**פרוייקט למידת מכונה**

**חלק ב'**



**עמית אורן 315142190**

**עדן כהן 207696246**

**קבוצה 14**

תוכן עניינים

[Initial Setup 3](#_Toc202353731)

[Model Training 3](#_Toc202353732)

[Decision Trees 3](#_Toc202353733)

[Hyperparameter Tuning 3](#_Toc202353734)

[ביצועי המודל (AUC-ROC) 4](#_Toc202353735)

[פירוש והבנת המודל 4](#_Toc202353736)

[Artificial Neural Networks 5](#_Toc202353737)

[MLP Default Training 5](#_Toc202353738)

[Hyperparameter Tuning 6](#_Toc202353739)

[SVM 8](#_Toc202353740)

[Hyperparameter Tuning וביצועי המודל 8](#_Toc202353741)

[משוואת ההחלטה של מודל SVM לינארי 9](#_Toc202353742)

[Clustering 9](#_Toc202353743)

[Evaluation 10](#_Toc202353744)

[Model choosing 10](#_Toc202353745)

[Roc Auc Graph 11](#_Toc202353746)

[Confusion Matrix 11](#_Toc202353747)

[Summary 11](#_Toc202353748)

[Improvements 12](#_Toc202353749)

[Dataset Creation 12](#_Toc202353750)

[Test 12](#_Toc202353751)

# 

# Initial Setup

* שיטת הוולידציה שבחרנו בחלק הראשון היא Cross Validation - K folds.

הנתונים יחולקו ל-10 קבוצות fold באמצעות StratifiedKFold, תוך שימוש   
בrandom state = 120. הגדרת random state נועדה להבטיח הדירות מלאה – כך שכל הרצה של הקוד תניב בדיוק את אותה חלוקה. בכל איטרציה, המודל יאומן על 9 קבוצות ויבחן על הקבוצה העשירית, בתהליך שחוזר על עצמו עבור כל אחת מהקבוצות. שיטת חלוקה זו תוביל אותנו לאורך הפרויקט.

* Data leakage הוא מצב שבו המודל נחשף למידע מסט הבדיקה בזמן האימון, מה שעלול להוביל להערכת ביצועים שגויה. כדי להימנע מכך, השתמשנו ב־Pipeline שמבצע חילוץ פיצ'רים ונורמליזציה של פיצ'רי ה־Ngram — רק על סט האימון בכל fold של ה־Cross-Validation. לאחר מכן אותם טרנספורמציות מוחלות בנפרד על סט הוולידציה. כך הבטחנו שכל הערכת ביצועים מתבצעת באופן הוגן, מבלי לחשוף מידע מסט הבדיקה במהלך האימון. (פיצ'רי הNgram הם היחידים שעברו נורמליזציה ולכן רק אותם נרמלנו מחדש)

# Model Training

## Decision Trees

### Hyperparameter Tuning

בחרנו ב-Randomized Search CV עם 300 איטרציות מתוך 672 קונפיגורציות אפשריות. המספר הזה מאפשר כיסוי רחב של מרחב ההיפר-פרמטרים תוך חיסכון בזמן חישוב. השיטה מאפשרת לאתר קונפיגורציה מוצלחת מבלי לבדוק כל אפשרות, ומספקת איזון בין דיוק לביצועים חישוביים. היפר-הפרמטרים שבחרנו לבדוק הם:

* **: Max Depth** עומק מירבי לעץ. עומק גבוה מאפשר למודל ללמוד מבנים מורכבים, אך עלול להוביל להתאמת יתר. כללנו גם ערכים נמוכים וגם ללא הגבלה (None) כדי לבדוק את נקודת האיזון האופטימלית. טווח הערכים שבדקנו : [None, 5, 10, 15, 20, 30, 50].
* **Min Sample Split** : המספר המינימלי של סמפלים לפיצול בצומת העץ. ערכים קטנים מאפשרים פיצולים מרובים ודקים יותר לעומת ערכים גבוהים התורמים לפשטות המודל. טווח הערכים שבדקנו : [2, 5, 10, 20].
* **Min Sample Leaf** : כמות תצפיות מינימלית ליצירת פיצול (עלה). אם מספר התצפיות בקטגוריה נמוך ממספר זה, לא ייווצר פיצול בענף זה. טווח הערכים שבדקנו : [1, 2, 4, 10].
* **Max Features :** קובע כמה פיצ'רים ייבחנו בכל צומת. הפחתת מספר הפיצ'רים הנבחנים עשויה להפחית overfitting ולהאיץ את זמן האימון. הערכים שבחרנו : [None, 'sqrt', 'log2'] .
* **Criterion :** מייצג את קריטריון החישוב על העץ שני הקריטריונים שבחרנו לבדוק מחשבים את ה"רווח" מהמידע שיש בידינו בכל צומת בעץ בצורה שונה, ומחשבת עץ שונה בהתאם. הערכים שבחרנו : [gini, entropy] .

### ביצועי המודל (AUC-ROC)

* criterion= entropy , max depth=10, min samples split=20, min samples leaf=10,   
  max features=None
* AUC-ROC ממוצע על סט האימון: 0.9663
* AUC-ROC ממוצע על סט הוולידציה: 0.9582

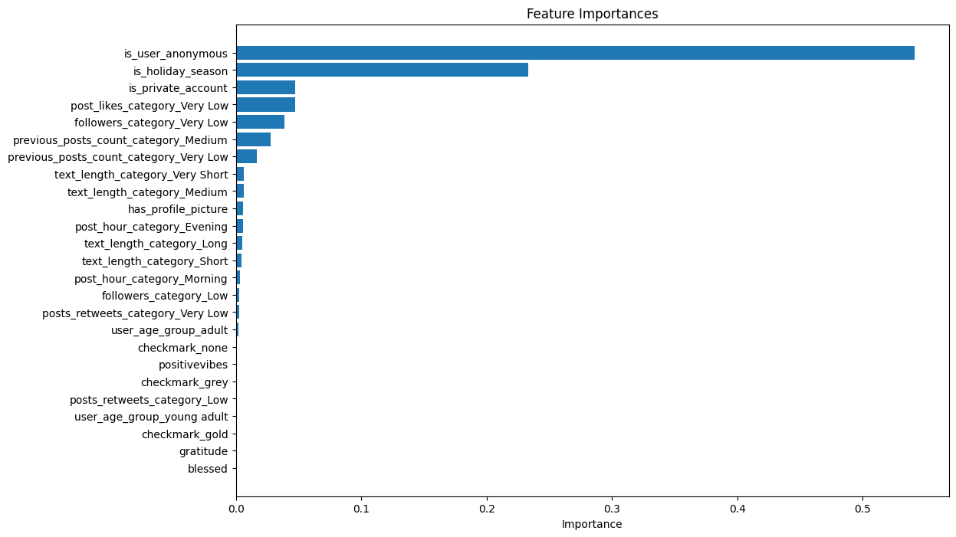
המודל השיג AUC גבוה (מעל 0.95) גם באימון וגם בוולידציה, עם פער קטן שמעיד על הכללה טובה. התוצאות היו עקביות בכל הקיפולים, והיפר-פרמטרים זהים נבחרו בכל פעם – מה שמעיד על יציבות המודל ודיוק תהליך הכוונון.

### פירוש והבנת המודל

**Interpretability** **:** אחד היתרונות המרכזיים של מודל עץ החלטה הוא יכולת הפירוש הגבוהה (interpretability) - כלומר, היכולת להבין כיצד המודל מקבל החלטות בפועל.  
במשימת חיזוי סנטימנט, תכונה זו קריטית: היא מאפשרת לנו את ההבנה מה משפיע על סיווג פוסט כחיובי או שלילי - דבר חשוב במיוחד בקונטקסטים שונים כמו שיווק, ניתוח תגובות או ניטור תוכן.  
בגרף של שלוש השכבות העליונות של עץ ההחלטה ניתן לזהות דפוסים מרכזיים שמתיישבים עם הממצאים שהתקבלו בשלבים הקודמים של הפרויקט. תכונות כמו מאפייני המשתמש, מצב החשבון ועונתיות חוזרות כפקטורים משמעותיים בהחלטת המודל, ומחזקות את הקשרים שנצפו בין משתנים אלו לבין סנטימנט הפוסט בניתוחים הקודמים.

תמונה שמכילה תרשים, תוכנית, טקסט, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

**Feature Importances :** בהתבסס על גרף חשיבות הפיצ'רים, עולה כי is\_user\_anonymous ו־is\_holiday\_season הן התכונות המרכזיות שמשפיעות על החלטות המודל, ובהתאמה מופיעות גם בצמתים הראשיים של עץ ההחלטה. תכונות נוספות כמו is\_private\_account ו־ post\_likes\_category תורמות גם הן, אם כי במידה פחותה.  
קיימת הלימה בין הפיצ'רים החשובים לבין מבנה העץ, מה שמעיד שהמודל מתבסס על מאפיינים מהותיים במשימת חיזוי הסנטימנט. בנוסף, ניתן לשקול צמצום תכונות בעתיד, בהתבסס על מידת התרומה הזניחה של הפיצ'רים בתחתית.

## Artificial Neural Networks

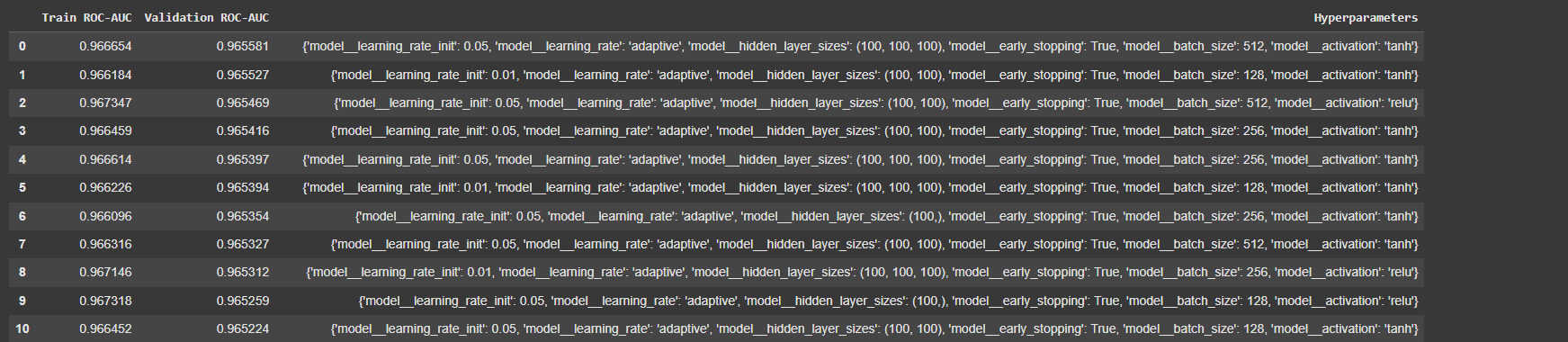
### MLP Default Training

אימנו מודל MLP באמצעות ספריית scikit-learn, תוך שימוש בערכי ברירת המחדל של המחלקה MLPClassifier. מבנה המודל כולל שכבת קלט עם 25 נוירונים – אחד עבור כל פיצ'ר בדאטה, שכבה חבויה אחת בלבד הכוללת 100 נוירונים, ושכבת פלט המתאימה לבעיה (למשל, נוירון יחיד במקרה של סיווג בינארי). ברירת המחדל של פונקציית האקטיבציה בשכבות החבויות היא ReLU, אשר מוסיפה לרשת יכולת למידת תבניות לא -ליניאריות. עם זאת, בשכבת הפלט נעשה שימוש אוטומטי בפונקציית סיגמואיד - logistic לצורך הפקת הסתברות בין 0 ל־1, דבר שמתאים לבעיות סיווג בינארי. שילוב זה מאפשר למודל ללמוד ייצוגים פנימיים בצורה פשוטה ויעילה, תוך שמירה על פלט פרשני בצורת הסתברות.

* AUC-ROC ממוצע על סט האימון: 0.9700
* AUC-ROC ממוצע על סט הוולידציה: 0.9648

מדובר בתוצאות גבוהות מאוד וקרובות זו לזו, מה שמעיד על כך שהמודל הצליח ללמוד את הנתונים בצורה טובה תוך שמירה על יכולת הכללה לנתונים שלא נראו באימון. הפער הקטן בין תוצאות האימון לוולידציה מצביע על כך שאין סימנים ברורים ל־overfitting . לכן, ניתן להסיק שהמודל מתכנס היטב ומפגין ביצועים יציבים ואמינים גם על דגימות חדשות.

### Hyperparameter Tuning

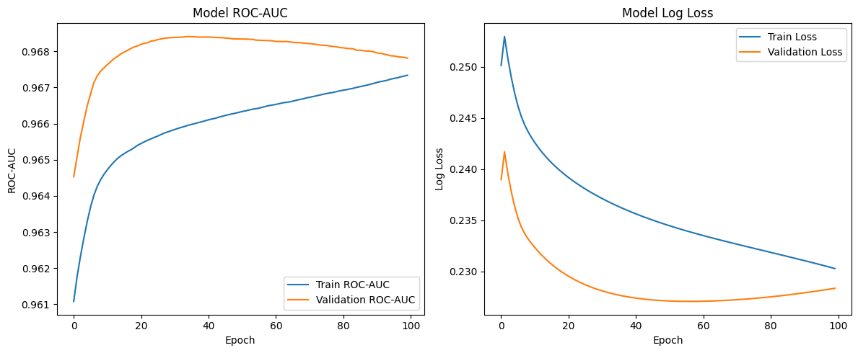
* **: Hidden layer sizes** קובע את מבנה השכבות החבויות ברשת, כמה שכבות כאלו יהיו וכמה נוירונים בכל שכבה. הגדרנו שלוש אפשרויות: שכבה אחת, שתי שכבות, ושלוש שכבות כאשר בכל אחת מהן יש 100 נוירונים. הגדלת מספר השכבות או מספר הנוירונים מגדילה את כושר הלמידה של המודל, אך גם מעלה את הסיכון ל־ overfitting ואת זמן האימון.
* **Activation** : פונקציית האקטיבציה קובעת כיצד מועבר האות בין הנוירונים. נבדקו relu - ללמידה מהירה. Logistic, tanh שמתאימות להתכנסות עדינה. הבחירה משפיעה על קצב הלמידה ויכולת המודל להתמודד עם אי-ליניאריות.
* **Learning Rate Init :** קובע את קצב הלמידה ההתחלתי. נבחנו הערכים 0.001, 0.005, 0.01, 0.05. ערך קטן מוביל ללמידה איטית אך יציבה, בעוד ערך גבוה מאיץ למידה אך עלול לגרום לאי-יציבות.
* **Learning Rate Adaptive :** מאפשר התאמה דינמית של קצב הלמידה במהלך האימון. כאשר אין שיפור, הקצב מואט, מה שתורם להתכנסות יציבה. בחרנו לקבע את היפר הפרמטר הזה ולא לכוונן בערכים שונים.
* **Batch Size :** גודל האצווה משפיע על יציבות ומהירות האימון. נבדקו ערכים של 128, 256 ו־512, המתאימים להרבה סמפלים. ערכים קטנים מזרזים עדכונים אך מכניסים הרבה רעש ולעומת זאת באצ'ים גדולים מייצבים את האימון.
* **Early Stopping :** קיבענו את היפר הפרמטר ל- True. זהו מנגנון שמפסיק את האימון מוקדם אם אין שיפור בביצועי המודל על סט הוולידציה, במטרה לחסוך בזמן אימון ולמנוע למידה מיותרת שעלולה להוביל ל־overfitting.
* **Solver :** קיבענו את היפר הפרמטר ל- SGD שיטה זו מאפשרת שליטה ישירה על קצב הלמידה באמצעות הפרמטרים learning rate ו־learning rate init . מה שמאפשר גמישות והתאמה עדינה של תהליך הלמידה.

הטבלה מציגה את ביצועי המודל (ROC-AUC) על סט האימון והוולידציה עבור עשר הקונפיגורציות עם המדד הכי גבוה של היפר-פרמטרים. כל הקונפיגורציות הניבו תוצאות גבוהות ויציבות, עם פערים קטנים מאוד בין האימון לוולידציה. דבר זה מעיד על התכנסות טובה והימנעות מ – overfitting. שלוש שכבות חבויות, פונקציית אקטיבציה מסוג tanh וקצבי למידה התחלתיים נמוכים 0.005–0.01 נמצאו כמשפיעים לטובה על ביצועי המודל.

* Learning rate init = 0.05, hidden layer sizes = [100, 100, 100],   
  batch size = 512, activation = tanh
* AUC-ROC ממוצע על סט האימון: 0.967
* AUC-ROC ממוצע על סט הוולידציה: 0.9656

מדובר בתוצאות גבוהות מאוד וקרובות זו לזו, המעידות על כך שהמודל מתכנס היטב ומצליח לבצע הכללה טובה על נתונים שלא נראו באימון, ללא סימנים ל־overfitting. חשוב לציין כי התוצאות נמוכות מהמודל הדיפולטיבי כנראה מכמה סיבות, אחת שמדובר בחיפוש רנדומלי ויכול להיות שלא הגענו לקונפיגורציה הטובה ביותר מבין כלל הקונפיגורציות הקיימות, השנייה היא שמדובר בטווח מאוד קטן 0.0006 ויכול לנבוע גם משונות סטטיסטית מקרית בין הפולדים.

נשים לב כי היפר הפרמטרים שהביאו לתוצאות הטובות ביותר היו learning rate init = 0. 05 סיפק איזון טוב בין קצב למידה מהיר לבין יציבות. המבנה העמוק של הרשת – בחירה בשלוש שכבות חבויות עם 100 נוירונים בכל אחת, מאפשר למודל ללמוד דפוסים מורכבים מהדאטה. בנוסף גודל batch גבוה של 512 סיפק איזון בין יציבות לעומק הרשת לבין קצב הלמידה הגבוה.ופונקציית ההפעלה tanh תרמה לייצוגים סימטריים ויציבים יותר בלמידה. השילוב בין הגדרות אלו הוביל לביצועים מיטביים.

הגרפים מציגים את מדדי ה־ROC-AUC וה־Loss לאורך 100 אפוקים עבור סט האימון והוולידציה. בגרפים ניתן לראות שה־ ROC-AUC בסט האימון והוולידציה עולה ומתייצב, וה־Loss יורד בהתמדה. את הקפיצה הקלה ב־Loss בתחילת האימון נייחס למשקלים אקראיים ועדכון ראשוני  
 ב־SGD , תופעה טבעית. ניתן לראות שהקווים של סט האימון והוולידציה מתקרבים משום שהמדד על סט האימון ממשיך לעלות בעוד שהמדד על סט הוולידציה מתחיל לרדת — מצב שמעיד על מגמת Overfitting בהמשך, ולכן נבחר לעצור את האימון במספר אפוקים מוקדם יותר כדי להימנע מכךOverfitting .סביב epoch 60 הביצועים מתייצבים, ולכן נקודה זו נחשבת מתאימה epoch אופטימלי

## SVM

### Hyperparameter Tuning וביצועי המודל

* Dual = False, class weight =balanced, C = 100
* AUC-ROC ממוצע על סט האימון: 0.901
* AUC-ROC ממוצע על סט הוולידציה: 0.9003

מדד ROC-AUC גבוה מעיד על יכולת הבחנה טובה של המודל בין דוגמאות חיוביות לשליליות לאורך מגוון ספים. הפער הקטן בין סט האימון לוולידציה (0.007) מרמז על כך שהמודל לא סובל מאובר-פיטינג משמעותי, ומציג יציבות טובה גם על נתונים שלא נראו באימון. בנוסף נסתכל על היפר הפרמטרים שנבחרו:

* **: C = 100** פרמטר הרגולריזציה, שולט בעוצמת הענישה שהמודל מקבל על שגיאות סיווג. ערך גבוה מעודד התאמה גבוהה לנתוני האימון על חשבון רגולריזציה חלשה יותר.
* **dual = False** **:** מגדיר שהפתרון ייעשה בצורה פרימלית (ולא דואלית), וזה מתאים במיוחד כאשר מספר הדגימות גדול ממספר הפיצ'רים.
* **class weight = balanced** **:** מתאים אוטומטית את משקל כל מחלקה לפי שכיחותה בנתונים, ובכך מתקן חוסר איזון בין הקלאסים ומשפר את רגישות המודל למחלקות נדירות.

### משוואת ההחלטה של מודל SVM לינארי

משוואת ההחלטה מאפשרת להבין אילו פיצ’רים משפיעים על הסיווג, ומה כיוון ההשפעה שלהם. כל פיצ’ר במודל מקבל משקל מספריW , כאשר ערך מוחלט גבוה מצביע על חשיבות גבוהה יותר בהחלטה, והסימן חיובי או שלילי מציין אם הפיצ’ר תורם להגדלת הסיכוי לסיווג חיובי או דווקא מקטין אותו.

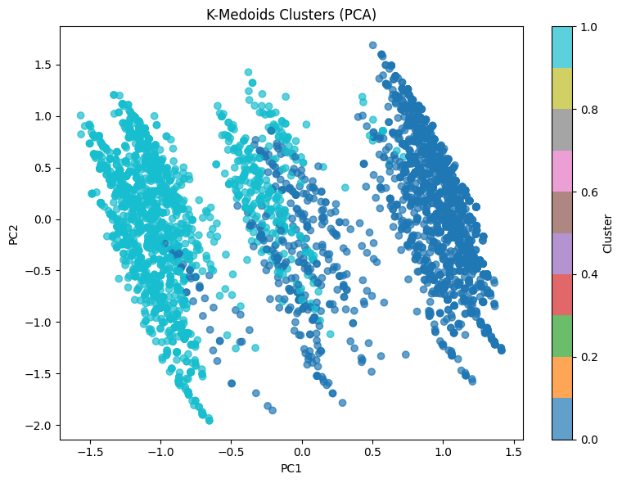
לדוגמה, הפיצ’רים positive vibes = 1.98, blessed = 2.34 קיבלו משקלים חיוביים גבוהים מאוד, מה שמעיד על כך שפוסטים המכילים ביטויים רגשיים וחיוביים תורמים במידה רבה להחלטה לסווג את המקרה כחיובי. לעומת זאת, הפיצ’רים is user anonymous = -1.07 , is holiday season = -0.85 מה שמעיד על כך שחשבונות אנונימיים, פעילות בתקופת חגים מפחיתים את ההסתברות לסיווג חיובי. משוואה זו מספקת תובנות פרשניות ברורות ותואמת חלק מהמסקנות הקודמות שעלו מהניתוח –פיצ’רים המעידים על אנונימיות או מעורבות נמוכה בעלי השפעה גבוהה על יכולות סיווג המודל . אך הפיצ'רים בעלי משקל חיובי פחות מתאימים למסקנות הקודמות שלנו מכיוון שקיבלו ציונים יחסית נמוכים בהשפעתם על יכול הסיווג במדדים אחרים.

## Clustering

Clustering היא שיטת למידת מכונה לא מונחית, שבה אין תגיות או ערכי מטרה ידועים מראש. מטרתה לזהות קבוצות טבעיות בתוך הנתונים לפי דמיון בין הדוגמאות, בניגוד למודלים מונחים שמטרתם לחזות ערכים ידועים. בנוסף, סט הנתונים שונה בכך ש- Clustering משתמש רק במאפיינים, בעוד שמודלים מונחים דורשים גם עמודת מטרה.

בחרנו להשתמש בכל הפיצ'רים מכיוון שסינון לפי שונות גרע מידע רציף שנחוץ למדדי המודל. חישבנו את מטריצת המרחק באמצעות מדד Gower, שמתאים לנתונים הטרוגניים. מדד זה יודע לנרמל את המרחקים בין ערכים רציפים, ולהשוות בין ערכים קטגוריאליים באופן הוגן כך שערכים זהים מקבלים מרחק 0 ושונים מקבלים 1. בזאת מתאפשר לשלב מידע ממקורות שונים מבלי להטות את החישוב לפי סוג מסוים של משתנים, על מנת לשלב פיצ'רים מסוגים שונים. לאחר מכן הרצנו את אלגוריתם K-Medoids עבור K בטווח 2–8.

לצורך השוואת תוצאות האישכול עבור ערכי K שונים השתמשנו בשני מדדים פנימיים. מדד Silhouette שבודק את הדמיון של כל דוגמה לאשכול שלה ביחס לאשכולות אחרים כאשר ערך גבוה יותר מעיד על פילוח איכותי, ומדד Davies-Bouldin שבוחן את היחס בין פיזור פנימי למרחק בין אשכולות כאשר ערך נמוך יותר מעיד על הפרדה טובה יותר. שני תמונה שמכילה טקסט, גופן, צילום מסך, מספר

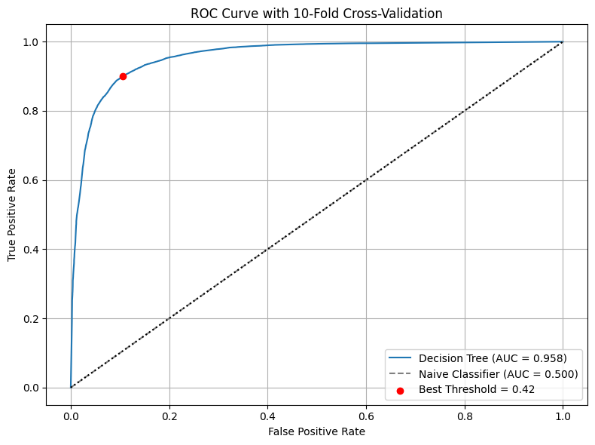
תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.מדדים אלו נבחרו כיוון שהם משלימים זה את זה ונותנים תמונה מדויקת על איכות הפילוח ללא תלות במשתנה מטרה. מבין הערכים שנבדקו (2-8) נמצא כי K=2 הוא הערך האופטימלי, עם מדד Silhouette הגבוה ביותר 0.313 ו־Davies-Bouldin הנמוך ביותר 2.02, מה שמעיד על אשכול ברור והפרדה טובה. מאחר שמשתנה המטרה הוא סנטימנט חיובי או שלילי, תוצאה זו מרמזת על כך שייתכן וקיים קשר כלשהו בין חלוקת הקבוצות לבין הסנטימנט, אך נדרש לאמת זאת מול התוויות בפועל.

הגרף מציג את תוצאות אשכולות K-Medoids עבור K=2 לאחר הקטנת ממדים באמצעות PCA לשני רכיבים עיקריים, לצורך ויזואליזציה. ניתן לראות הפרדה ברורה יחסית בין שתי קבוצות של נקודות, דבר שתואם את תוצאות המדדים ומחזק את הבחירה ב־K=2.

## Evaluation

### Model choosing

|  |  |  |  |
| --- | --- | --- | --- |
| מאפיין | Decision Tree | MPL | SVM |
| ממוצע AUC ROC אימון | 0.9663 | 0.967 | 0.901 |
| ממוצע AUC ROC ולידציה | 0.9582 | 0.9656 | 0.9003 |
| פער אימון ולידציה | קטן | קטן מאוד | קטן |
| יתרונות | אקספלינאבילי,  מתאים לדאטה לא מורכב | ביצועים מצוינים, גמישות גבוהה, מתאים לדאטה מורכב | מאפשר להבין מה משפיע על הסיווג בזכות משקלי הפיצ'רים |
| חסרונות | נוטה לoverfitting- אם לא מכוונן נכון | |  | | --- | |  |   דורש הרבה זמן אימון, קשה לפרשנות | מתקשה לתפוס קשרים מורכבים בין פיצ'רים בינאריים |

נבחר בעץ החלטה למודל הסופי. למרות שקיבל מדדי AUC ROC נמוכים מ-ANN מודל זה מאפשר רמה גבוהה של פרשנות כך שניתן להבין אילו פיצ’רים תרמו להחלטות הסיווג בנוסף עץ ההחלטה מתאפיין במהירות הרצה גבוהה הן באימון והן בתחזיות דבר שמקל על שילובו במערכות בזמן אמת. חסרון של עץ ההחלטה היא נטייה ל־overfitting כאשר אינו מכוונן כראוי, מה שעלול לפגוע בביצועיו במקרים שבהם הנתונים החדשים - test שונים מהדפוסים שעליהם אומן.

### Roc Auc Graph

עקומת ROC שהופקה באמצעות CV הציגה ערך AUC של 0.958. נקודת הסף שנבחרה היא 0.42, ונקבעה לפי מדד Youden לאיזון אופטימלי בין זיהוי חיוביים לבין מניעת חיוביים שגויים. הבחירה בסף זה ולא ב־0.5 נבעה מחוסר איזון במחלקות.

### Confusion Matrix

בהתבסס על מטריצת המבוכה שהתקבלה ניתן לראות שהמודל לא מציג הטיה משמעותית כלפי אחת המחלקות. ניתן לראות שהמודל נוטה מעט לסווג דוגמאות כ־ Positive גם כשהן שליליות False Positives - גבוהים יחסית, אך ללא הטיה קיצונית. בחרנו בסף סיווג נמוך 0.42, לפי מדד Youden כדי לאזן בין רגישות לספציפיות. מה שגורם להגדלת הרגישות והפחתת פספוס של דוגמאות חיוביות, אך יוצר יותר טעויות בזיהוי שליליים. בחרנו במדד זה כי הדאטה אינו מאוזן, יש יותר דוגמאות שליליות ולכן המודל עשוי לשגות יותר בהבחנה ביניהן.

### Summary

ניתן לראות כי הפיצ'רים הבולטים ביותר במודל עץ ההחלטה ובמודל SVM הם is\_user\_anonymous is\_holiday\_season, שנמצאו כמאפיינים המזוהים בעיקר עם פוסטים בעלי סנטימנט שלילי. לעומת זאת, במודל SVM ניתן לזהות גם פיצ'רים המאפיינים פוסטים חיוביים, כמו blessed positivevibes,מילים בעלות קונוטציה חיובית מובהקת. העובדה ששני המודלים מבליטים בעיקר פיצ'רים הקשורים למחלקה השלילית עשויה להעיד על נטייה מסוימת של המודל להתמקד בזיהוי שליליים. הסבר לכך הוא שהדאטה אינו מאוזן, יש בו יותר דוגמאות שליליות מחיוביות, ולכן המודלים לומדים טוב יותר את הדפוסים של המחלקה הדומיננטית.

לא זיהינו תופעה מובהקת של Overfitting .בכל שלושת המודלים התקבלו ערכי AUC-ROC גבוהים הן בנתוני האימון והן בוולידציה, עם פערים קטנים מאוד בין המדדים. דבר המעיד על כך שהמודלים מצליחים ללמוד היטב את הדאטה תוך שמירה על יכולת הכללה. אילו הייתה מתרחשת תופעת Overfitting ,היינו מצפים לראות פער משמעותי בין ביצועי האימון לביצועי הוולידציה. ניתן להתמודד עם תופעה זו באמצעות הפחתת מורכבות המודל, למשל על ידי הורדת מימדים – PCA , שלב שעלול להכליל את המודל אך פוגע בהסברתיות.

## Improvements

### Dataset Creation

בשלב עיבוד הנתונים ניסינו שני כיווני שיפור, במטרה לשפר את איכות הפיצ'רים שיוזנו למודל. בניסיון הראשון השתמשנו במדד Mutual Information במקום Fisher Score כדי לבחור את 25 הפיצ'רים עם התרומה האינפורמטיבית הגבוהה ביותר – מתוך כוונה לחדד את הקשר בין הפיצ'רים לתווית המטרה. בניסיון השני המרנו פיצ'רים קטגוריאליים לנומריים ונרמלנו באמצעות MinMaxScaler מתוך הנחה שייצוג כמותי אחיד עשוי לשפר את ביצועי עץ ההחלטה. בפועל, שני הניסיונות הובילו לירידה ב- ROC AUC לעומת המודל המקורי (0.9578 ,0.9328 בהתאמה). דבר מעניין היה שרוב הפיצ'רים שנבחרו היו קטגוריאליים, בעוד שהנומריים כמעט ולא נבחרו, מה שמרמז כי ההמרה לייצוג מספרי לא תרמה. לסיכום, אף אחת מהשיטות לא שיפרה את ביצועי המודל בשלב זה.

**Model Improvement**בשלב שיפור המודל החלטנו להחליף את עץ ההחלטה הבסיסי במודל Gradient Boosting מתוך מטרה לשפר את יכולת ההכללה והדיוק של המודל. בניגוד לעץ החלטה בודד, מודל Gradient Boosting בנוי מסדרה של עצים חלשים שכל אחד מהם מתמקד בטעויות של הקודמים לו. בכל שלב, המודל לומד מהשגיאות שנעשו ומשפר את התחזיות בהדרגה. השילוב הזה יוצר מודל שמסוגל להתמודד עם מבנים מורכבים, קורלציות בין פיצ'רים, ויחסים לא ליניאריים. התוצאה שהתקבלה שיפרה את ביצועי המודל במדד של AUC ROC -0.9657 בוולידציה. תוצאות אלה מצביעות על כך שהאלגוריתם המשופר מצליח לזהות את הדפוסים בדאטה בצורה טובה ומאוזנת יותר.

# Test

ביצענו עיבוד מקדים עבור סט הטסט כפי שביצענו בשלב הקודם. לאחר מכן החלנו את המודל הנבחר לקבלת סיווג עבור דגימות הטסט השונות (קובץ group14.csv)