אוניברסיטת תל אביב

הפקולטה להנדסה ע"ש איבי ואלדר פליישמן

המחלקה להנדסת תעשייה

דוח מסכם לפרויקט

בנושא סיווג מוצרים מחברת Otto Group

מוגש במסגרת הקורס "כריית ידע"

מרצה: ד"ר אייל קולמן
ע' הוראה: מר איתי מרגולין

מגישות:

לירז מאיר 204292189
חן אברהם 203446315
רותם פינצ'ובר 313245987

תקציר מנהלים

במסגרת הפרויקט פעלנו על מנת לסווג דגימות מתוך סט נתונים קיים לפי סוגי מוצרים שונים הקיימים בחברת Otto Group, תוך שימוש בערכי פיצ'רים שונים לכל דגימה בסט הנתונים.

ראשית, בחנו את סט הנתונים על מנת לקבל תמונה רחבה על אופן התפלגות הנתונים, שונות הנתונים עבור סוגי המוצרים השונים וכו'. לאחר מכן, ביצענו עיבוד מקדים על הנתונים, לדוג' הסרת תצפיות חריגות. בנוסף, פעלנו להתאמת סט הנתונים למודלים שלמדנו, המאפשרים קלסיפיקציה בינארית - זאת על ידי צמצום המידע לשני סוגי הנתונים הפופולריים ביותר בסט הנתונים. בהמשך, בחנו 4 סוגי מודלים שנלמדו במסגרת הקורס, כולל אימון המודלים, בחינתם על סט ווליד‌ציה, סיווג נתוני Test לכל מודל והשוואה בין דיוק המודלים השונים באמצעות מדד AUC ועקומת ROC.

מתוך הניתוח עלה כי המודל אשר ניבא באופן הטוב ביותר את הסיווג לנתוני ה-Train, לפי מדד AUC, הינו מודל Decision Tree, ולפיכך סווגו באמצעותו את ההסתברויות לכלל נתוני ה-Test.

הרחבה אודות שלבי הניתוח בפרויקט

לאורך ההסברים במסמך, המונחים הנ"ל שקולים: *סוג מוצר = Class, *מאפיין = Feature, *סט נתונים = Data.

שלב מקדים

בשלב Data Exploration, נאספו נתונים סטטיסטיים שונים אודות סט הנתונים -

- אחוז התצפיות מכל Class: ניתן לראות שמרבית התצפיות שייכות ל-Class 2 או ל-Class 6.
- היסטוגרמה עבור כל Feature (איור 1): ניתן לראות כי עבור כל הפיצ'רים, מרבית התצפיות הינן בעלות ערך 0.
- מטריצת קורלציה בין הפיצ'רים השונים (איור 2): ניתן לראות כי ישנם פיצ'רים אשר הקורלציה ביניהם הינה 0 (בלתי תלויים), בעוד ישנם פיצ'רים הקורלציה ביניהם גבוהה מ-0, כלומר הינם תלויים (למשל פיצ'ר 21 ו-פיצ'ר 61).
- ממוצע, ס"ת, מינימום ומקסימום לכל פיצ'ר, וכן לכל קומבינציה בין Class ו-Feature: ניתן לראות למשל כי התוחלת וסטיית התקן עבור Class 2 נמוכה יותר מ-Class 6 עבור מרבית הפיצ'רים.

ויזואליזציות שנוצרו במסגרת שלב זה מופיעות בנספחים.
כאמור, הוחלט לצמצם את ממדי סט הנתונים לסוגי מוצרים 2 ו-6 (Class 2, Class 6) בלבד, לאור האחוז הנרחב שהם מהווים מתוך כלל התצפיות.

בשלב Preprocessing, בוצעו השלבים הבאים –

1. Outlier removal: הסרת דגימות רחוקות שלרוב הינן מוטעות. חלק מהמסווגים מושפעים משמעותית מדגימות אלו.
חישבנו את מספר הדגימות שנמצאות במרחק גדול יותר מ-4 סטיות תקן מהממוצע, אך ראינו

כי דגימות רבות עונות על כך (2,582 דגימות), לכן החלטנו לא להסירן.
במידה ונתבונן בנתונים נראה כי רובו אפסים, לכן ניסינו לחשב ממוצע ושונות ללא אפסים, ולהסיר דגימות שנמצאות בטווח של 4 סטיות תקן מממוצע זה. נראה עדיין כי הרבה דגימות יורדות לנו מהנתונים, ולכן בחרנו לא להסיר נתונים.

2. בשלב זה בוצעה הקטנת מדד ה-skewness לכל פיצ'ר- פירוט על כך בשלב Feature Selection and Engineering (בוצע בשלב זה כי מדובר בשימוש ב-log והוא לא עובד על מספרים שליליים).

3. <u>Data Normalization</u>:

העברת כל המאפיינים לקנה מידה אחיד. הנתונים נורמלו תוך שימוש בממוצע ושונות:

$$x_{ik} = \frac{x_{ik} - \mu_k}{\sigma_k}$$

כאשר-

- x_{ik} הערך של פיצ'ר k עבור דגימה i
- μ_k התוחלת עבור ערכי הפיצ'ר k עבור כלל הדגימות
- σ_k ס"ת עבור ערכי הפיצ'ר k עבור כלל הדגימות

בשלב Feature Selection and Engineering, בוצעו השלבים הבאים -

1. <u>הורדת פיצ'רים ע"י שימוש במסווג Random Forest</u>: מודל זה מייצר אנסמבל של עצים, אוסף תתי עצים המוגבלים בגודלם כאשר הסיווג מתבצע ע"י הרצה על כל העצים ואיחוד תוצאות הסיווג (בדרך כלל על ידי סיווג לקבוצה עם מירב ההצבעות). שימוש במודל זה מאפשר לנו לקבל את החשיבות של כל פיצ'ר ע"י קריאה למאפיין של המסווג. ערכי החשיבות לכל פיצ'ר הוצגו כגרף נקודות, ובנוסף הוצגה החשיבות כערך מספרי עבור כל פיצ'ר. נראה היה כי פיצ'ר 12 הוא בעל חשיבות אפסית ולכן ניסינו להריץ את המודלים ללא פיצ'ר זה, כלומר בסה"כ נשארו 9 פיצ'רים המסבירים את סט הנתונים.

2. <u>שימוש ב-PCA לטובת טיוב וצמצום ממדי הבעיה</u>: PCA מסייע ביצירת קומבינציה לינארית של המאפיינים עבורה השונות מקסימלית. כך ניתן לייצג את הנתונים במספר פיצ'רים קטן/שווה למספר הפיצ'רים המקורי, כאשר כל הפיצ'רים המתקבלים הינם בלתי תלויים. בחרנו לייצג את הנתונים באמצעות פיצ'רים חדשים אשר יסבירו לפחות 95% ממנו. ייצוג זה הקטין את כמות הפיצ'רים המסבירים מ-9 שנותרו ל-7, אשר כאמור אינם תלויים זה בזה.

3. <u>עיצוב פיצ'רים קיימים</u>: **שלב זה בוצע טרם שלב Data Normalization** בוצע עיבוד נוסף על הנתונים, במטרה להקטין את מדד ה-skewness לכל פיצ'ר. בתהליך עיבוד זה עודכנו הנתונים לפי הנוסחה -

$$x_{ik} = \log(x_{ik} + \mu_k)$$

כאשר הסימונים זהים לסימונים עבור Data Normalization. נוסחה זו מקטינה את השונות בין הנתונים - ביצוע log מקטין את ערך הדגימה, כאשר הוספת התוחלת נועדה למנוע ביצוע log למספרים אי-חיוביים.
נבחנו מדדי ה-skewness לכל פיצ'ר טרום ביצוע העיבוד ולאחריו. ניתן לראות כי עיבוד זה אכן מקטין את ה-skewness ומכאן תורם להקטנת הפיזור של הנתונים.

בשלב הכנת סט הנתונים לסוג המשימה, בחנו מהו אחוז התצפיות בסט הנתונים מכל Class. לטובת ביצוע סיווג בינארי בשלבי הפרויקט המתקדמים, צומצמו ממדי סט הנתונים לשני סוגי המוצרים הנפוצים ביותר בסט - מוצרים 2 ו-6 (Class 2, Class 6) בלבד.
בנוסף, החלפנו את השמות "Class 2", "Class 6" בערכים 0 ו-1 בהתאמה, לטובת נוחות והתאמה למודלים שונים בסעיפים הבאים דוגמת Logistic Regression.

הרצת המודל לקלסיפיקציה בינארית והערכתו

בכל המודלים, הופרדו הנתונים המסווגים לשתי קבוצות - 80% מהדגימות המסווגות בלבד שימשו כמידע לטובת אימון המודל (train) ולאחר מכן המודל נבחן ע"י 20% הדגימות המסווגות שנותרו, לטובת ולידציה (validation).

לכל מודל, פעלנו תוך ביצוע השלבים הבאים:

- התאמת המודל כולל היפר פרמטרים מתאימים לפי נתוני ה-train.
- ניבוי נתוני הוולידציה תוך שימוש במודל המותאם.
- ביצוע ניבוי לנתוני ה-Test תוך שימוש במודל המותאם.

להלן הרחבה על כל אחד מהמודלים שנבחנו:

מודל Naïve Bayes

מודל הניבוי הבסיסי, לפי ההסתברות של דגימה להשתייך לכל קבוצה בהסתמך על הפרמטרים שלה. במודל זה הגדרנו priors=None מאחר ואין לנו מידע נוסף על הדגימות לטובת הניבוי.

מודל Logistic Regression

המודל מייצר רגרסיה לוגיסטית, משוואה לינארית לפיה משייכים את הדגימות למספרים 0/1, המייצגים סוגי מוצרים שונים. המודל שנבחר מבוסס על מרחק ריבועי.

מודל Decision Tree

עצי החלטה מחלקים את מרחב הדגימות לתתי-אזורים. הנחה היא שניתן לחלק לאזורים קטנים מספיק ואחידים מספיק כך שהתנהגות המערכת זהה בכל האזור. בכל צומת בעץ מתבצעת חלוקה ע"פ מאפיין מסוים ובעלים (כלומר, בקצה העץ) מתבצעת החלטה – סיווג במקרה של classification tree על מנת להביא למצב בו כל הדגימות בעלה מסוים שייכות לאותה קבוצה.
מכיוון שלעיתים קשה להגיע לכך בגדלים סבירים של עצים, נשאף לכך שהדגימות הומוגניות / אחידות ככל שניתן. מוגדר את מדד האחידות:

Gini impurity: $\sum_{i \neq j} P(w_i)P(w_j) = \frac{1}{2}\left[1 - \sum_j P^2(w_j)\right]$

כאשר $p(w_i)$ הוא אחוז הדגימות השייכות לקבוצה i. מספר הדגימות המינימאלי בכל צומת עומד על 2, לא הגבלנו את מספר הפיצ'רים או העלים בעץ על מנת לא להגביל את העץ וכי זמן ריצת המודל היה עדין קטן. הגבלנו את עומק העץ על ידי ניסוי וטעייה ל-10 רמות תוך כדי בדיקה של השפעת השינוי על מדדי הערכה בסוף הקוד.

מודל Neural Network

במודל זה אנו מייצרים רשת הלומדת את הבעיה, תוך שימוש במספר שכבות ברשת, מתן משקלות לכל חיבור בין Nodes ברשת ושימוש בפונקציית אקטיבציה. הרשת לומדת ומתאימה את עצמה ככל האפשר באמצעות נתוני ה-Train ולאחר מכן מסווגת את נתוני הווליידציה וה-Train תוך שימוש משקלות שנלמדו ע"י הרשת.

הרשת שואפת למזער את פונקציית ה-Loss:

$$(y-\hat{y})^2+\alpha\sum_{i=1}^{n}w^2$$

אודות התאמת היפר הפרמטרים- במודל זה נבחן פרמטר hidden_layer_sizes, הכולל הן את כמות רמות הביניים (שאינן שכבת קלט/פלט) ברשת, וכן את כמות ה-Nodes בכל שכבה. בחינת ההיפר פרמטרים בוצעה תוך שימוש ב-Grid-Search, כאשר נבחנו 4 קומבינציות של כמות שכבות וכמות Nodes שונה.
הפונקציה Grid-Search התאימה קבעה כי עבור הרשת יתקיים hidden_layer_sizes=(100,) כלומר מספיקה שכבת ביניים יחידה הכוללת 100 Nodes.

הערכת המודלים ומתן פרשנות למודלים הנבחנים

בשלב בניית Confusion Matrix, קיבלנו את המטריצה הרלוונטית לכל מודל תוך שימוש בחבילת sklearn, לכל אחד מהמודלים ולכל Kfold. בעזרת המטריצה חישבנו את הפרמטרים Precision ו-Recall לכל מודל.

פערי ביצועים בין המודל וה-Validation:

להלן ערך מדד AUC לכל מודל, עבור דגימות ה-Train ודגימות ה-Validation (ממוצע עבור 5 ערכי Kfold):

	Train	Validation
Naïve Bayes	0.72143	0.72137
Logistic Regression	0.71871	0.71874
Decision Tree	0.78477	0.78479
Neural Network	0.74755	0.74752

ראשית, לפי מדד זה, ניתן לקבוע כי מסווג ה-Decision Tree הוא המדויק ביותר לחיזוי הנתונים, ומכאן נעדיף להשתמש בו לטובת סיווג ערכי ה-Test.

בנוסף, ניתן לראות כי לכלל המודלים התקבל ערך דומה למדד AUC עבור ה-Train וה-Validation - ומכאן ניתן להסיק כי המודלים אינם סובלים מהתאמת יתר (overfitting) - אינם מותאמים לtrain באופן הדוק מדי שיפגע ביכולת הסיווג על דגימות חדשות.

בשלב בניית פלט ROC, ביצענו Kfold 5 פעמים עבור כל מודל, וחישבנו לכל מודל ולכל Kfold את הערכים : fpr, tpr. מתוך ערכים אלה, ייצרנו גרף עבור כל מודל הכולל את עקומת ROC לכל מודל הכוללת את כלל הנתונים לכל Kfold.

לאחר ביצוע כל הפעולות הנ"ל, כאמור נמצא כי מסווג ה-Decision Tree הוא המדויק ביותר לחיזוי. לפיכך, בוצע באמצעות מסווג זה ניבוי הסתברויות לכלל נתוני ה-Test. הנתונים יוצאו לקובץ CSV נפרד המצורף לפרויקט, כנדרש. הקובץ מכיל את וקטור ההסתברויות עבור 7538 התצפיות מ-test_no_target (מציג עבור כל תצפית את ההסתברות להיות מסווגת ל-Class 2, כאשר ההסתברות להיות מסווגת ל-Class 6 הינה ההסתברות המשלימה ל-1).

סיכום

בפרויקט זה ביצענו עבודת ניתוח סט נתונים, הפעלת מודלים מתקדמים לסיווג דגימות והצגת מדדי השוואה לבחינת כל אחד מהמודלים.

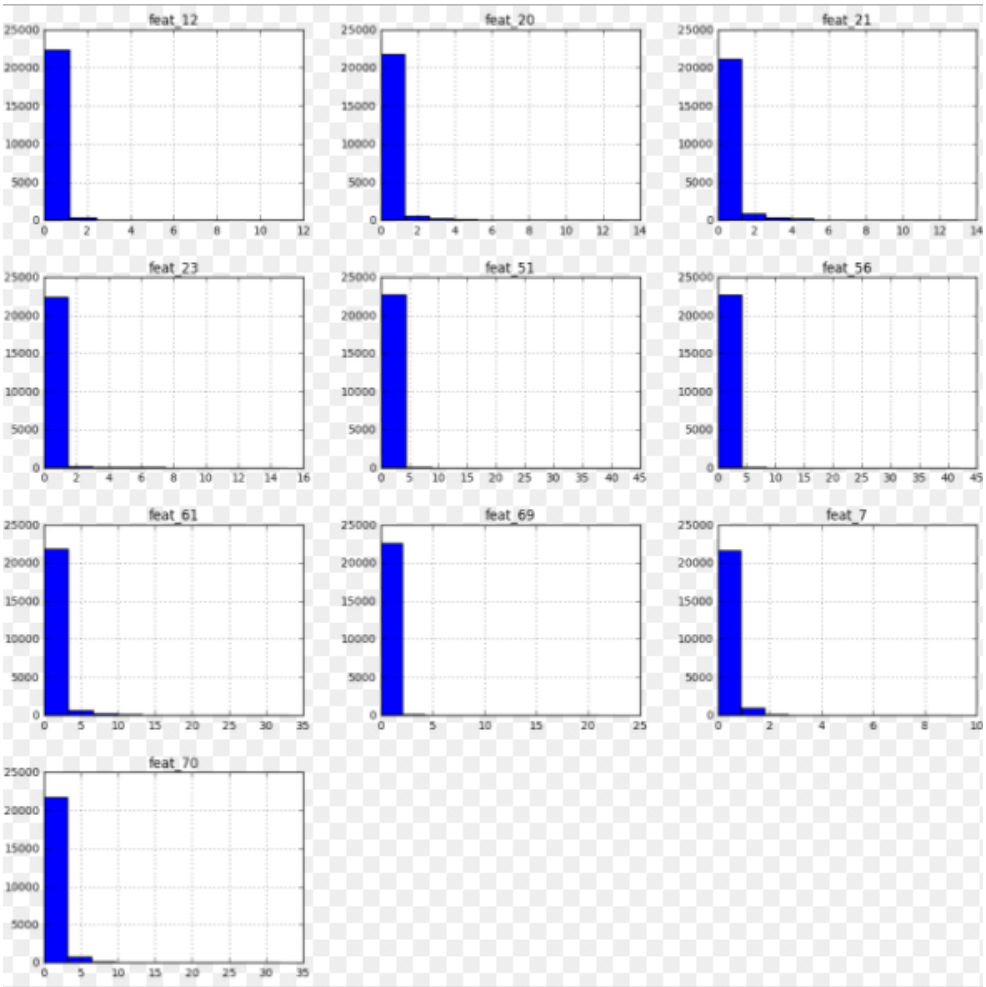
מתוך הפרויקט, בוצע ניבוי של נתוני Test תוך שימוש במודל הטוב ביותר שנמצא עבור סט הנתונים המסווג Train, לפי מדד AUC.

חשוב להדגיש, כי המודל שנבחר לסיווג בסט נתונים זה (Decision Tree), הוא אינו בהכרח המודל הטוב ביותר לכל סט נתונים, אלא יש לבצע את הניתוח הנ"ל במלואו על מנת להסיק מסקנות, לכל סט נתונים בנפרד.

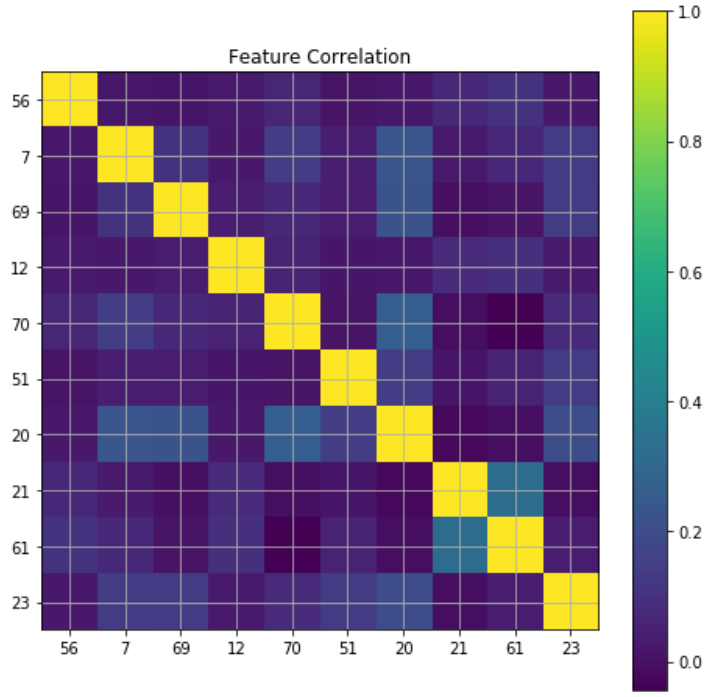
מטרה נוספת שהושגה במהלך ביצוע הפרויקט היא הטמעה של שיטות עבודה ומודלים אשר נלמדו במסגרת קורס "כריית ידע", לטובת ביצוע ניתוחים נוספים בעתיד, באקדמיה או בתעשייה.

נספחים

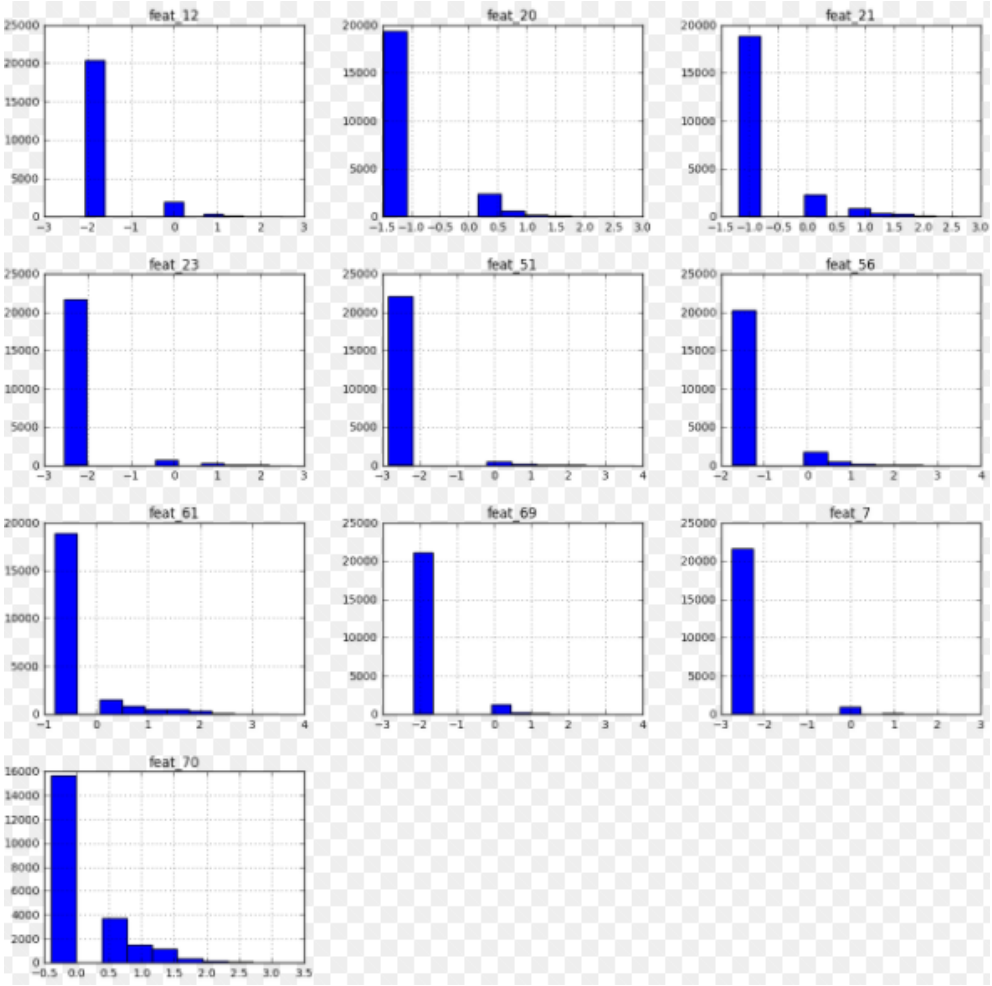
ויזואליזציה בפרויקט



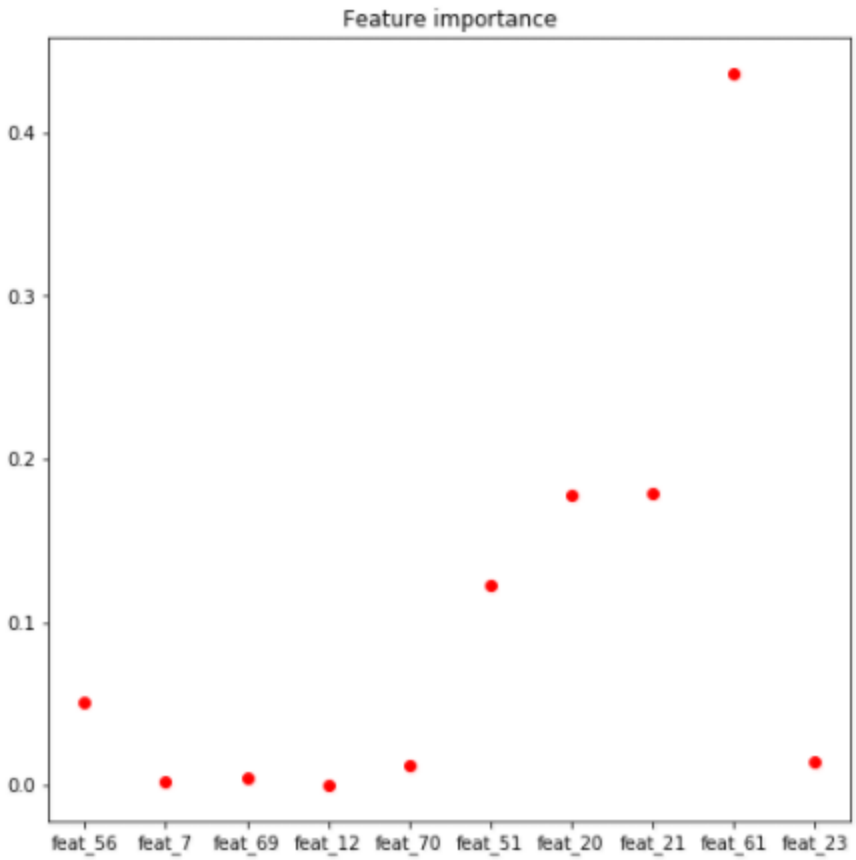
איור 1 - היסטוגרמה של הפיצ'רים המקוריים בסט הנתונים



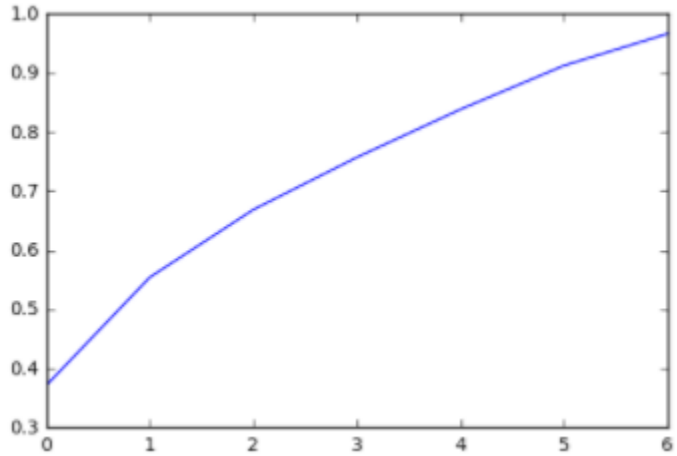
איור 2 - מטריצת קורלציה בין הפיצ'רים המקוריים בסט הנתונים



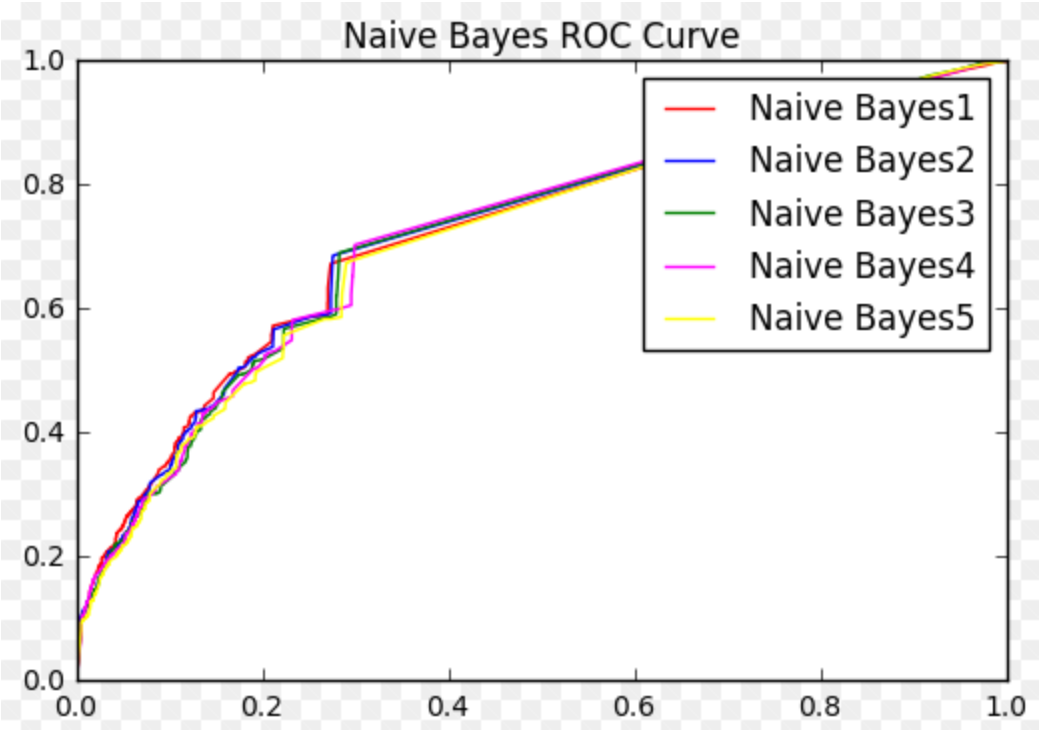
איור 3 - היסטוגרמה של הפיצ'רים לאחר הקטנת ה-skewness



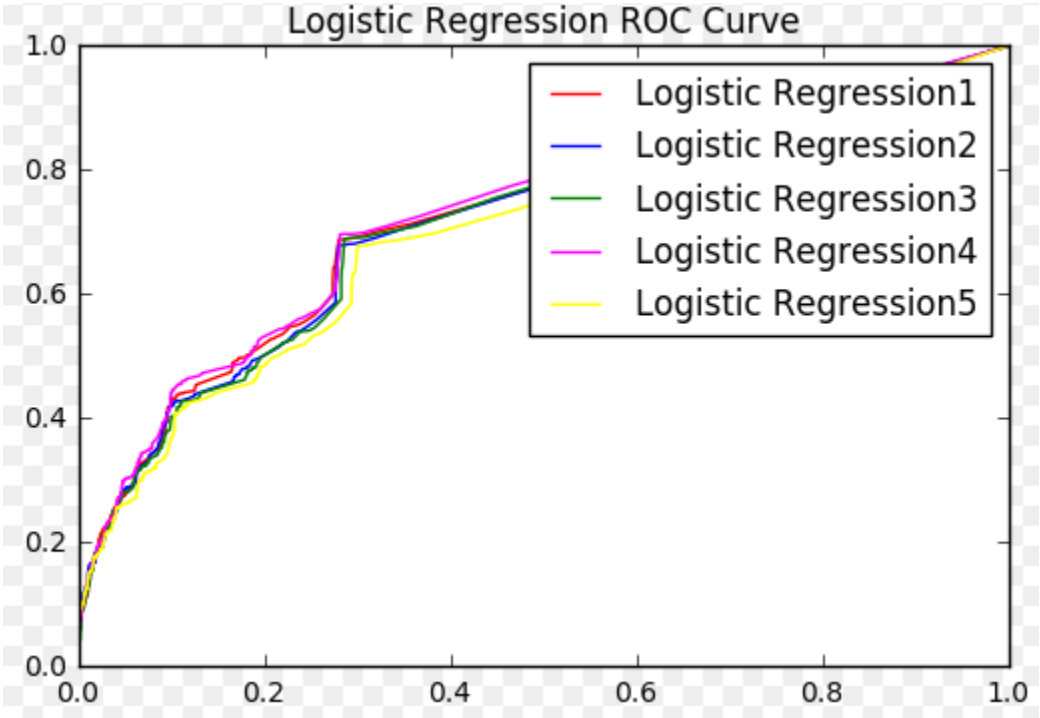
איור 4 - גרף חשיבות הפיצ'רים תוך שימוש ב-Random Forest



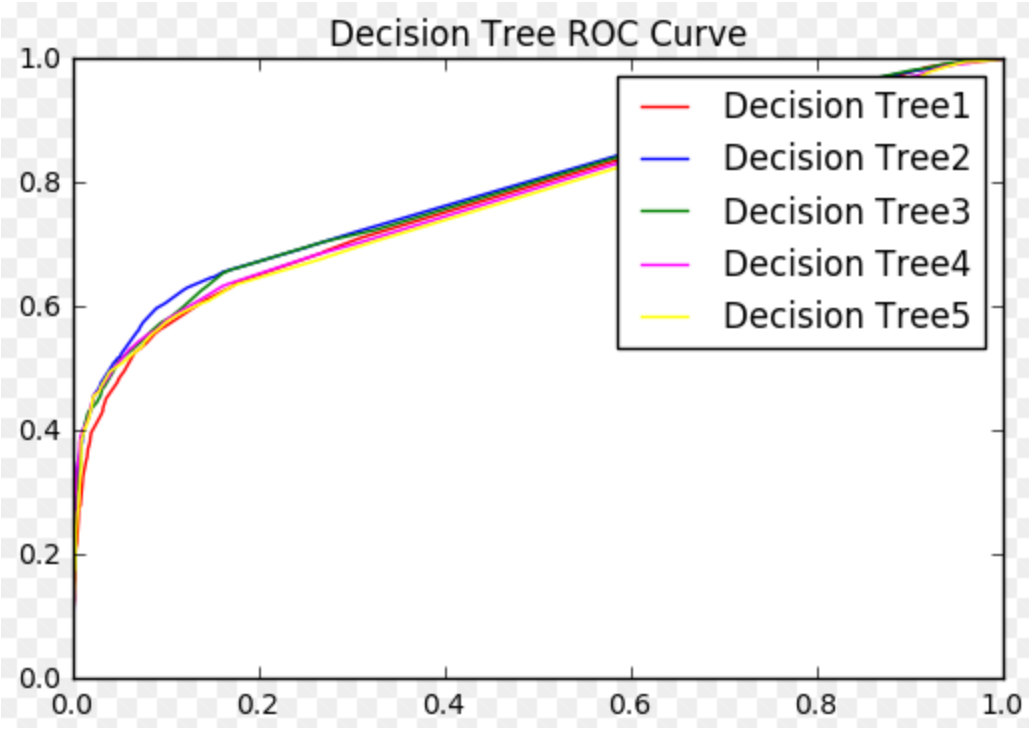
איור 5 - כמות השונות המוסברת מכל אחד מ-7 הפיצ'רים, לאחר ביצוע PCA



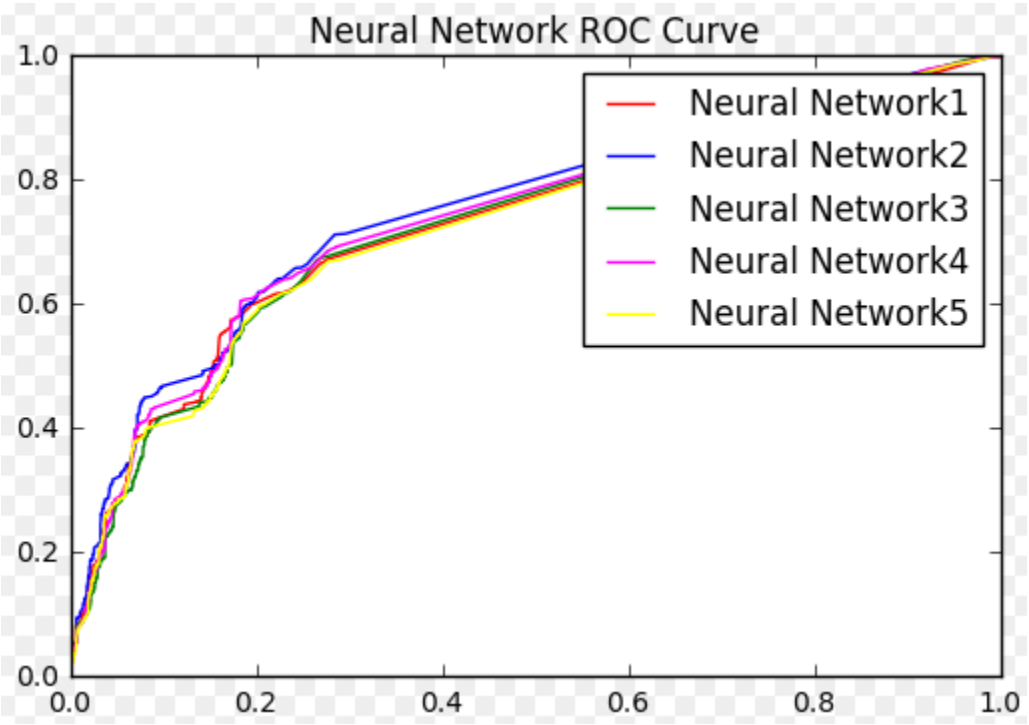
איור 6.1 – עקומת ROC למודל Naïve Bayes עבור Kfolds 5



איור 6.2 – עקומת ROC למודל Logistic Regression עבור Kfolds 5



איור 6.3 – עקומת ROC למודל Decision Tree עבור Kfolds 5



איור 6.4 – עקומת ROC למודל Neural Network עבור Kfolds 5