עם משתנה המטרה. משתנים אשר הקורלציה שלהם עם משתנה המטרה היא נמוכה בערך מוחלט מ-0.1 הוסרו מהנתונים. החלטה זו נבעה מכיוון שמשתנים עם קורלציה נמוכה כל כך עלולים להוות רעש בלבד ולא לתרום מידע משמעותי למודל. פעולה זו תורמת ליצירת מודל יעיל יותר שאינו מוטה על ידי נתונים שאינם רלוונטיים ומגבירה את היכולת לזהות את המשתנים שבאמת משפיעים על התנודות במחירי המניות. בנוסף, זה עוזר להבטיח שהמודל יהיה רגיש יותר לסיגנלים חשובים המגיעים מהמשתנים שנשארים ומעבירים מידע רלוונטי ומדויק על השוק.

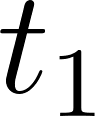
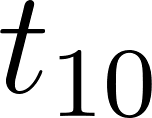
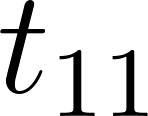
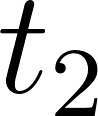
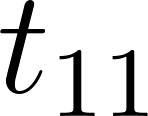
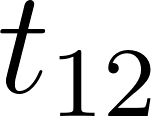
אחרי בחינת הקורלציות , המשתנים שנשארו במודל הם:

1. **MACD\_diff**: מדד זה הוא ההפרש בין שתי ממוצעים נעים המשמשים לזיהוי מגמות בשוק.
2. **RSI**: מדד כוח יחסי שמשקף מצבי יתר קנייה או יתר מכירה.
3. **Stochastic K**: נועד לזהות נקודות קנייה ומכירה אפשריות על ידי השוואת המחירים הנוכחיים של מניה לטווח המחירים שלה בפרק זמן מוגדר.
4. **CCI**: מדד חוזק המטבע, משמש להערכת סטיות מהמחיר המדד לאורך זמן.
5. **תזוזות מחיר**: המשתנים כמו 'Return\_1d', 'Return\_5d', 'Momentum\_1d', 'Momentum\_5d', ו-'Momentum\_10d' מייצגים תזוזות מחיר בפרקי זמן שונים. תזוזות אלו מעידות על התנהגות המחירים בטווחים קצרים ובינוניים ויכולות להיות מחוונים חשובים להתפתחויות עתידיות.

## חלוקת המידע והכנת המידע למודל

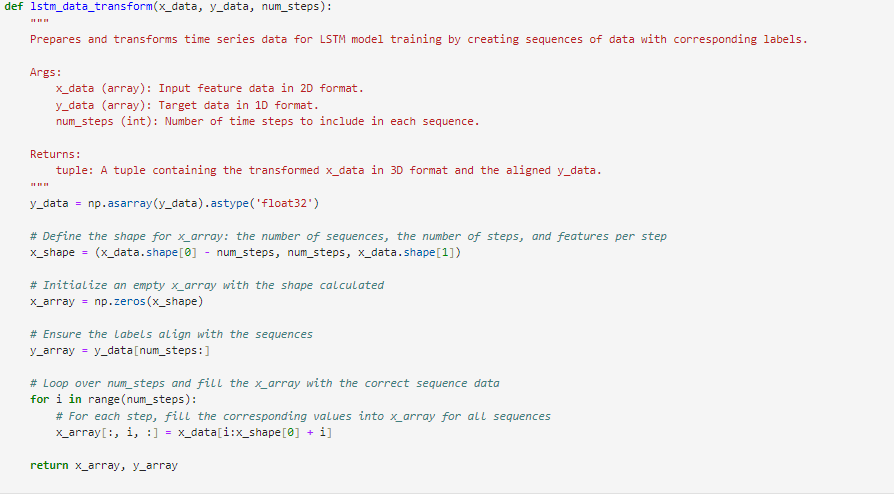
### 4.1 טרנספורמציית נתונים בשיטת חלון הזזה

הפונקציה lstm\_data\_transform מבצעת את השלב הראשוני בתהליך שבו היא מסדרת את הנתונים כך שיהיו מותאמים לרשת ה-LSTM. פונקציה זו בונה רצפים של נקודות זמן רציפות מתוך המידע ההיסטורי, כאשר כל רצף כולל מספר קבוע של צעדים (num\_steps), שמייצג חלון זמן מסוים. תהליך זה חשוב במיוחד, שכן הוא מאפשר לרשת ה-LSTM לראות קטעים רציפים של נתונים וללמוד מהם דפוסים של עליות או ירידות במחירי המניות.

תהליך חלון ההזזה מחלק את הנתונים לרצפים חופפים בעלי אורך קבוע של צעדים. כל רצף כולל מספר צעדים רצופים שמתקדמים בצעד אחד בכל פעם. לדוגמה, אם אורך החלון הוא 10 צעדים, הרצף הראשון יכלול את הנתונים בין הזמנים [](https://www.codecogs.com/eqnedit.php?latex=t_1#0)​ ל-[](https://www.codecogs.com/eqnedit.php?latex=t_%7B10%7D#0)​, והמודל ינבא את הערך ב-[](https://www.codecogs.com/eqnedit.php?latex=t_%7B11%7D#0). הרצף הבא יכלול את הנתונים בין [](https://www.codecogs.com/eqnedit.php?latex=t_%7B2%7D#0)​ ל-[](https://www.codecogs.com/eqnedit.php?latex=t_%7B11%7D#0)​ והחיזוי יהיה עבור הזמן [](https://www.codecogs.com/eqnedit.php?latex=t_%7B12%7D#0), וכן הלאה.

בכל שלב, המודל מתבונן בנתונים עבור מספר נקודות זמן קודמות (הנתונים בחלון) ומנסה לנבא את המגמה העתידית של המניה על בסיס רצף נתונים זה.

הבחירה בטכניקת חלון הזזה היא קריטית במודלים של חיזוי סדרות זמן, מאחר שהיא יוצרת רצפים חופפים שמאפשרים לרשת ללמוד את התלות בין הנתונים בנקודות זמן שונות ולנבא את הצעד הבא. כל חלון מכיל קבוצה של נתוני עבר שלפיהם מתבצע החיזוי של העתיד הקרוב.



### 4.2 חלוקת הנתונים לאימון, ולידציה ובדיקה

לאחר עיבוד הנתונים, התהליך ממשיך בחלוקה לשלושה סטים עיקריים: סט אימון (70%), סט ולידציה (15%) וסט בדיקה (15%). חלוקה זו מאפשרת לאמן את המודל על רוב הנתונים, להשתמש בסט הוולידציה להעריך את המודל עוד בשלב האימון, ולבסוף, לבדוק את הביצועים האמיתיים של המודל על נתונים שהוא לא נחשף אליהם במהלך האימון. אסביר על כל קבוצה:

### **קבוצת אימון (Training Set)**

קבוצת האימון היא הקבוצה הראשית שעליה מתבצע אימון המודל. נתונים אלו משמשים ללמד את המודל איך לזהות דפוסים, לעשות חיזויים, או לבצע כל פעולה אחרת שהמודל נועד לה. האלגוריתם מתאים את עצמו לנתונים הללו על ידי תיקון הפרמטרים וההיפרפרמטרים שלו כדי למקסם את ביצועיו על פי פונקציית העלות שנבחרה.

**קבוצת אימות (Validation Set)**

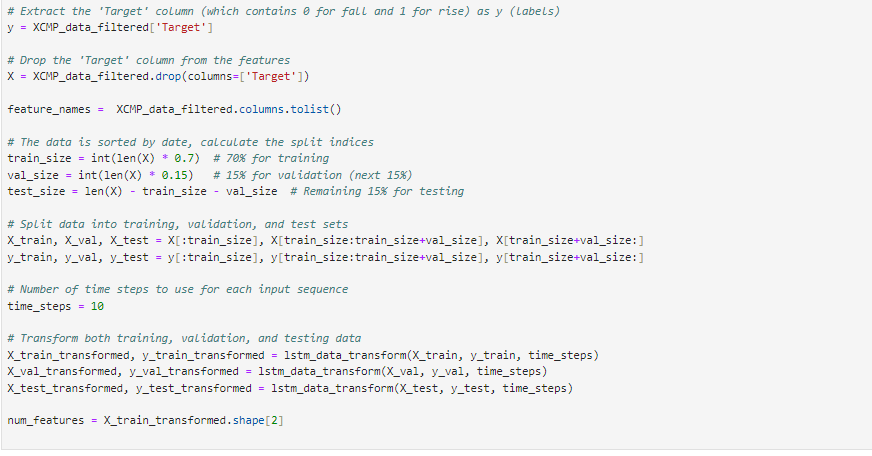
קבוצת האימות משמשת לאימות ביצועי המודל במהלך תהליך האימון, ולעזור לאפיין את רמת ההתאמה של המודל לנתונים שאינם חלק מקבוצת האימון. שימוש נפוץ בקבוצה זו הוא לכיוון היפרפרמטרים (למשל, לבחור את גודל הרשת ומספר הנוירונים) ולבחור בגרסת מודל מתאימה. היא מספקת מדד לאובייקטיביות של ביצועי המודל בעת ובעונה אחת כדי למנוע התאמה יתר של המודל לנתוני האימון.

**קבוצת בדיקה (Test Set)**

קבוצת הבדיקה היא קבוצת נתונים שמופרדת לחלוטין מהתהליך ומשמשת לבדוק את ביצועי המודל לאחר שהתהליך כולו הסתיים. היא מספקת הערכה סופית של איך המודל צפוי להתנהג כאשר הוא ייתקל בנתונים חדשים ולא ידועים בפועל. המטרה היא לאמוד את היכולת של המודל להכליל את למידתו למצבים שלא נתקל בהם במהלך האימון או האימות.

חשיבות חלוקת הנתונים טמונה בכך שהיא מאפשרת להעריך את היכולת של המודל להכליל דפוסים שהוא למד אל נתונים חדשים, דבר הכרחי לצורך יצירת מודל חיזוי מוצלח.

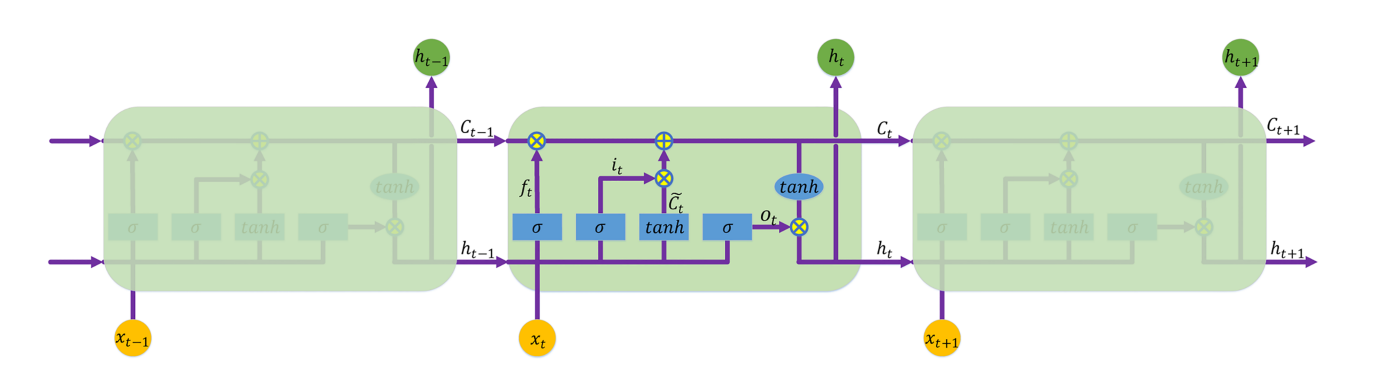
בנוסף לחלוקה לקבוצות שתיארתי בקוד הבא אנו מורידים את העמודה שאותה אנו רוצים שהמודל יחזה ומבצעים את הטרנפורמציה של המידע עבור המודל שתיארנו בסעיף הקודם 4.1



## מודל LSTM

### 5.1 בחירת המודל

ישנם גישות שונות לפתרון בעית חיזוי מניות. אלגוריתמים שתוכננו לפתור בעיות כאלו נחשבים למאתגרים במיוחד. נצפה שלרוב, בנתוני סדרות זמן ובמצבים שדורשים זיכרון של פלטים קודמים, ה-LSTM הוכיח כמשפר משמעותית את הביצועים בהשוואה למודלים אחרים שתוכננו למשימה הזו. ה-LSTM מצטיין בבעיות חיזוי רצפים בהשוואה לרשת נוירונים מחזורית (RNN). לכל תא LSTM יש שער קלט, שער פלט ושער שכחה. נתונים שנכנסים לרשת ה-LSTM נשמרים, בעוד שהנתונים המיותרים נמחקים על ידי שער השכחה. ניתן להשתמש ב-LSTM במגוון יישומים, כולל חיזוי מזג אוויר, עיבוד שפה טבעית (NLP), זיהוי דיבור, זיהוי כתב יד, חיזוי סדרות זמן ועוד. [2]

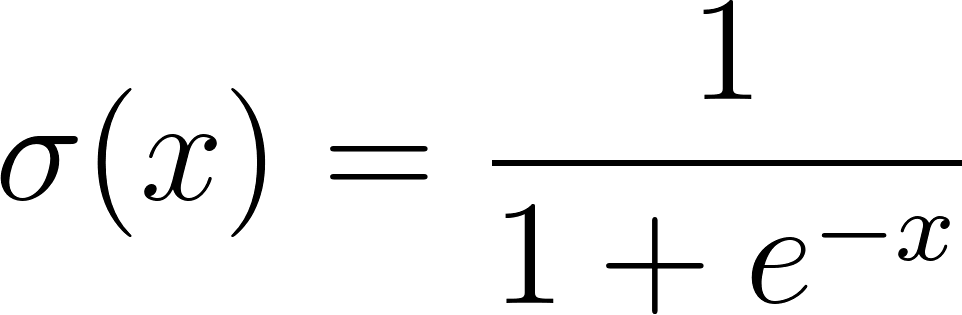


תמונה 2: A deep learning framework for financial time series using stacked autoencoders and long short term memory

נתאר את התרשים , את הכמתים והפונקציות המופיעות בו:

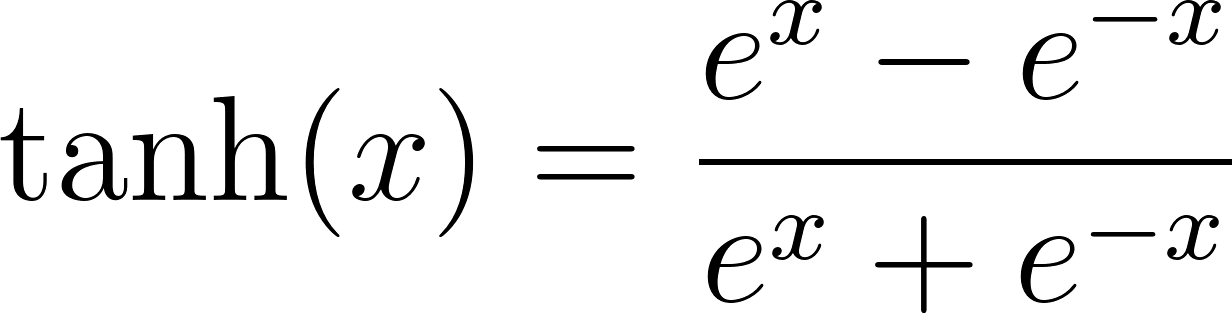
**פונקציית הסיגמואיד** [](https://www.codecogs.com/eqnedit.php?latex=%5Csigma#0) היא פונקציה המחזירה ערכים בין 0 ל-1 ומשמשת לעיתים קרובות במודלים מתמטיים ובלמידת מכונה להערכת הסתברויות. הפונקציה מתאימה לשימוש ברשתות נוירונים מכיוון שהיא יכולה להמיר טווח גדול של ערכים לטווח קטן ונשלט של ערכים.

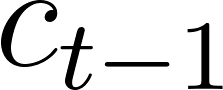
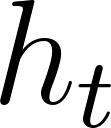
נוסחת פונקציית הסיגמואיד היא:

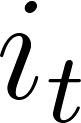
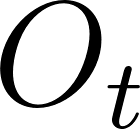
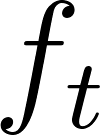
[](https://www.codecogs.com/eqnedit.php?latex=%5Csigma(x)%20%3D%20%5Cfrac%7B1%7D%7B1%20%2B%20e%5E%7B-x%7D%7D#0)

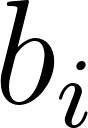
**פונקציית הטנגנס ההיפרבולי** [](https://www.codecogs.com/eqnedit.php?latex=%5Ctanh#0) היא פונקציה שמחזירה ערכים בין -1 ל-1. היא משמשת בעיקר במודלים מתמטיים ובלמידת מכונה לשימושים שדורשים נורמליזציה של ערכי קלט לטווח ממוזער יותר של ערכים.

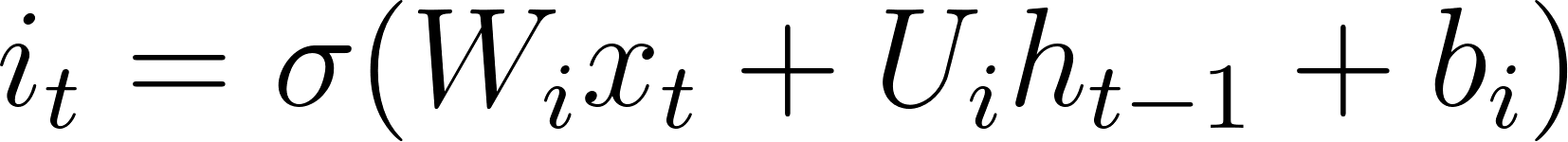
נוסחת פונקציית הטנגנס ההיפרבולי היא:

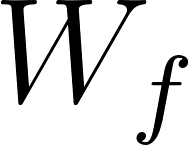
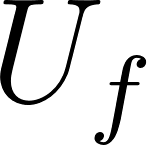
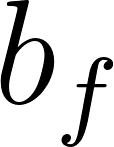
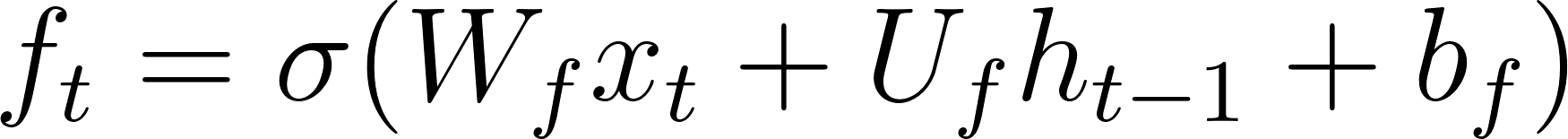
[](https://www.codecogs.com/eqnedit.php?latex=%5Ctanh(x)%20%3D%20%5Cfrac%7Be%5E%7Bx%7D%20-%20e%5E%7B-x%7D%7D%7Be%5E%7Bx%7D%20%2B%20e%5E%7B-x%7D%7D#0)

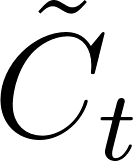
כפי שמוצג באיור למעלה, הקלטים לתא ה LSTM בזמן t הם [](https://www.codecogs.com/eqnedit.php?latex=h_%7Bt-1%7D#0) הזיכרון מהתא הקודם, מצב התא הקודם [](https://www.codecogs.com/eqnedit.php?latex=c_%7Bt-1%7D#0) והקלט הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=x_t#0) והפלט כמובן הוא הזיכרון הנוכחי (נקרא גם מצב חבוי) [](https://www.codecogs.com/eqnedit.php?latex=h_t#0) ומצב התא הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=c_t#0) שמחושבים באופן שמתואר בהמשך.

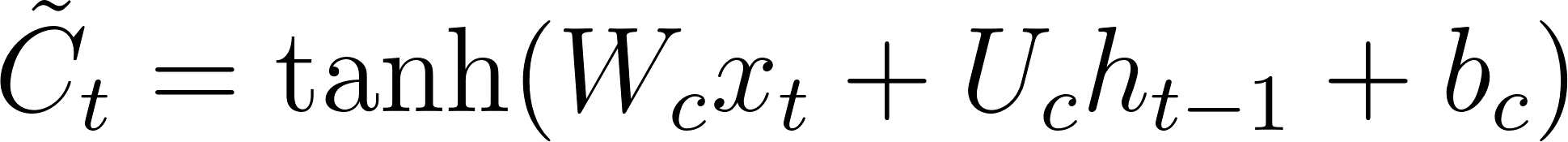
שער הקלט (Input gate) מסומן כ [](https://www.codecogs.com/eqnedit.php?latex=i_t#0), כלומר שער הקלט בזמן t , שער הפלט (output gate) [](https://www.codecogs.com/eqnedit.php?latex=O_t#0) ושער השכחה (Forget gate) [](https://www.codecogs.com/eqnedit.php?latex=f_t#0) מרכיבים את תא ה-LSTM הבודד.

**שער הקלט:** שער זה אחראי לשליטה בנתונים שמוזנים אליו מהקלט. כדי לסנן קלט מסוים, משתמשים בפונקציית הסיגמואיד. [](https://www.codecogs.com/eqnedit.php?latex=W_i#0) ו- [](https://www.codecogs.com/eqnedit.php?latex=U_i#0) הן מטריצות המשקולות ו- [](https://www.codecogs.com/eqnedit.php?latex=b_i#0) וקטור ההטייה.

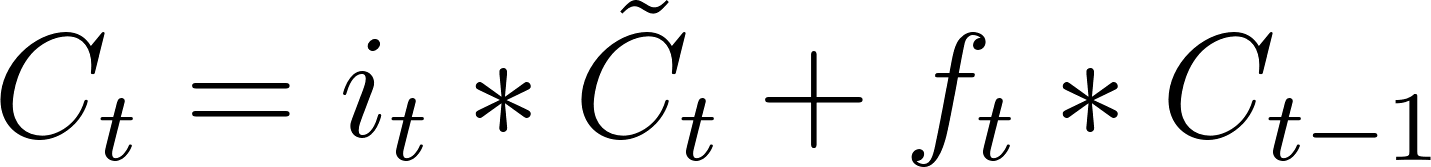
[](https://www.codecogs.com/eqnedit.php?latex=%20i_t%20%3D%20%5Csigma(W_i%20x_t%20%2B%20U_i%20h_%7Bt-1%7D%20%2B%20b_i)%20#0)

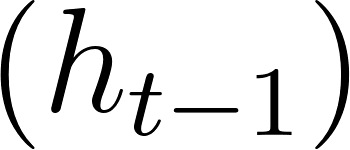
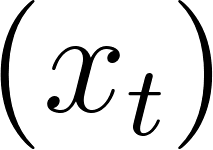
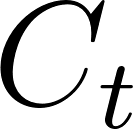
**שער השכחה:** הוא מסיר נתונים לא נחוצים מהתא. על ידי כפל של המצב החבוי בפונקציית סיגמואיד. זהו למעשה וקטור שמקבל ערכים בין 0 ל-1, שבהם 1 מציין "שמור הכל" ו-0 מציין "שכח הכל", כך מידע שאינו חשוב או נדרש להבנת ה-LSTM מוסר. שלב זה קריטי לאופטימיזציה של ביצועי המודל. [](https://www.codecogs.com/eqnedit.php?latex=W_f#0) ו- [](https://www.codecogs.com/eqnedit.php?latex=U_f#0) הן מטריצות המשקולות, [](https://www.codecogs.com/eqnedit.php?latex=b_f#0) הוא וקטור הטיה.[](https://www.codecogs.com/eqnedit.php?latex=f_t%20%3D%20%5Csigma(W_f%20x_t%20%2B%20U_f%20h_%7Bt-1%7D%20%2B%20b_f)#0)

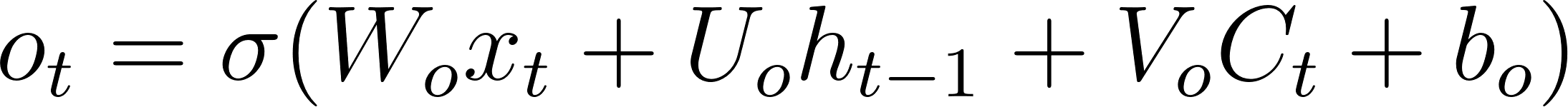
המועמד להחלפת המצב הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=%20%5Ctilde%7BC%7D_t#0) הוא וקטור המחושב על ידי פונקציית טנגנס היפרבולי על הקלט הנוכחי ומצב התא החבוי הקודם. וקטור זה מציע את העדכון האפשרי למצב התא, בהתבסס על המידע החדש והנוכחי.

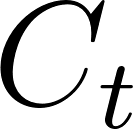
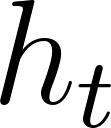
[](https://www.codecogs.com/eqnedit.php?latex=%5Ctilde%7BC%7D_t%20%3D%20%5Ctanh(W_c%20x_t%20%2B%20U_c%20h_%7Bt-1%7D%20%2B%20b_c)#0)

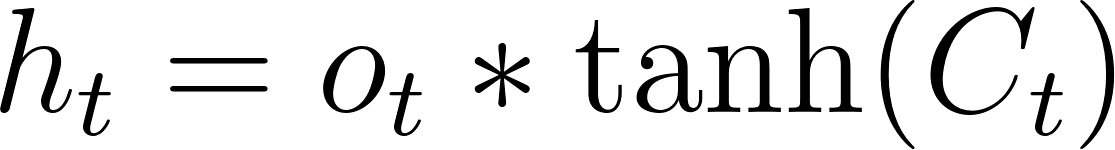
את המצב הנוכחי נחשב באמצעות המשוואה:

[](https://www.codecogs.com/eqnedit.php?latex=C_t%20%3D%20i_t%20%5Cast%20%5Ctilde%7BC%7D_t%20%2B%20f_t%20%5Cast%20C_%7Bt-1%7D#0)

**שער הפלט:** פונקציית סיגמואיד מופעלת על מצב התא החבוי הקודם [](https://www.codecogs.com/eqnedit.php?latex=%5C(h_%7Bt-1%7D%5C)#0) הקלט הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=%5C(x_t%5C)#0) והמצב הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=C_t#0)

[](https://www.codecogs.com/eqnedit.php?latex=o_t%20%3D%20%5Csigma(W_%7Bo%7D%20x_t%20%2B%20U_%7Bo%7D%20h_%7Bt-1%7D%20%2B%20V_%7Bo%7D%20C_t%20%2B%20b_o)#0)

בעזרת שער הפלט [](https://www.codecogs.com/eqnedit.php?latex=o_t#0) והמצב הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=C_t#0) נחשב את תא הזכרון (מצב התא החבוי) הנוכחי [](https://www.codecogs.com/eqnedit.php?latex=h_t#0):

[](https://www.codecogs.com/eqnedit.php?latex=h_t%20%3D%20o_t%20%5Cast%20%5Ctanh(C_t)#0)

### 5.2מימוש המודל

הפונקציה שבונה באופן מעשי את המודל לקראת השימוש בהמשך נקראת build\_lstm\_model, במהלך כתיבת הקוד, שמתי דגש על כך שהפונקציה תהיה גנרית וגמישה מספיק כדי לאפשר פיין טיונינג (fine-tuning) בהמשך. הכוונה הייתה לבנות פונקציה שניתן לשנות ולהתאים בקלות מבלי לבצע שינויים מהותיים במבנה שלה. השתמשתי במשתנים כמו n\_neurons ו- dropout\_rate כקלטים לפונקציה, כך שיהיה ניתן לבצע ניסויים עם ערכים שונים ולשפר את ביצועי המודל על ידי התאמת ההיפר-פרמטרים בצורה מדויקת יותר בהמשך תהליך הפיתוח והאימון. גישה זו מאפשרת גמישות מקסימלית, התנסות עם מודלים שונים והגעה לתוצאות אופטימליות.



המודל LSTM ממומש בספריית Keras בה השתמשתי.

אני אתאר את הפרמטרים העיקריים שבונים את רשת הנוירונים של המודל ומשפרים אותה:

**מספר השכבות ומספר הנוירונים בכל שכבה:** כל שכבה מוסיפה יכולת למודל לזהות תבניות יותר מורכבות בנתונים, כאשר הנוירונים הם היחידות שמבצעות את החישובים ברשת. מספר הנוירונים בכל שכבה משפיע ישירות על יכולת הלמידה של המודל. ככל שיש יותר נוירונים, המודל יכול ללמוד יותר מידע מהנתונים, אך גם נהיה יותר מורכב ודורש משאבים גדולים יותר לאימון.

בבניית המודל, כל שכבה חוץ מהשכבה האחרונה מוגדרת עם האפשרות return\_sequences=True, מה שמבטיח שכל שכבה תעביר את המידע המלא שלה לשכבה הבאה. השכבה האחרונה לעומת זאת מוגדרת עם return\_sequences=False, כך שהיא מעבירה רק את התוצאה הסופית לשכבת הפלט.

**Dropout** היא טכניקה שנועדה למנוע overfitting ברשתות נוירונים, ובמיוחד ברשתות עמוקות כמו LSTM, מתרחש overfitting כאשר המודל "לומד" את הנתונים ברמה מאוד מדויקת, כולל את הרעש והפרטים הלא חשובים, מה שמוביל לכך שהמודל לא מצליח לבצע תחזיות טובות על נתונים חדשים.

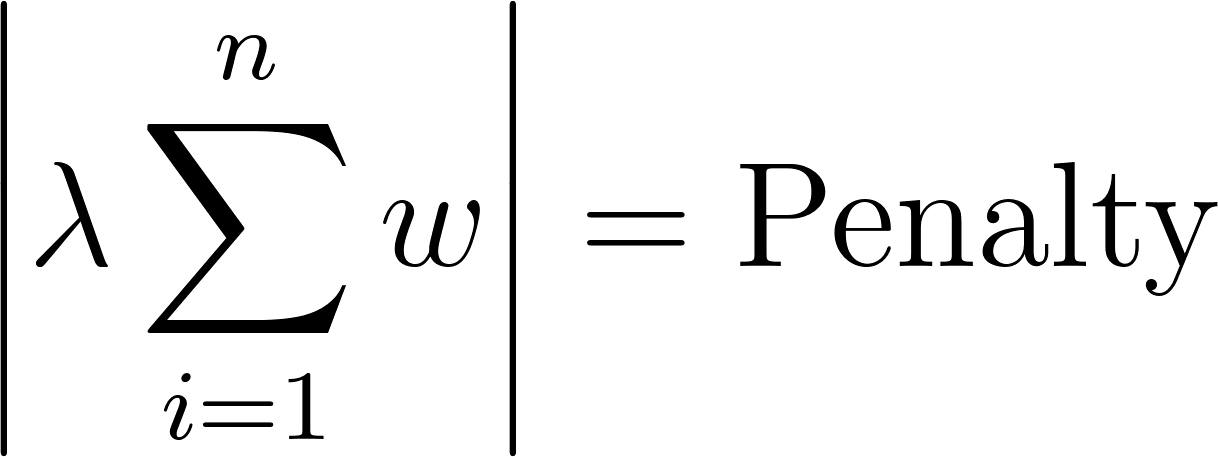
Dropout נועד להתמודד עם בעיה זו על ידי כיבוי אקראי של אחוז מסוים מהנוירונים במהלך כל שלב של אימון. כלומר, בכל שלב של העברת הנתונים , אחוז מוגדר מראש של נוירונים לא ישתתף בחישוב, והמשקלים שלהם לא יעודכנו במהלך ה-backpropagation (התהליך בו המודל מתאמן ומעדכן את מטריצות המשקולות ווקטורי ההטייה). כך, המודל לומד להיות פחות תלוי בנוירונים ספציפיים ומחזק את היכולת הכללית שלו ללמוד דפוסים בצורה רחבה יותר.

dropout\_rate בפונקציה מייצג את האחוז מהנוירונים שמכובים באופן אקראי בכל פעם. לדוגמה, אם ה-dropout\_rate מוגדר כ-0.2, זה אומר שבכל שלב של אימון, 20% מהנוירונים בשכבה מסוימת "יכובו" ולא ישתתפו בתהליך החישוב.

**רגולריזציה (Regularization)** היא טכניקה שמטרתה להקטין את הסיכוי ל-overfitting בעת אימון מודלים של למידת מכונה, בעיקר כאשר יש למודל כמות גדולה של פרמטרים. במצב של overfitting, המודל לומד לא רק את הדפוסים המהותיים בנתונים אלא גם את הרעש והפרטים הבלתי רלוונטיים שמופיעים בדגימה ספציפית של הנתונים. במילים אחרות, המודל עלול להיות "מותאם מדי" לנתוני האימון ולא לבצע תחזיות טובות על נתונים חדשים שלא ראה קודם לכן.

המטרה של רגולריזציה היא להוסיף מגבלה כלשהי למודל כך שהלמידה תתמקד במאפיינים המהותיים והכלליים של הבעיה, ולא בפרטים ספציפיים וייחודיים לסט האימון. רגולריזציה משיגה זאת על ידי הוספת עונש למשקלים גבוהים מדי של המודל, ובכך מגבילה את יכולתו ללמוד תבניות יתר.

רגולריזציית L1 מוסיפה עונש למודל שמתבסס על הערכים המוחלטים של המשקלים שלו. העונש מחושב בצורה הבאה:

[](https://www.codecogs.com/eqnedit.php?latex=%5Cleft%7C%20%5Clambda%20%5Csum_%7Bi%3D1%7D%5E%7Bn%7D%20w%20%5Cright%7C%20%3D%20%5Ctext%7BPenalty%7D#0)

כאשר [](https://www.codecogs.com/eqnedit.php?latex=w_i#0) הם המשקלים של המודל ו [](https://www.codecogs.com/eqnedit.php?latex=%20%5Clambda#0) הוא הפרמטר של רגולריזציה, הידוע גם בשם קבוע רגולריזציה (Regularization Parameter), שקובע את עוצמת העונש.

בשלב הקומפילציה (compilation) של המודל, אנו מגדירים כיצד המודל יתבצע בפועל במהלך האימון, כולל פונקציית ההפסד, האופטימייזר, והמדדים להערכת המודל. בשלב זה, המודל מקבל הוראות ברורות כיצד לחשב את השגיאות ולבצע עדכונים במשקלים, בהתאם לאסטרטגיה שנבחרה.

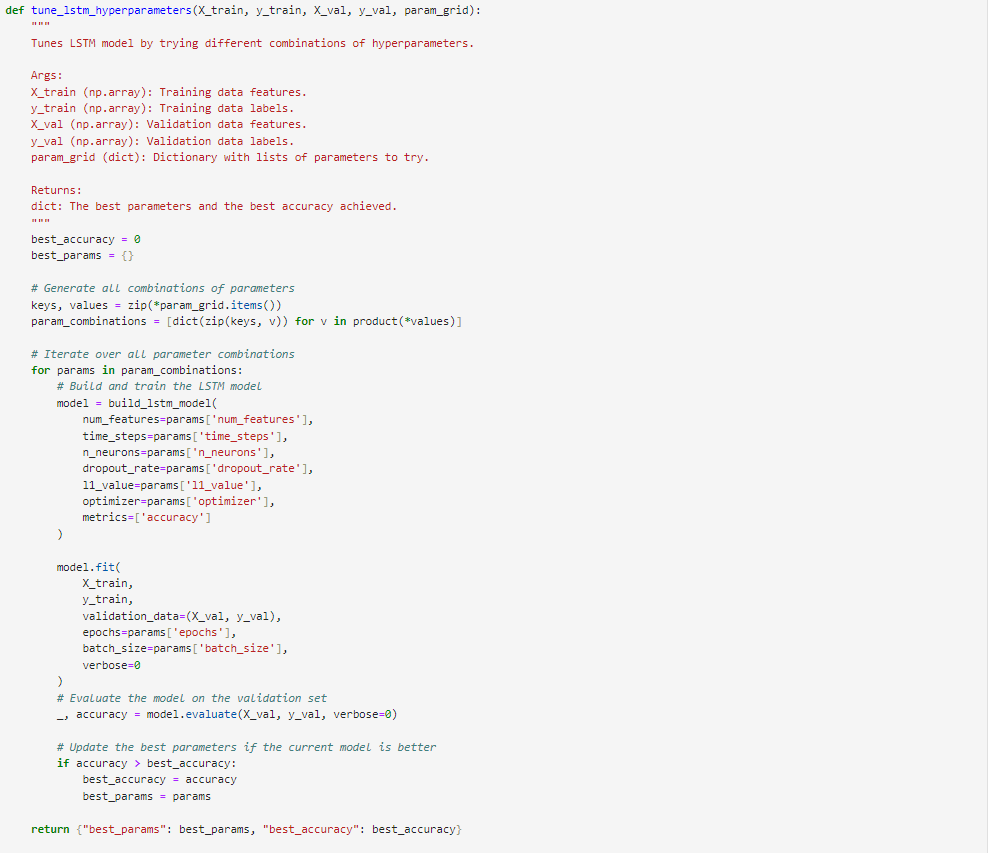
במודל זה נבחרה פונקציית ההפסד binary\_crossentropy. פונקציה זו מתאימה למשימות של סיווג בינארי, שבהן יש צורך להבדיל בין שתי קטגוריות בלבד – כמו במקרה של חיזוי אם מניה תעלה (1) או תרד (0). פונקציית binary\_crossentropy מודדת את חוסר ההתאמה בין הפלט של המודל לבין התוצאה האמיתית (label). בחירת פונקציית הפסד זו נעשתה כדי לשקף את ההסתברות של התחזית ולהעניש טעויות בהתאם. ככל שהתחזיות של המודל רחוקות יותר מהערכים האמיתיים, הערך של פונקציית ההפסד גדל, והמודל מבצע עדכונים כדי לשפר את ביצועיו. מכיוון שהפלט של המודל נע בין 0 ל-1, סף הסיווג נקבע בדרך כלל כ-0.5. כאשר הפלט גדול מ-0.5, המודל יסווג את המניה כעולה (1), ואילו אם הפלט קטן מ-0.5, המודל יסווג את המניה כיורדת (0).

### 5.3 כוונון היפר פרמטרים לשיפור המודל

במהלך הפיתוח, הושם דגש על האפשרות לכוונון היפר-פרמטרים ושיפור המודל לאורך זמן. המודל נבנה בצורה גנרית כך שניתן לבצע פיין טיונינג על ידי התאמת פרמטרים כמו מספר הנוירונים בכל שכבה, שיעור ה-dropout, מקדם הרגולריזציה והיפר-פרמטרים נוספים. גישה זו מאפשרת לבצע ניסויים רבים עם ערכים שונים ולהתאים את המודל כך שיגיע לתוצאות מיטביות, תוך שמירה על גמישות בתהליך האימון והאופטימיזציה.

הפונקצייה שמבצעת זאת בפועל נקראת tune\_lstm\_hyperparameters, בתחילת הפונקציה, נוצרות כל הקומבינציות האפשריות של היפר-פרמטרים מתוך המילון שניתן כקלט, המכיל רשימות של ערכים עבור פרמטרים שונים. לכל שילוב של פרמטרים הפונקציה בונה מודל LSTM חדש על ידי קריאה לפונקציה ייעודית שמייצרת את המודל. לאחר מכן, המודל מאומן על נתוני האימון והביצועים שלו נבדקים על סט הוולידציה.

מטרת תהליך זה היא לזהות את השילוב האופטימלי של פרמטרים כך שהמודל יוכל להכליל טוב יותר ולבצע תחזיות מדויקות יותר על נתונים שהוא לא ראה בעבר. לאחר כל סבב של אימון ובדיקה, הפונקציה משווה את הדיוק שהושג על סט הוולידציה, ובוחרת את השילוב שמביא לתוצאה הטובה ביותר. בסוף התהליך, הפונקציה מחזירה את הפרמטרים הטובים ביותר יחד עם הדיוק המרבי שהושג. תהליך כוונון זה הוא חשוב במיוחד כאשר עובדים עם מודלים מורכבים כמו LSTM, מכיוון שהבחירה הנכונה של היפר-פרמטרים יכולה לשפר משמעותית את הביצועים.

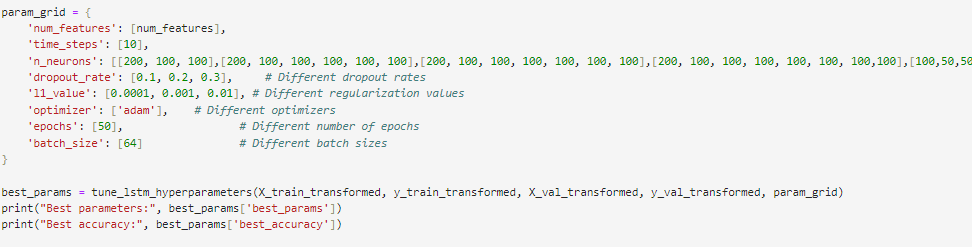


המשתנה param\_grid מכיל מילון עם כל היפר-פרמטר חשוב בתהליך האימון, ולכל פרמטר יש רשימה של ערכים אפשריים שהוא יכול לקחת. לדוגמה, לn\_neurons יש רשימה של תצורות שכבות שונות, וההגדרה הזו תאפשר לבדוק קומבינציות שונות של שכבות ומספרי נוירונים. בנוסף, ערכי ה-dropout השונים והרגולריזציה (L1) מאפשרים כוונון מדויק כדי למנוע overfitting.

לאחר מכן, הפונקציה tune\_lstm\_hyperparameters מבצעת את תהליך החיפוש על כל קומבינציה אפשרית מתוך הערכים ב-param\_grid, ובסוף היא מחזירה את השילוב הטוב ביותר של היפר-פרמטרים לפי הביצועים על סט הוולידציה.

בסופו של דבר, מוצגים הפרמטרים המובילים לצד הערך הטוב ביותר של הדיוק שהושג. תהליך זה חיוני לייעול הביצועים של מודל LSTM, במיוחד בבעיות מורכבות כמו חיזוי מניות, שבהן יש צורך בכוונון מדויק של המודל כדי להשיג תחזיות מיטביות.

בגלל שתהליך כוונון ההיפר-פרמטרים עבור מודל ה-LSTM הוא מאוד איטי, נדרש להפעיל שיקול דעת בבחירת כמות הפרמטרים שנבדקים. כל הרצה של שילוב פרמטרים לוקחת זמן רב, ולכן בחרתי מספר מצומצם של פרמטרים שניתן לבצע עליהם ניסויים מבלי שהזמן הדרוש לכך יהפוך לבלתי פרקטי. מסיבה זו, ישנם פרמטרים מסוימים שמוגדרים עם ערך אחד בלבד, כמו האופטימייזר או מספר הצעדים, כדי למנוע הרחבה נוספת של זמן ההרצה ובמקביל להתמקד בכוונון הפרמטרים הקריטיים ביותר להשגת התוצאות הטובות ביותר.



האופטימייזר שבחרתי הוא 'Adam'. הוא אחד האופטימייזרים הפופולריים ביותר בלמידת מכונה, במיוחד לרשתות נוירונים, בשל היכולת שלו לבצע אופטימיזציה יעילה ומהירה גם עבור מודלים מורכבים כמו רשתות עמוקות.

כדי להבין את היתרון של Adam, יש להבין קודם את הרעיון הבסיסי של אופטימיזציה באמצעות **ירידה גרדיאנטית** (Gradient Descent). בשיטה הבסיסית הזו, המודל מתאים את המשקלים שלו במטרה למזער את פונקציית העלות (loss function) על ידי חישוב השיפוע של הפונקציה (הגרדיאנט) ומתאים את המשקלים בהתאם לכיוון שבו השיפוע יורד (גרדיאנט שלילי). הבעיה היא שקצב הלמידה (learning rate) של הירידה הגרדיאנטית הוא קבוע, מה שעלול לגרום לבעיות – כמו שהקצב יהיה גבוה מדי ויגרום להתקדמות תנודתית, או נמוך מדי שיגרום להתכנסות איטית.

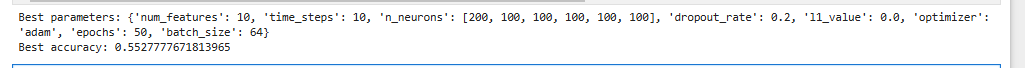
האופטימייזר **Adam** מטפל בבעיה של שיטת הירידה הגרדיאנטית בכך שהוא **מתאים את קצב הלמידה** באופן אוטומטי לכל פרמטר בנפרד. במקום להשתמש בקצב למידה קבוע לכל שלבי האימון, Adam מתחשב בשינויים ההיסטוריים של הגרדיאנטים ומבצע תיקונים בהתאם לשיפוע שנצפה בעבר. כלומר, הפרמטרים שהשינויים בהם גדולים יקבלו קצב למידה נמוך יותר, בעוד פרמטרים עם שיפועים קטנים יקבלו קצב למידה גבוה יותר. זה עוזר לאופטימייזר להתאים את עצמו לבעיות מורכבות שבהן כל פרמטר מתנהג בצורה שונה.

**Epoch** היא מעבר אחד מלא על כל סט הנתונים במהלך האימון. במהלך Epoch, המודל מעבד את כל הדוגמאות שבסט האימון פעם אחת ומעדכן את המשקלים בהתאם לחישוב השיפועים (gradients). עם כל סיום של Epoch, המודל "למד" מהנתונים וחוזר שוב על כל סט האימון לשיפור נוסף של המשקלים. הבחירה ב-Epoch=50 מצביעה על כך שהמודל יעבור על כל הנתונים 50 פעמים לאורך כל תהליך האימון. החזרה על הנתונים עוזרת למודל ללמוד את הדפוסים המרכזיים ולשפר את התחזיות.

**Batch Size** מתייחסת למספר הדוגמאות שמעובדות יחד בכל פעם לפני שהמודל מעדכן את המשקלים. כאשר ישנה כמות גדולה של נתונים, לא ניתן לעבד את כולם בבת אחת, ולכן מחלקים את הנתונים לקטעים קטנים יותר. כל קטע כזה נקרא Batch. במקרה הזה, בחירה של Batch Size=64 מציינת שכל קטע יכיל 64 דוגמאות. לאחר עיבוד של כל Batch, המודל מחשב את השיפועים ומבצע עדכון של המשקלים על סמך המידע שהתקבל מהדוגמאות ב-Batch.

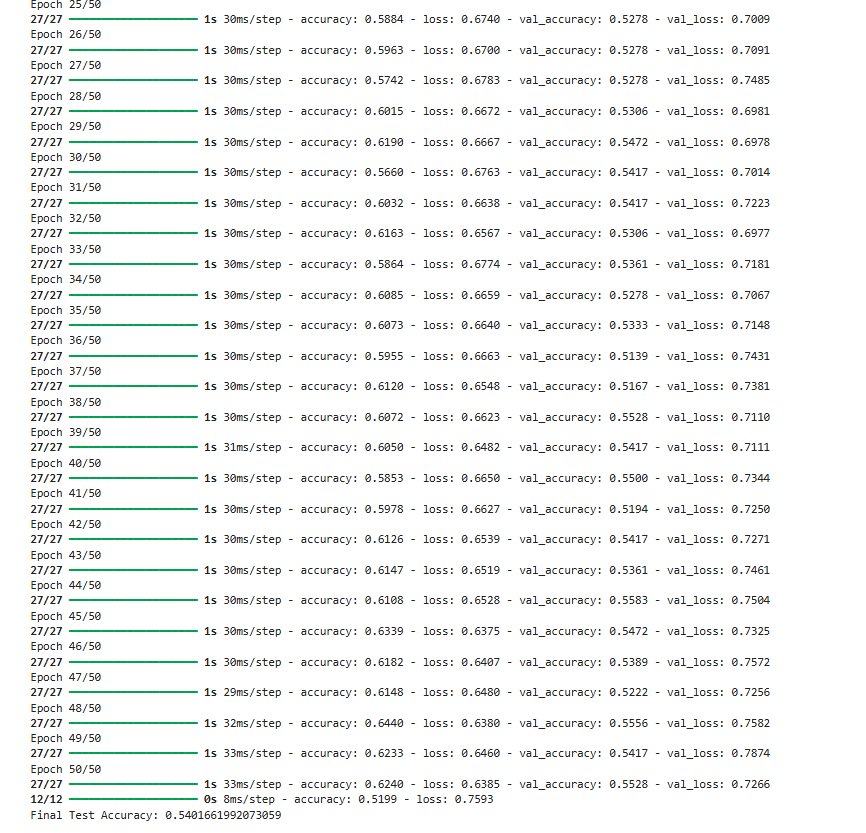
כאשר נבחר Epoch=50 ו-Batch Size=64, המשמעות היא שהמודל יעבור על כל אוסף הנתונים 50 פעמים במהלך האימון הכולל. בכל מעבר כזה, המידע מחולק לתת-קבוצות של 64 דוגמאות, ולאחר עיבוד כל Batch, מתבצע עדכון של המשקלים. כך, המודל מסוגל ללמוד בצורה הדרגתית ומאוזנת מהנתונים, תוך כדי חישוב ועדכון המשקלים לאחר עיבוד כל תת-קבוצה של נתונים.

לבסוף אחרי ההרצה הפרמטרים שהובילו לתוצאות הטובות ביותר כללו 10 ארכיטקטורת רשת עם 6 שכבות נסתרות שבראשונה יש 200 נוירונים ובשכבות הבאות 100 נוירונים, שיעור ה dropout של 0.2 ומקדם רגולרזציה 0 כלומר בהינתן מקדם רגולריזציה חיובי התוצאות נפגעות ולכן לא מתבצעת רגולריזציה.



## הערכת ביצועים של המודל

כעת אני אשתמש בתוצאות שהתקבלו בשלב הקודם ואאמן את המודל בעזרת הפרמטרים שנבחרו , נבחן את תהליך האימון



התמונה מציגה רישום של תהליך אימון מודל למידת מכונה במהלך 50 אפוקים (Epoch), כאשר כל אפוק מציג מדדים שונים המתייחסים לביצועי המודל על נתוני אימון ונתוני אימות. העמודות בתמונה מכילות מידע על דיוק המודל וערכי הפסד בשני סוגי הנתונים(קבוצת אימון וקבוצת אימות):

1. **accuracy** - מציין את דיוק המודל על נתוני האימון ב-epoch מסוים. דיוק זה מודד את היחס בין מספר התחזיות הנכונות לכלל התחזיות שביצע המודל, מה שמאפשר הערכת יעילות המודל בזיהוי דפוסים נכונים בנתונים שעליהם הוא אומן.
2. **loss** - מציין את ערך ההפסד של המודל על נתוני האימון, שמחושב בהתאם לפונקציית ההפסד שהוגדרה לאימון (binary\_crossentropy). ערך זה מעיד על מידת השגיאה של המודל בחיזוי הנתונים ומשמש כמדד לאיכות האימון.
3. **val\_accuracy** ו-**val\_loss** - שני המדדים הללו מתייחסים לביצועי המודל על נתוני האימות. דיוק האימות (val\_accuracy) מספק אינדיקציה על יכולת המודל להכליל את מה שלמד על נתונים חדשים שלא הוצגו לו במהלך האימון. ערך ההפסד על נתוני האימות (val\_loss) משקף את השגיאה של המודל כאשר הוא מוערך על נתונים אלה.

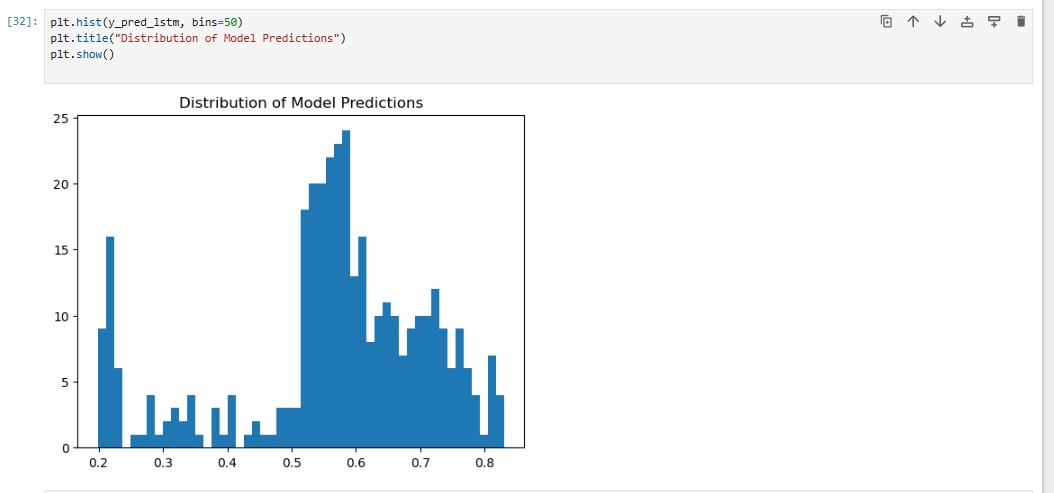
במהלך ה-50 epoch, ניתן לראות כיצד המודל משתפר, עם זינוקים וירידות בדיוק ובהפסד, שמעידים על תהליך אופטימיזציה שמבקש למצוא את האיזון הנכון בין למידה מעמיקה להימנעות מהתאמה יתר. בסיום התהליך, דיוק המודל על נתוני הבדיקה הסופיים הוא 54.16%, שמראה על יכולת מסוימת של המודל להכליל ולפעול בסביבה לא ידועה.

בנוסף נערכה הערכה של המודל באמצעות טכניקת מטריצת בלבול (confusion matrix). מטרת המטריצה היא להציג באופן ויזואלי ומספרי את התוצאות של החיזויים שביצע המודל לעומת התוצאות האמיתיות, כך שניתן להבחין במספר התחזיות הנכונות והשגויות.

המטריצה מכילה ארבעה ערכים מרכזיים:

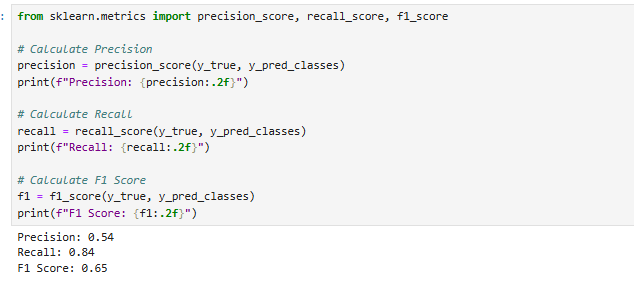
* **True Negative-TN**: המקרים בהם המודל חזה בצורה נכונה שמחיר המניה לא יעלה. במקרה זה, 38 התחזיות היו נכונות.
* **False Positive-FP**: המקרים בהם המודל טעה וחזה שמחיר המניה יעלה, אך למעשה הוא ירד. כאן נרשמו 135 טעויות.
* **False Negative-FN**: המקרים בהם המודל טעה וחזה שמחיר המניה לא יעלה, אך הוא עלה. נמדדו 31 טעויות כאלו.
* **True Positive-TP**: המקרים בהם המודל חזה בצורה נכונה שמחיר המניה יעלה. המודל הצליח לחזות נכונה 157 פעמים.

תוצאות המטריצה מצביעות על יכולת חיזוי מסוימת של המודל, עם נטייה מסוימת לטעויות מסוג שלילי כוזב (FP) וחיובי כוזב (FN) שצריך לקחת בחשבון. חשוב להדגיש שבעולם הפיננסי, חיזוי מחירי מניות נתון להשפעות רבות חיצוניות ולא צפויות תמיד, מה שמוסיף קושי בהגעה לדיוק גבוה יותר.



מוצג גרף היסטוגרמה של התפלגות תחזיות המודל עבור נתוני הבדיקה. כל עמודה בגרף מייצגת את כמות התחזיות בטווח מסוים של ערכי ההסתברות שהמודל החזיר. ההסתברות מתייחסת לסבירות שהמודל מייחס לכך שמחיר המניה יעלה (למשל, הסתברות של 0.8 שווה לחזון שיש 80% סבירות שמחיר המניה יעלה).

בהסתכלות על ההיסטוגרמה, ניתן לראות שהתפלגות התחזיות מתמקדת בעיקר בטווחים שבין 0.5 ל-0.8, עם עמודה בולטת בטווח ה-0.5-0.6, שמעידה על כך שלמודל יש נטייה לחזות עלייה במחירי המניות בהסתברות גבוהה בטווח זה. העובדה שיש מספר נמוך של תחזיות עם הסתברות נמוכה מ-0.5 מרמזת שהמודל מציג אופטימיות כללית לגבי העלייה במחירי המניות. המידע הזה חשוב להבנת כיצד המודל מתנהג וכיצד הוא נוטה להעריך את השוק.



לצורך ההערכה נבצע חישוב של שלושה מדדים סטטיסטיים חשובים להערכת ביצועי מודלים בלמידת מכונה: דיוק (Precision), רקול (Recall), ו-F1 Score. אסביר את כל אחד מהמושגים ומשמעות הערכים שנמצאו:

1. **Precision (דיוק)** - מדד זה מבטא את היחס בין מספר החיוביים האמיתיים (True Positives) שהמודל זיהה לבין כלל המקרים שבהם המודל זיהה כחיוביים (סך החיוביים האמיתיים והחיוביים הכוזבים, True Positives + False Positives). דיוק של 0.54 מציין ש-54% מהמקרים שהמודל זיהה כחיוביים היו אכן חיוביים אמיתיים.
2. **Recall (רקול או רגישות)** - מדד זה מבטא את היחס בין מספר החיוביים האמיתיים שהמודל זיהה לבין כלל החיוביים האמיתיים בנתונים (True Positives + False Negatives). רקול של 0.84 מציין שהמודל הצליח לזהות 84% מכלל החיוביים האמיתיים שבנתונים.
3. **F1 Score (ניקוד F1)** - מדד זה הוא הממוצע ההרמוני של הדיוק והרקול. הוא מספק איזון בין הדיוק והרקול ומתאים במיוחד כאשר יש צורך להתחשב בשני המדדים במידה שווה. ניקוד F1 של 0.65 מרמז שהמודל מאזן יחסית טוב בין הדיוק לרקול, אך עדיין יש מקום לשיפור בהתחשב בשיעורי הדיוק והרקול בנפרד.

הערכים הללו משמשים להערכת איכות המודל בזיהוי האירועים או התופעות הרלוונטיות, במקרה זה חיזוי עלייה או ירידה במחירי מניות. הציון הנמוך יחסית של הדיוק מעיד על כך שישנם לא מעט חיוביים כוזבים, כלומר תחזיות שגויות של המודל לכך שמחיר המניה יעלה. הרקול הגבוה יותר מראה שהמודל טוב באופן יחסי באיתור מקרים שבהם המחיר עולה.

המודל שפותח הוא כלי שימושי להערכת תנודות בשוק המניות, והביצועים שלו מעידים על פוטנציאל גבוה לשימוש בטכנולוגיות למידה עמוקה בתחום זה, תוך קבלת הבנה שהתוצאות עלולות להיות מושפעות מגורמים שאינם תמיד נשלטים על ידי המודל.

## סיכום והצעות למחקר עתידי

הפרוייקט בחן את האפשרויות והאתגרים בחיזוי מניות באמצעות שיטות מתקדמות של למידת מכונה. דגש נוסף הושם על חשיבות הפרמטרים הטכניים השונים המשפיעים על תנודות מחירים בשוק ההון. המחקר מתאר כיצד תנודות אלה נמדדות בזמן אמת ומספק תובנות על היכולת לחזות מועדי עלייה או ירידה במחירים עתידיים של מניות.

הדיוק שהושג על נתוני האימות היה כ-63%, ובסיום האימונים, על נתוני הבדיקה, המודל הציג דיוק של כ-54%. תוצאה זו מדגימה את האתגרים הגדולים הטמונים בחיזוי הפיננסי, תחום בו קיימים גורמים חיצוניים ובלתי צפויים שיכולים להשפיע על מחירי המניות. למרות זאת, הדיוק שהושג מהווה תוצאה מעודדת, שכן הוא מעיד על יכולת המודל להתמודד עם משימה מורכבת ולהציג ביצועים סבירים בהינתן הגבלותיו של המודל והתחום.

התובנות שהתקבלו מהמחקר מציעות בסיס לשיפורים עתידיים ולהרחבת המחקר לשלב מודלים נוספים שעשויים להוביל לשיפור ביצועי המודל. כמו כן, הממצאים מצביעים על הפוטנציאל של שימוש ברשתות נוירוניות לחיזויים פיננסיים, תוך הכרה במגבלות ובאתגרים שהתחום מציב בפנינו.

## מקורות

1. Muhammed, A., Olaosun, K. S., Popoola, S. J., & Byers, J. W. (2024). Deep Learning in Finance: Time Series Prediction. Preprints. <https://doi.org/10.20944/preprints202408.0836.v2>
2. Kuber, V., Yadav, D., & Yadav, A. K. (2022). Univariate and Multivariate LSTM Model for Short-Term Stock Market Prediction. arXiv preprint arXiv:2205.06673.‏
3. Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PloS one, 12(7), e0180944.‏
4. Zhang, C., Zhou, Z., & Wu, R. (2024). Analyzing and Predicting Financial Time Series Data Using Recurrent Neural Networks. Journal of Industrial Engineering and Applied Science, 2(4), 15–21. <https://doi.org/10.5281/zenodo.12786717>
5. Bhandari, H. N., Rimal, B., Pokhrel, N. R., Rimal, R., & Dahal, K. R. (2022). LSTM-SDM: An integrated framework of LSTM implementation for sequential data modeling. Software Impacts, 14, 100396.‏
6. Understanding LSTM networks. Understanding LSTM Networks -- colah’s blog. (n.d.). https://colah.github.io/posts/2015-08-Understanding-LSTMs/