Iby and Aladar Fleischman
Faculty of Engineering
Tel Aviv University

הפקולטה להנדסה
ע"ש איבי ואלדר פליישמן
אוניברסיטת תל-אביב

פרויקט- מבוא ללמידת מכונה

26/7/2020

קבוצה 10

תקציר מנהלים

בפרויקט זה נתון לנו סט נתונים המכיל 25 פיצ'רים שונים ומספר רשומות גדול, כך שלכל רשומה יש סיווג לשתי קטגוריות: '0' או '1', כלומר נעסוק בבעיית סיווג בינארית. ראשית, ביצענו אקספלורציה עבור סט הנתונים שקיבלנו לצורך אימון המודלים. מטרת חלק זה הייתה להיחשף אל הנתונים דרך בחינתם על ידי טיפוסי הנתונים, סטטיסטיקה עליהם, ערכים חסרים, התפלגות הנתונים, קשרים קורלטיביים ביניהם וויזואליזציה רבה. תוצאות חלק זה סייעו לנו להבנת המשך תהליך העיבוד המקדים לצורך בניית המודלים. לכן בשלב השני ביצענו עיבוד מקדים על סט האימון שכלל: הסרת ערכים חריגים, נרמול נתונים, מילוי ערכים חסרים לנתונים, בניית פיצ'רים חדשים על ידי מניפולציות מתמטיות, הפיכת עמודות קטגוריאליות לנומריות והקטנת ממדיות על ידי הסרת עמודות עם מספר רב של ערכים חסרים או עם קורלטיביות ושימוש ב-PCA. לאחר מכן החלנו את העיבוד המקדים גם על סט הבחינה על ידי הפרמטרים שנבחרו בסט האימון מלבד הסרת החריגים שכן לא רצינו להסיר שורות שנרצה לחזות. לאחר מכן בשלב השלישי בחרנו 4 מודלים לבחינה: Logistic Regression, ANN, Adaptive Boosting, Gaussian naïve bayes ועבורם בחרנו את ההיפר פרמטרים הטובים ביותר לסט האימון על ידי שימוש ב-Gridsearch. בשלב הבא, הרביעי, ביצענו הערכה למודל על ידי מספר דרכים: בניית confusion matrix על Logistic Regression, חישוב מדד דיוק משוקלל נוסף, שימוש ב-K-Fold cross validation בתוספת תרשים ROC לכל K-Fold ולבסוף בדקנו האם המודל Overfitted על ידי פער ביצוע בין הדיוק של סט האימון לדיוק של סט הבחינה. לבסוף ביצענו חיזוי על הנתונים, שהוכנו לנו לצורך כך, תוך ציון ההסתברות לקבל תיוג '1' לכל רשומה.

חלק ראשון- אקספלורציה

חלק זה יכלול קריאה של הקבצים הנתונים עבור ה-train וה-test. לאחר מכן מטרת חלק זה הינו להיחשף לנתונים שנמצאים ב-data דרך הקובץ "train.csv" על ידי פעולות סטטיסטיות, ויזואליזציה, התפלגות הנתונים, קורלציות בין הפיצ'רים וכדומה. חלק זה יעזור לנו להבין כיצד ממשיכים בכל שאר התהליך הן מבחינת העיבוד המקדים והן מבחינת בחירת המודלים והחיזוי. ראשית, על מנת להתרשם מהנתונים, נציג כמה שורות מההתחלה ומהסוף, נציג את ממדי הנתונים ואינפורמציה על הנתונים הכוללת בעיקרה את סוג הערכים של כל פיצ'ר. לאחר מכן מצאנו לנכון להסתכל על כמה נתונים סטטיסטיים דרך המתודה describe() בנפרד עבור הנתונים הנומריים ובנפרד עבור הקטגוריאליים, שכן לכל אחד יש נתונים סטטיסטיים הנכונים עבורו. דבר נוסף שחשוב לדעת על הנתונים הוא כמות ה-null שישנם בכל עמודה על מנת לדעת האם עלינו לטפל בהם בשלב העיבוד המקדים והאם הם בכמות גדולה מידי שכן העמודה כבר אינה תורמת באופן מיטבי. בעיבוד המקדים יפורט אופן מילוי הערכים הריקים עבור העמודות הקטגוריאליות והנומריות. מסקנה עיקרית שהגענו אליה עד כה היא כי ישנם 2 סוגי עמודות עיקריות והן נומריות וקטגוריאליות וכי יש לטפל בהן בצורה נפרדת.
בחלק השני של החקירה החלטנו לחקור את הנתונים בדרך יותר ויזואלית וזאת דרך בניית פונקציות שישמשו אותנו גם בהמשך העיבוד המקדים. בנינו פונקציה שתייצר מפת חום (נספח 1) עבור העמודות הנומריות עם ציון של ערך הקורלציות מה שיעזור לנו לדעת היכן יש קשרים חזקים יותר ופחות. בנוסף בנינו פונקציה שלכל זוג תראה את הקשרים הקורלטיביים בצורה ויזואלית (נספח 2), כך שבאלכסון יש היסטוגרמה לכל עמודה. פונקציה נוספת תדפיס לנו היסטוגרמה לכל עמודה (נספח 3) כך שנראה איך כל פיצ'ר מתפלג. בנוסף תהיה פונקציה שתראה את הצפיפות בין כל גרף לעצמו ובין כל הגרפים (נספחים 4-5). עוד פונקציה תפצל את הנתונים לפיצ'רים ולעמודת התיוגים. בנוסף יש פונקציה שתקבל את עמודת התיוגים ותייצא תרשים עוגה (נספח 6) כדי לראות את אופן החלוקה בין התיוג של '0' לבין '1'. מכאן ראינו כי 76.4% תויג כ-'0' ו-23.6% תויג כ-'1'. ישנה פונקציה שתראה לנו האם ישנם שורות כפולות שהם זהות בנתונים. פונקציה נוספת תחלק את הפיצ'רים לעמודות נומריות וקטגוריאליות שכן נרצה לבצע פעולות נפרדות עליהן. בנוסף יש פונקציה שתציג גרף Box Plot (נספח 7) לעמודות הנומריות כך שנראה חריגות מהנתונים. ופונקציה אחרונה שתייצר גרף דומה להיסטוגרמה אך עבור העמודות הקטגוריאליות (נספח 8). כמה מסקנות נוספות שקיבלנו מהניתוח הם: הפיצ'רים נעים בטווחי ערכים שונים ועל כן עלינו לחשוב על נרמול הנתונים, אין שורות כפולות שיש להוריד, הרוב המוחלט של התיוגים הם '0' ולכן ייתכן והאימון על המודל יתבצע יותר טוב עבור תיוג זה (מכאן נשער כי מדד ה-specificity יהיה גבוה), עמודה '13' הינה בינארית, ישנן עמודות קטגוריאליות המכילות ערכי null רבים יחסית (מעל כאלף) וכי ישנן עמודות עם קורלציה גבוהה.

חלק שני- עיבוד מקדים

בחלק זה לקחנו את הנתונים מהקובץ המיועד לצורך האימון של המודל והכנו אותו לצורך בניית המודלים והחיזוי העתידי. נפרט אודות השלבים השונים שהחלנו על הנתונים. יש לציין כי בחרנו לקבע seed לפני העיבוד המקדים, שכן ישנן פעולות רנדומליות. מניתוח הנתונים ראינו כי עמודה '22' מכילה נתונים שנראים כמו שנים ועל כן שינינו את סוגה להיות אובייקט ולא מספר. ראינו כי עמודה 14 מכילה את הרצף 'mm' לאחר כל מספר מה שהעיד עבורנו כי מדובר ביחידות מידה של מילימטר וכי העמודה היא מספרית. על כן הורדנו עבורה את הסיומת והפכנו אותה לסוג int (פונקציית delete_end). בנוסף לכך הבחנו כי קיימות המחרוזות 'nanmm' ו-'unknown' בעמודות '14' ו-'13' בהתאמה ועל כן ראינו לנכון להפוך אותם ל-None כדי לא לשבש את הפונקציות בעיבוד המקדים של המודל.

<u>נתונים חריגים-</u> נתונים חריגים ברובם מוטעים ועלולים להשפיע בצורה משמעותית על המודלים ומכאן חשיבות הבדיקה שלהם. עבור כל פיצ'ר שנתון לנו ייתכנו נתונים חריגים משלהם ועל כן נרצה להתייחס אליהם בנפרד. בשלב ניתוח הנתונים, למרות שאיננו מכירים את משמעות הנתונים, ראינו את ההיסטוגרמות ואת ה-Box Plot וזיהינו שישנן תופעות שנראות כחריגות. לכן החלטנו לבצע תהליך הסרת חריגים על ידי Zscore. את תהליך זה ביצענו רק עבור העמודות הנומריות, שכן לעמודות מסוג אובייקט לא ניתן לחשב ערכים סטטיסטיים ועבור עמודות בינאריות ישנם 2 סוגי ערכים בלבד (0/1). ראשית "זרקנו" את השורות בהם יש null ובדקנו תצפיות שחורגות מ-4 סטיות תקן. לאחר מכן מצאנו את האינדקסים של השורות שמכילות חריגים ואותם "זרקנו" מה-data הכללי שכולל null. לבסוף בדקנו את אחוז החריגים שמצאנו כדי לוודא שלא הוצאו שורות רבות מידי מהנתונים וראינו כי קיבלנו כ-2.23% ועדיין כמות הנתונים שנשארת היא גדולה (21667 רשומות). נציין כי בחרנו ב-4 סטיות תקן כיוון שלפי ההיסטוגרמות הרבה עמודות נומריות לא מתפלגות בצורה נורמלית וייתכן שב-3 סטיות תקן נאבד מידע שלא נחשב כמידע חריג. בנוסף מדדי ה-AUC הסופיים יצאו טובים יותר עבור 4 סטיות תקן.

<u>נרמול הנתונים-</u> בשלב ניתוח הנתונים הבחנו כי כל פיצ'ר מתפלג בסקלה שונה, כלומר הנתונים שלנו אינם מנורמלים. לנרמול הנתונים יש חשיבות רבה ובגללה נרצה שכן לבצע נרמול. החשיבות נובעת מכמה סיבות: 1) אם נבחר במודל KNN שמשתמש במרחקים אוקלידיים, המודל ייתן עדיפות לפיצ'רים עם טווח גדול יותר. 2) אם נבחר במודל רגרסיה לוגיסטית האופטימיזציה תתבצע באופן איטי יותר- משקולות המקושרות לפיצ'רים עם טווח גדול יותר ילמדו מהר יותר. 3) אם נבחר להוריד ממדיות עם PCA הוא מחפש כיווני שונות מקסימליים וכאשר הנתונים לא מנורמלים הוא ייתן העדפה לפיצ'רים עם טווח גדול יותר. לשם כך השתמשנו בשיטת Z-score normalization שתתאים לכל פיצ'ר סקלה של התפלגות נורמלית סטנדרטית עם תוחלת 0 וסטיית תקן 1 וכל זאת עם שמירת צורת ההתפלגות המקורית (נספח 9 מציג היסטוגרמות לאחר שינוי). למדנו כי באופן כללי כלל אצבע הוא להשתמש בשיטה זו ולא ב-Min-Max (שמיועד בעיקר לעיבוד תמונות) ומעבר לכך לאחר בדיקת הדיוק עבור שתיהן ראינו כי השיטה שנבחרה טובה יותר. סיבה נוספת לבחירת שיטה זו היא כי ב-PCA נרצה פיצ'רים שימקסמו שונות. ביצענו את הנרמול עבור העמודות שהם נומריות בלבד וביצענו הפרדה עבור האפשרות שהנתונים הם על ה-test שכן הטיפול שם שונה בעיבוד המקדים עבורו.

<u>נתונים חסרים-</u> בשלב ניתוח הנתונים הבחנו כי ישנם פיצ'רים עבורם קיימים ערכים חסרים (null). עבור העמודות הנומריות בנינו פונקציה null_to_median(data,nums_cols) אשר מקבלת את ה-data ואת העמודות הנומריות. הפונקציה תעבור בכל עמודה קיימת ובמידה ומספר הנתונים החסרים בה גדול מאפס היא תחשב את החציון של אותה עמודה ותמלא בה את הערכים החסרים. לבסוף בדקנו שאכן כעת מספר הערכים החסרים לעמודה הוא אפס. החציון פחות תלוי בערכים קיצוניים ולכן העדפנו למלא בו מאשר בממוצע. עבור העמודות הקטגוריאליות בנינו פונקציה null_in_categorial(new_data,new_categorial) אשר מקבלת data ואת העמודות הקטגוריאליות. הפונקציה עוברת על כל עמודה ובודקת האם יש בה ערכים חסרים כלומר גדול מאפס, ואם כן מוציאה את כלל הערכים הקיימים בעמודה. לכל ערך חסר היא תבחר רנדומלית מהערכים הקיימים ותמלא בה. דרך זו נבחרה כיוון שראינו כי הגרפים של העמודות הקטגוריאליות שנשארנו לנו בשלב זה נראות די אחידות בגודלן. ובנוסף לאחר כמה בדיקות של מילוי ערכים בצורה רנדומלית ראינו כי ההתפלגות נשארת זהה באופן כזה שהעמודה שהייתה הכי גדולה עדיין הכי גדולה וכן הלאה. לאחר כלל הטיפול בערכים החסרים הדפסנו לצורך בדיקה את כמות הערכים החסרים לכל עמודה ווידאנו כי כעת היא 0. נציין כי את החציונים ואת הערכים בעמודות הקטגוריאליות שלחנו כפרמטרים לעיבוד המקדים של ה-test וזאת בהנחה שקובץ האימון מייצג את המציאות בצורה טובה יותר.

לאחר הטיפול בערכים החסרים עבור העמודות הנומריות וטרם הטיפול בערכים חסרים בעמודות הקטגוריאליות ביצענו כמה צעדים נוספים. ביצענו בדיקה של כמות הערכים החסרים מכלל הנתונים עבור עמודות קטגוריאליות. שכן אם עמודה קטגוריאלית הכילה המון מידע חסר ובלאו הכי זו עמודה שקשה להתמודד איתה בערכים חסרים אזי ראינו לנכון כי עמודה זו לא תורמת הרבה ואף יכולה להטות את התוצאות לכיוונים לא נכונים. על כך בנינו פונקציה check_nulls_categorial(data,cols,threshold) אשר מקבלת את ה-data את העמודות הקטגוריאליות וערך סף. הפונקציה תחשב את הפרופורציה בין כמות הערכים החסרים לבין כלל הנתונים ובמידה והפרופורציה גבוהה מערך הסף שנקבע אז העמודה מבחינתנו לא תתרום ואף עלולה להפריע ולכן "תיזרק" לבסוף. אנו החלטנו כי 4.5% זה ערך הסף שכן כבר הוצאנו כ-2.23% מה-data ולא נרצה להוסיף עוד ערכים שעלולים להטות את התוצאה.

לאחר מכן בדקנו את הערכים החסרים בעמודות הקטגוריאליות המעודכנות לאחר מחיקה. בנוסף, אנו יודעים כי המסווגים מניחים כי משתנים הם לא קטגוריאליים ועל כן עלינו להמיר אותם לנומריים. על כן השתמשנו בשיטת one-hot encoding. עבור עמודות שנראו כמו מספרים למשל עמודה 6 שהיה ניתן להוריד לה את התחילית 'a' היינו יכולים לקבל עמודה נומרית, אך בחרנו שלא כיוון שלמספרים יש סדר, חשיבות ומשמעות ואנו לא מכירים את ה-data כדי להחליט האם זה נכון. הפיכתם נעשתה על ידי פונקציית dummies(data,cols) אשר מקבלת data ועמודות קטגוריאליות והופכת אותן לנומריות (בינאריות). נציין כי ראינו בעמודות הקטגוריאליות ככאלה שעלולות לגרום לאימון טוב יותר של המודל בייחוד כאשר אנו לא יודעים מה המשמעות שלהם בסט נתונים זה, ועל כן לא הורדנו את כולן.

<u>פיצ'רים חדשים-</u> למדנו כי ניתן לבצע מניפולציה מתמטית על הפיצ'רים כך שננסה לבחון האם בצורה כזו נקבל אימון טוב יותר וכתוצאה מכך מודלים שיחזו טוב יותר. כיוון שאנו לא יודעים מה מכילה כל עמודה ב-data אין לנו דרך לחשוב אילו שילובים יתנו משמעות יותר טובה מאחרים. לשם כך החלטנו להסתכל על מפת החום שתראה לנו קורלציות ובפרט להסתכל על העמודות שהכי קורלטיביות עם עמודת התיוג הסופי. בחרנו ב-5 העמודות עם הקורלציה הגבוהה ביותר עם התיוג ובנוסף עוד 4 עם הקורלציה הגבוהה ביותר בכללי ועליהם החלטנו לעשות מניפולציות. יצרנו פונקציה (new_features(cols,data,cols_rand אשר מקבלת את העמודות הנומריות, את ה-data ומשתנה cols_rand שאליו ניתן להכניס עמודות שנבחרו מראש. תחילה ניסינו לבחור רנדומלית עמודות, אך הדבר השפיע על דיוק המודל בצורה תנודתית ועל כן בחרנו עמודות ספציפיות כנאמר לעיל. הפונקציה מפרידה את הנתונים הקטגוריאליים שהפכו לנומריים (dummies) מכל ה-data ותבצע את כל ההכפלות האפשריות בין כל זוג עמודות שיש עד דרגה 2 כלומר לכל היותר בריבוע וזאת על ידי המתודה PolynominalFeatures. בצורה כזו נקבל מספר לא קטן של מניפולציות מתמטיות ונוכל לבחון כמות פיצ'רים חדשה. לבסוף הפונקציה תחזיר את העמודות שנבחרו כדי לבצע מניפולציה זהה על סט ה-test.

<u>ממדיות הבעיה-</u> בעיה זו מכילה כמות פיצ'רים גדולה וזאת גם לאור בחירתנו להפיכת העמודות הקטגוריאליות לנומריות מה שיוצר מספר רב יותר של עמודות מהמקור. וזאת גם לאור כך שבחרנו ליצור פיצ'רים חדשים תוך מניפולציות מתמטיות על 9 עמודות כך שנוספות לנו עוד יותר עמודות. ראינו כי נוצרו לנו 110 עמודות לפני הורדת ממדים ועל כן עלינו כן להקטין. החשיבות של הקטנת הממדיות נובעת מכך שכאשר עושים ניתוח מידע בממד גבוה גם סיבוכיות החישוב עולה ואימון המודל ייקח יותר זמן וגם התובנות על הנתונים עלולות להיות פחות טובות. למעשה ייתכן וייווצר מצב בו יהיה רעש גדול ואמינות המודל תיפגע. מעבר לכך בכמות גדולה כזו של פיצ'רים עלינו להשתמש ביותר דגימות בצורה בה יהיו כל השילובים שנרצה על מנת לאמן את המודל בצורה המיטבית על כל האפשרויות. כתוצאה מכך כפי שאמרנו סיבוכיות המודל תעלה וישנו סיכוי יותר גדול להיכנס למצב של overfitting. על כן נרצה לנסות להוריד את ממדי הבעיה ועדיין להיות מסוגלים להסביר את הנתונים ולאמן מודל עם דיוק טוב.

<u>הקטנת הממדיות-</u> השתמשנו במספר שיטות להקטנת הממדיות לאורך העיבוד המקדים והן:

- <u>עמודות עם קורלציה גבוהה:</u> כבר בשלב הניתוח המקדים ראינו במפת החום פיצ'רים עם קורלציה גבוהה יותר (נספח 10), כלומר עמודות אלה מסבירות באופן דומה את אותם הנתונים. על כן אנו לא רואים צורך בהשארתם הן מבחינת זמן וכוח החישוב והן מבחינת הקלה על אימון המודל. לצורך כך הצגנו את הקורלציות עם ערכי True\False עבור קורלציות מעל 85% על גבי מטריצת הקורלציות לצורך המחשה ובנינו פונקציה הנקראת remove_cor(data,cols) אשר מקבלת את ה-data ואת העמודות הנומריות. הפונקציה בודקת לכל עמודה מי העמודות האחרות שנמצאות איתה בקורלציה של מעל 85% ושומרת את זוגות העמודות הקורלטיביות. לבסוף תוציא את העמודות הקורלטיביות מה-data כך שתשאיר כל פעם עמודה אחת מזוג עמודות קורלטיביות. כלומר תשמר את האינפורמציה רק מעמודה אחת. העמודות שמצאנו לנכון להוריד לפי ערך הסף של 85% הן 0,1,7,8,11,16.
- <u>מחיקת עמודות קטגוריאליות:</u> כפי שהוסבר קודם לכן ראינו לנכון להסיר עמודות קטגוריאליות עבורן מצאנו כי פרופורציית הערכים החסרים ביחס לכמות הנתונים עוברת ערך סף מסוים. הסיבות העיקריות היו שחסר בה מידע רב וזוהי עמודה שלא קל להתמודד עימה. העמודות שנמצאו הם 5,19. ומעבר לסיבות אלה, אם עמודות אלה היו נשארות אז בעת הפיכתם לנומריים היינו מקבלים מספר הרבה יותר רב של עמודות שעלולות לגרום לאימון לא נכון של המודל.

עבור 2 התהליכים לעיל נשלחו העמודות הרלוונטיות עבור העיבוד המקדים של סט ה-test.
נציין כי לפני החלת ה-PCA הפרדנו שוב את הנתונים שלנו לעמודות נומריות ולעמודות הקטגוריאליות שנהפכו לנומריות, וכל זאת כיוון שהבנו את חשיבות ההפרדה עבור עמודות נומריות ועמודות קטגוריאליות.

- <u>PCA:</u> ישנה חשיבות לבחירת מאפיינים אשר מכילים ערכים שמשתנים, כלומר עם שונות מוסברת מקסימלית. על כן אנו רואים בחשיבות של שיטה זו להקטנת הממדיות באופן כזה שייקח את כיווני השונות המקסימליים בממד גבוהה ויטילו לממד נמוך של פיצ'רים תוך שמירת מקסימום אינפורמציה. שיטה זו יעילה מאוד עקב כך שכאשר נרצה לשחזר וקטור מקורי לא נשחזר בדיוק והוא עלול לאבד מידע אך עם זאת מבטיח שנאבד כמה שפחות אינפורמציה ומכאן כי השגיאה הריבועית הכוללת תהיה המינימלית. לשם כך יצרנו את פונקציית (pca_cols(data,cols,threshold,pca אשר מקבלת data, עמודות נומריות/קטגוריאליות, ערך סף ומשתנה נוסף (אובייקט PCA) שישמש אותנו רק אם אנו בעיבוד המקדים של סט ה-test. ראשית הפונקציה תראה לנו בצורה ויזואלית את השונות המוסברות (נספחים 12-11) על ידי סך העמודות שנשלחו אליה ולאחר מכן תבחר בכמות הפיצ'רים שיתנו 95% שונות מוסברת מצטברת. ערך של 95% יסביר את הרוב המוחלט של השונות המוסברת ועל כן נחשב ככלל אצבע כערך טוב. לאחר החלת ה-PCA על העמודות הנומריות והקטגוריאליות בנפרד הגענו ל-82 עמודות בסך הכל שמסבירות 95% שונות מוסברת מצטברת. לפני כן היו לנו 110 עמודות והצלחנו להוריד ל-82 מה שמראה כי עדיין רוב האינפורמציה נשמרת ושהבעיה הקודמת אכן הייתה בעלת ממדיות גדולה.

החלת העיבוד המקדים על סט ה-test- עלינו לבצע עיבוד מקדים באופן זהה על סט זה כמו שנעשה על סט האימון כדי לבצע מניפולציה זהה על סט ה-test וזאת לצורך החיזוי העתידי עליו. בהינתן והורדנו עמודות מסוימות בשל קורלציה למשל מסט האימון עלינו להוריד את אותן העמודות בדיוק גם מסט הבחינה ועל כן מהעיבוד המקדים של סט האימון החזרנו פרמטרים רבים כדי להשתמש בהם. בצורה זו ביצענו את כלל השלבים באותו הסדר שנעשה בסט האימון גם על סט הבחינה אך מלבד שלב אחד של הסרת החריגים. בשלב זה אנו מורידים שורות מה-data הנתון ובסט הבחינה נרצה לחזות את כלל הרשומות בו מבלי להוריד ממנו שורות ומכאן החשיבות לא לבצע שלב זה.

חלק שלישי- הרצת המודלים

בחלק זה בחרנו 4 מודלים לבחינה: Gaussian naïve bayes, Logistic Regression, ANN ו-Adaptive Boosting .
לצורך בחינת הפרמטרים הטובים ביותר עבור כל מודל שנבחר החלטנו ליצור פונקציה
Gridesearch(classifier,parameterOptions,final_train_data,train_labels) היא מקבלת את המודל שנרצה לבחון, את הפרמטרים שנרצה לבחון, את סט האימון הסופי ואת התיוגים שלו. הפונקציה תשתמש במתודה של GridSearchCV שתמקסם לנו את מדד ה-AUC ותחזיר את הפרמטרים הטובים ביותר עבור המודל. פונקציה זו הופעלה על כל מודל והדפיסה את הפרמטרים הכי טובים ואת המדד. את הפרמטרים הנבחרים של המודל ואת תוצאותיהם ניתן לראות בחלק של הערכת המודלים על פי K-Fold cross validation. בנוסף עבור פירוט מעמיק על הפרמטרים ניתן לראות בנספח 17.

חלק רביעי- הערכת המודלים

לפני ההערכה החלטנו לבצע חלוקה של הנתונים על ידי המתודה train_test_split על הנתונים הסופיים לסט האימון ועל התיוגים בהתאמה באופן בו גודל סט הבחינה יהיה 0.3.

Confusion Matrix- בחרנו לבצע תהליך זה עבור מודל ה-Regression Logistic (נספח 13).
מהמטריצה קיבלנו 4 ערכים:

- TP- כלומר מצב שבו המודל מנבא כי החיזוי הוא 1 ואכן התיוג האמיתי גם כן 1, כלומר הסיווג התבצע באופן נכון. למעשה זהו מצב הנקרא hit. הערך שהתקבל הוא 850.
- FP- כלומר מצב שבו המודל מנבא כי החיזוי הוא 1 אך התיוג האמיתי הוא 0, כלומר הסיווג התבצע באופן שגוי. למעשה זה מצב הנקרא False alarm. הערך שהתקבל הוא 237.
- TN- כלומר מצב שבו המודל מנבא כי החיזוי הוא 0 ואכן התיוג האמיתי גם כן 0, כלומר הסיווג התבצע באופן נכון. למעשה זהו מצב הנקרא correct rejection. הערך שהתקבל הוא 4793.
- FN- כלומר מצב שבו המודל מנבא כי החיזוי הוא 0 אך התיוג האמיתי הוא 1, כלומר הסיווג התבצע באופן שגוי. למעשה זה מצב הנקרא Miss. הערך שהתקבל הוא 621.

K-Fold cross validation- השתמשנו בשיטה זו כיוון שגורמת למצב בו מאמנת את המודל על כמות נתונים גדולה ועם זאת משאירה כמות נתונים מספקת עבור בחינת המודל. ננצל את יתרון שיטה זו בכך שפחות חשוב איך ה-data מתחלק שכן כל נקודה תופיע פעם אחת בוולידציה. לפונקציה זו הכנסנו כל אחד מהמודלים שנבחרו כך שבחרנו K=5 לאחר ניסיון של K=5,10,15. הפונקציה הוציאה לבסוף מדד דיוק לכל מודל גם עבור סט האימון וגם עבור סט הוולידציה וגרף ROC לכלל המודלים. בחרנו ב-K=5 גם בשל שבכלל אצבע נבחר לרוב וגם כיוון שבמצב כזה סט הבחינה מספיק גדול כדי לבצע הערכה. המודלים ותוצאותיהם הם:

שם המודל	היפר פרמטרים	AUC ל-train	AUC ל-validation	פער ביצוע
Gaussian naïve bayes	{'priors': None, 'var_smoothing': 0.7}	0.8477	0.8464	0.0013
Logistic Regression	{'C': 0.3, 'class_weight': None, 'dual': False, 'fit_intercept': True, 'intercept_scaling': 1, 'l1_ratio': None, 'max_iter': 100, 'multi_class': 'auto', 'n_jobs': None, 'penalty': 'l1', 'random_state': None, 'solver': 'liblinear', 'tol': 0.0001, 'verbose': 0, 'warm_start': False}	0.8950	0.8909	0.0041
ANN	{'activation': 'logistic', 'hidden_layer_sizes': (100, 100), 'batch_size': 10, 'learning_rate_init': 1e-05, 'max_iter': 1250}	0.8940	0.8907	0.0033
Adaptive Boosting	{'n_estimators': 400, 'base_estimator': None, 'learning_rate': 0.5, 'algorithm': 'SAMME.R', 'random_state': None}	0.9265	0.8814	0.0451

חישוב מדד נוסף- עלות סיווג שגוי של דגימה בעלת תיוג אמיתי של 1 (שתויגה כ-0 על ידי המודל) כלומר FN חמורה פי 5 מעלות סיווג שגוי לתיוג אמיתי של 0 כלומר FP. על כן לדעתנו 'הענשת' המדד FN פי 5 ייתן מדד דיוק משוקלל בצורה נכונה כך ששאר הפרמטרים יהיו ביחס זהה. כלומר מדד הדיוק יחושב כעת על פי: $\frac{TP+TN}{TP+TN+FP+5\cdot FN}$. ביצענו בקוד בדיקות של מקרי קצה על מנת לראות כי המדד מתנהג כמצופה (נספח 15), ואכן ראינו כי הכל היה כמצופה. על כן חישוב זה נעשה עבור כל מודל שנבחר קודם לכן ועבור כל מודל הדפסנו את הדיוק המשוקלל. ניתן לראות את התוצאות בנספח 18.

Overfitting- נבדוק האם קיימים פערי ביצוע בין הרצת המודל על ה-train ועל ה-validation ובכך להסיק על overfitting. לשם כך מהרצת ה-K-Fold Cross Validation שמרנו לכל מודל את מדד הדיוק הממוצע שלו לכל סט (אימון,ווליד‌יציה) ובצורה ויזואלית הצגנו בגרף פיזור את 2 נקודות אלה לכל מודל (נספח 16). בנוסף לצורה הוויזואלית של הפער הדפסנו גם באופן כמותי את הפער שיש בין הדיוק של סט האימון לדיוק של סט הוולידציה. ציפינו כי הפער ביניהם יהיה חיובי שכן סט האימון תמיד מגדיל את הדיוק שלו וסט הוולידציה עלול לרדת בשלב מסוים בשל overfitting. נגדיר ערך סף של 0.03 שכן אם הפער יעבור ערך סף זה משמע המודל המוצג overfitted וכבר משנן את הנתונים. בבחירת ערך זה אנו נחמיר עם הכרת המודל כ-Overfitting כדי לנסות ולהיות מדויקים כמה שאפשר. מצב של overfitting הוא בעייתי שכן בהינתן דגימה חדשה שלא תתאם את סט האימון אז המודל לא יחזה בצורה טובה. קיבלנו כי ישנו מודל אחד שהוא Overfitted והוא Adaptive Boosting עם פער ביצוע של 0.0451. לאורך כלל התהליך השתמשנו בשיטות שונות שיכולות להוביל להכללת המודל ובנוסף לכך ישנן שיטות נוספות שיכולנו לפעול בהם:

- הקטנת ממדי הבעיה- פעולה זו עשויה לעזור כאשר משאירים פיצ'רים שמסבירים אחוז שונות מוסברת מצטברת גבוהה. בצורה כזו אנו ממזערים את הסיכוי שכמות גבוהה של פיצ'רים יכולה לפגוע במודל. בעיבוד המקדים שלנו ביצענו זאת על ידי הורדת עמודות קטגוריאליות הלוקות בחסר גדול, PCA ועמודות קורלטיביות. אנו חושבים שיכולנו לחשוב על דרכים נוספות להקטנת הממדיות ואולי ליצור דיוק יותר טוב.
- היפר פרמטרים מתאימים- ישנן אפשרויות רבות לבחירת היפר פרמטרים עבור מודל שנבחר ולבחירה יש השפעה על דיוק המודל. על כן השתמשנו במתודת GridSearch שתעזור לבחור את היפר הפרמטרים הכי טובים שניתן. ייתכן מצב בו היפר הפרמטרים שנבחרו הם אלה העלולים לגרום ל-overfitting ועל כן יכולנו לבדוק מצבים כאלה.
- עיבוד מקדים- את העמודות הקטגוריאליות הלוקות בחסר גדול בערכים ריקים החלטנו להוריד. ייתכן כי אם היינו בוחרים להשאירם המודל היה לומד על יותר data ומכאן כי ייתכן והיינו מכלילים את המודל טוב יותר. בנוסף במילוי הערכים החסרים לעמודות קטגוריאליות שכן נשארו, יכולנו לבדוק את הפרופורציות של כל קטגוריה בעמודה ולמלא בהתאם לפרופורציות.

חלק חמישי- ביצוע פרדיקציה

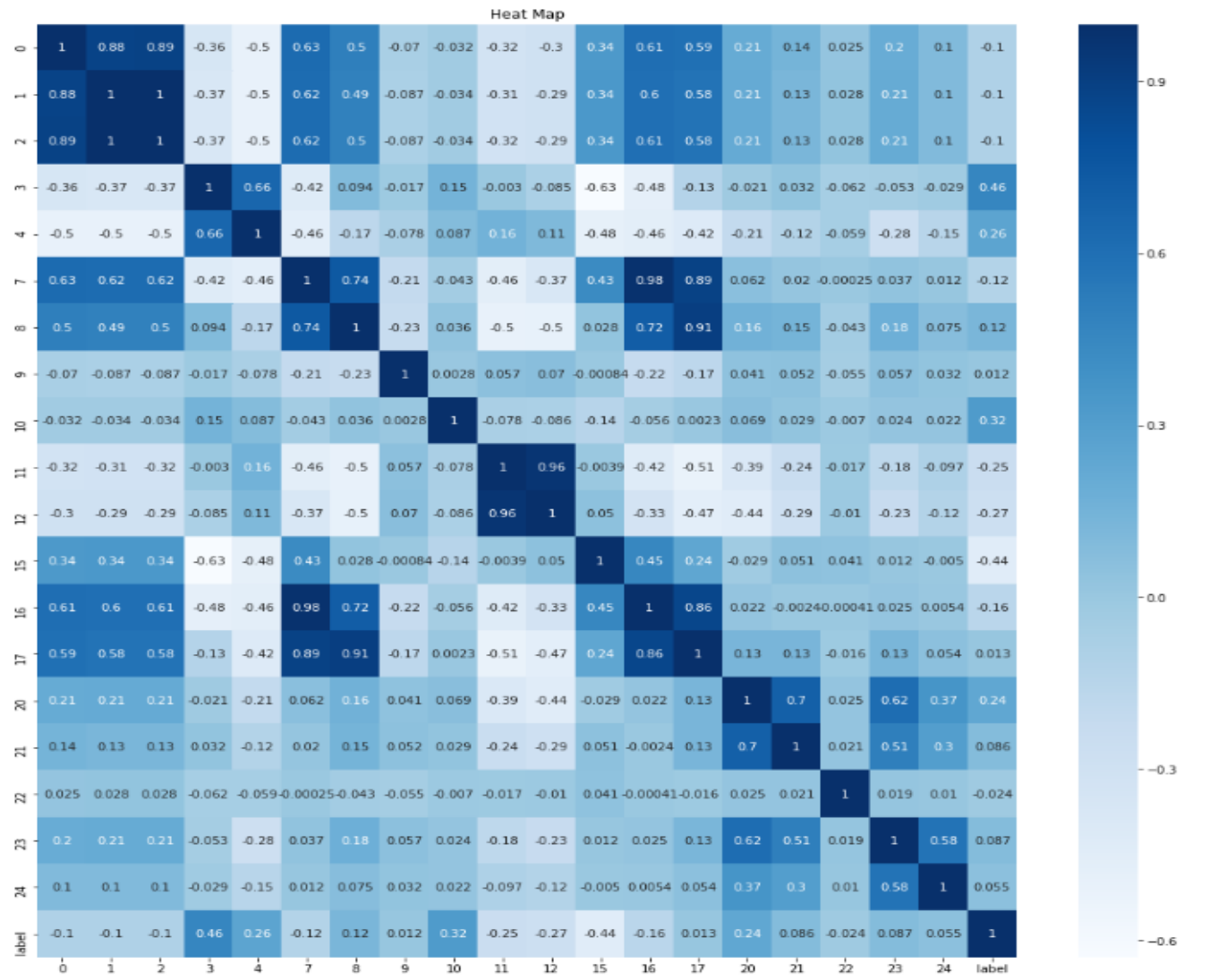
לקחנו את הקובץ של ה-test לאחר שבוצע עליו עיבוד מקדים מתאים והשתמשנו במתודה של predict_proba עבור המודל הטוב ביותר שנבחר לפי מדד הדיוק הטוב ביותר ושאינו Overfitted. אימנו את המודל ומשם הוצאנו את הסיכוי להיות מתויג כ-'1' וכתבנו זאת לקובץ הנתונים כנדרש.

סיכום

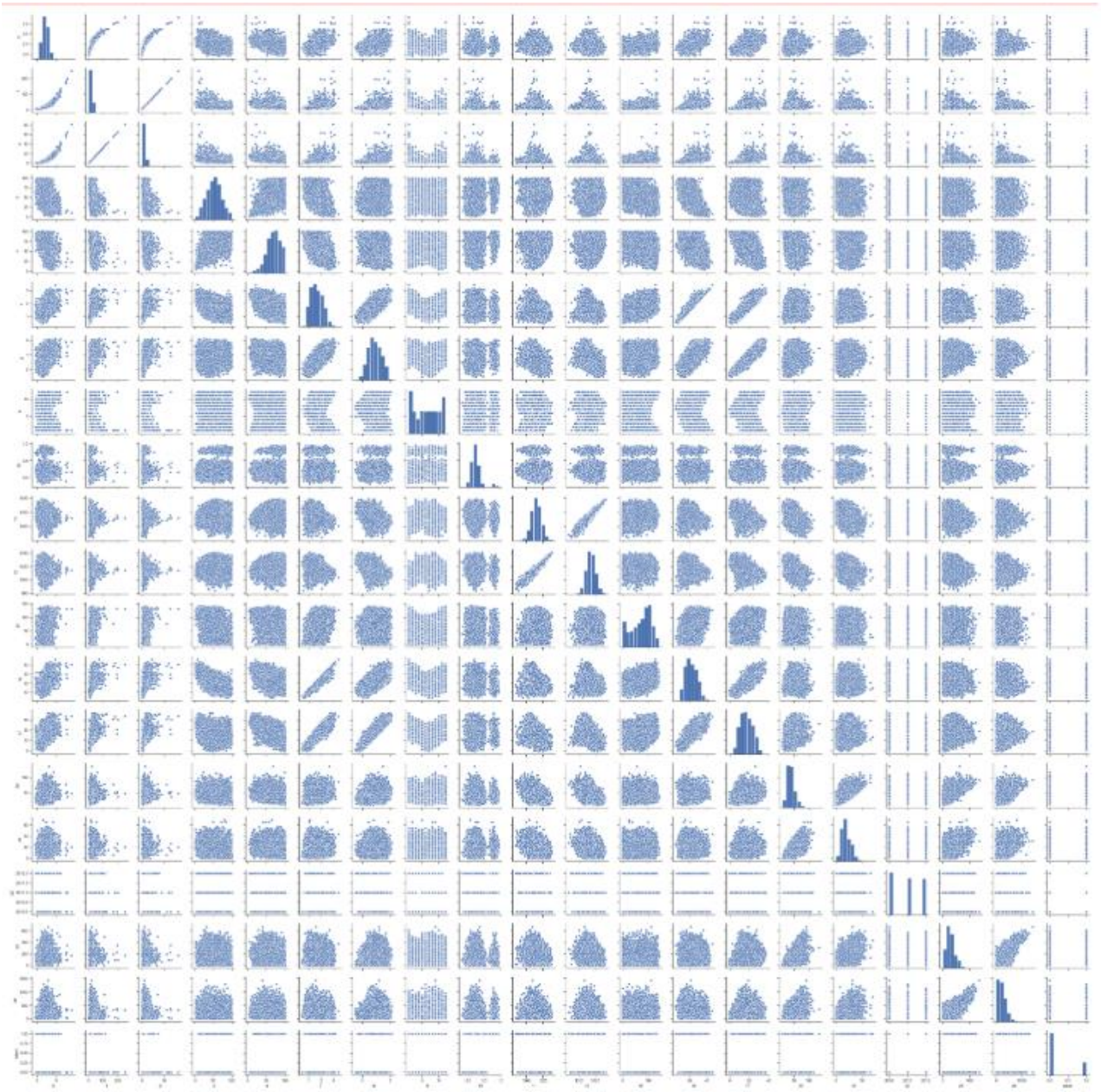
בפרויקט היה עלינו לבנות מודל classification לבעיית סיווג בינארית עם סט נתונים ועל סמך המודל לחזות סט נתונים אחר ללא תיוג. ראשית, ראינו כיצד שימוש מושכל בוויזואליזציה עוזר לנו ללמוד על הנתונים גם כאשר איננו יודעים את המשמעות לכל פיצ'ר. בנוסף למדנו כי יש לתת את הדעת לטיפול שונה עבור נתונים קטגוריאליים ונתונים נומריים שכן לכל אחד אופי שונה. משם הבנו כי לשלבים הרבים בעיבוד המקדים יש השפעה רבה על המודל בצורה בה ננסה למזער את ההטיות ככל הניתן לצורך חיזוי נכון ומדויק יותר. עם זאת יש לבצע הרבה ניסויים ולבדוק כיצד נכון לעבד את הנתונים לפי סט נתון, שכן יכול להיווצר מצב בו נעבד את הנתונים יתר על המידה בצורה בה נגרום למודל לשנן את הנתונים ולהיות overfitted. לשם כך ראינו כי יש תהליכים שנעשו ושיש לעשות כדי לנסות להכליל את המודל כמה שיותר. למדנו כי עלינו לבצע עיבוד מקדים באופן זהה על סט הבחינה שכן לא נרצה ליצור עיבוד שונה מזה שהמודל בנוי עליו. כמו כן למדנו כי בחירת היפר הפרמטרים למודל יכולים להשפיע ועל כן עלינו לבדוק מה יכול לגרום למודל להיות הטוב ביותר. דבר נוסף שלמדנו זה שחשוב לבצע הערכת מודל תוך ביצוע אימון על חלק גדול מהנתונים אך עם זאת לדעת להשאיר חלק לא קטן לצורך הערכה נכונה. בסוף ההערכות על 4 המודלים קיבלנו מכולם מדד AUC של כמעל 85%, כלומר לדעתנו המודלים יתנו תוצאות חיזוי די טובות. המודל הכי טוב שנבחר הינו ANN עם מדד AUC של 89.1% כפי שציפינו מ-4 האפשרויות הנבחרות על ידנו, שכן מודל זה יותר מורכב ונבחר עבורו היפר הפרמטרים הטובים ביותר שניתן לצורך אימון הרשת. יש לציין כי עבור העיבוד המקדים בדקנו אפשרויות שונות כמו ביצוע PCA על כלל העמודות, ביצוע נורמליזציה לפי min-max, מילוי ערכים חסרים בצורות שונות ועוד, אך הדיוק הטוב ביותר שמצאנו היה עבור העיבוד המקדים שהוצג. מכאן אנו מסיקים כי עיבוד מקדים זהו תהליך מעמיק ומתמשך שדורש תיקונים לצורך הבנה מה הדבר הנכון ביותר עבור סט נתונים נתון. דבר נוסף שעלה מריצות המודלים הוא כי מודלים מתקדמים יותר הם גם יותר מורכבים ולהם סיבוכיות חישוב גבוהה יותר ומכאן גם ארוכה יותר.

נספחים

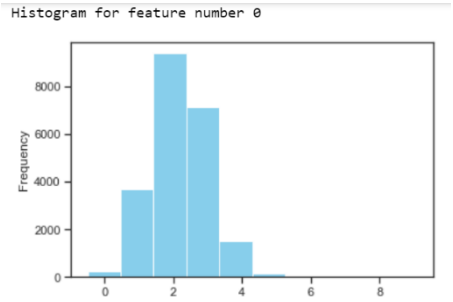
נספח 1- תרשים Heat Map עבור חלק האקספלורציה- עבור סט הנתונים המקורי.

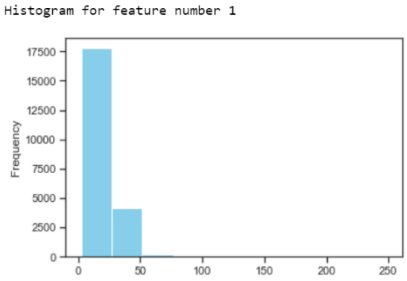


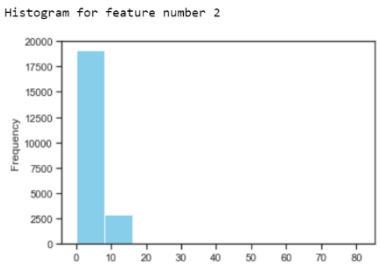
נספח 2- תרשים עבור הקשרים של כלל הזוגות על ידי תרשימי פיזור, כך שבאלכסון ישם היסטוגרמות לכל פיצ'ר.



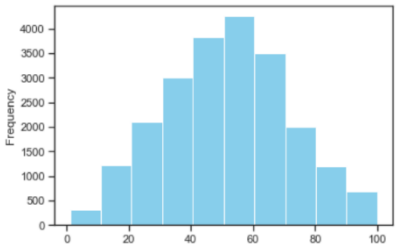
נספח 3- תרשימי היסטוגרמות לכל פיצ'ר נומרי בנפרד.



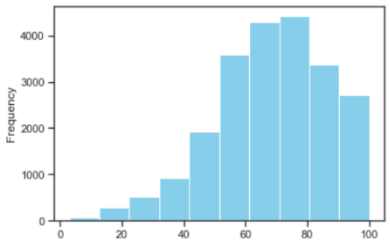




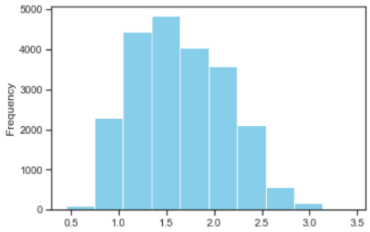
Histogram for feature number 3



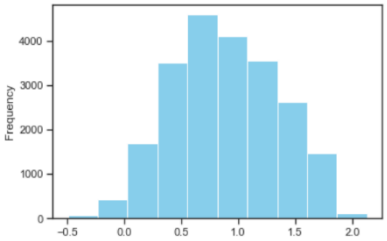
Histogram for feature number 4



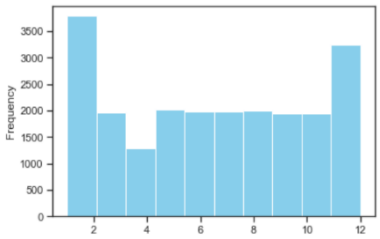
Histogram for feature number 7



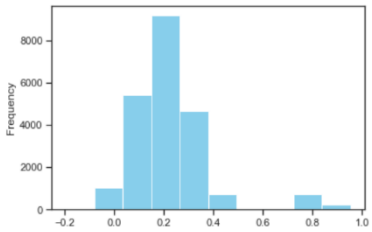
Histogram for feature number 8



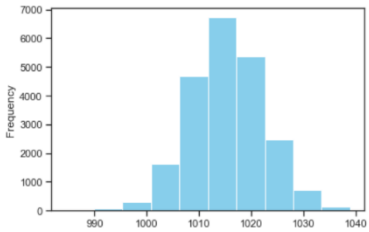
Histogram for feature number 9



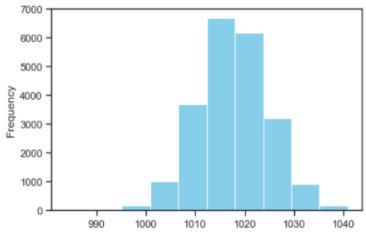
Histogram for feature number 10



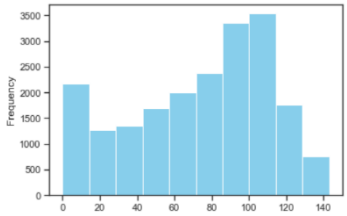
Histogram for feature number 11



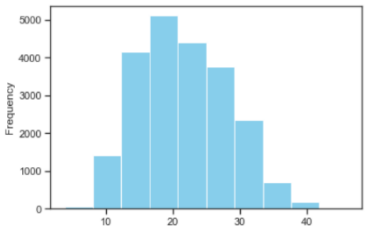
Histogram for feature number 12



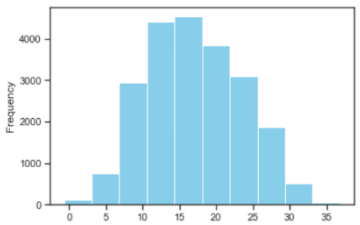
Histogram for feature number 15



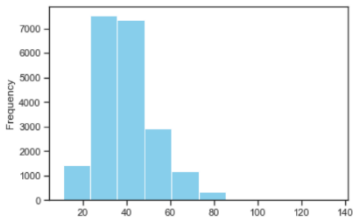
Histogram for feature number 16



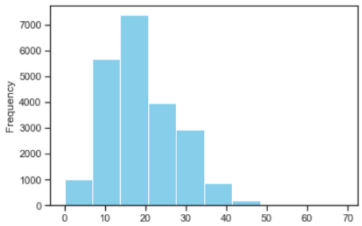
Histogram for feature number 17



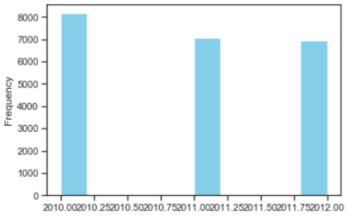
Histogram for feature number 20



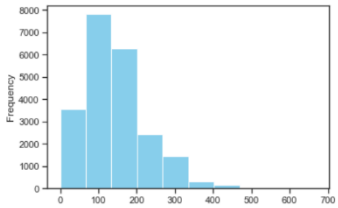
Histogram for feature number 21



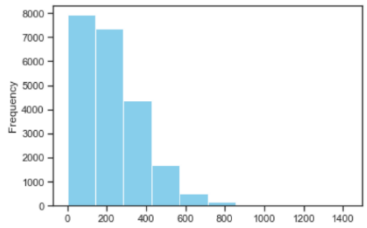
Histogram for feature number 22



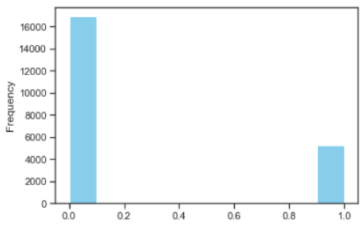
Histogram for feature number 23



Histogram for feature number 24

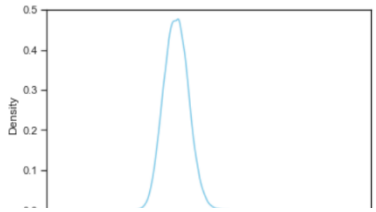


Histogram for feature number label

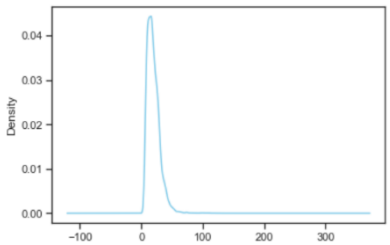


נספח 4- הצגת גרף צפיפות עבור כל פיצ'ר נומרי כדי ללמוד עוד על אופן ההתפלגות שלו.

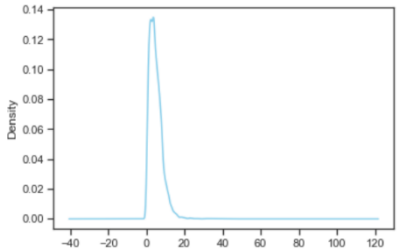
Density plot for feature number 0



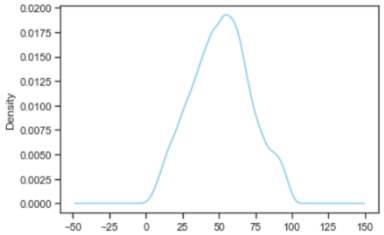
Density plot for feature number 1



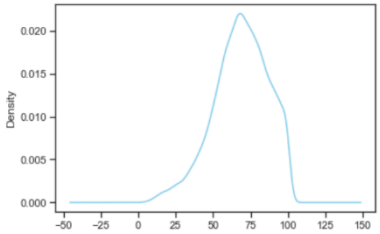
Density plot for feature number 2



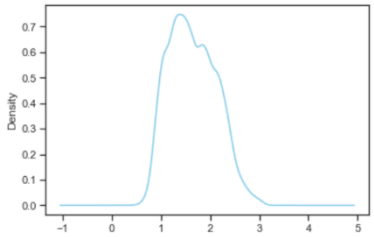
Density plot for feature number 3



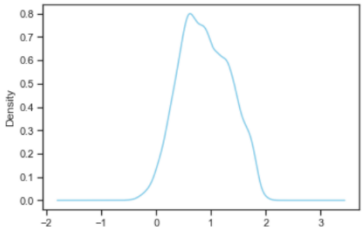
Density plot for feature number 4



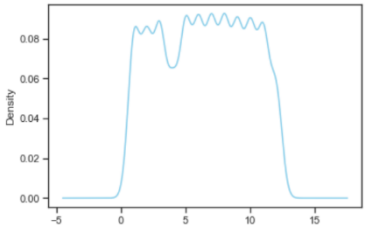
Density plot for feature number 7



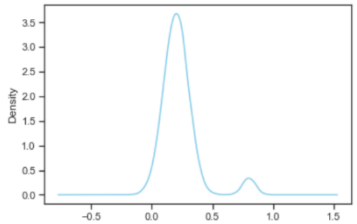
Density plot for feature number 8



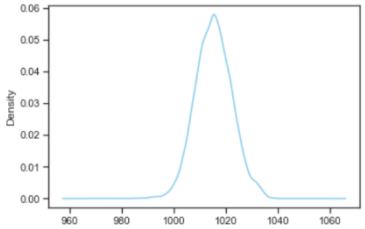
Density plot for feature number 9



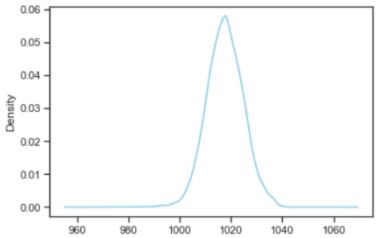
Density plot for feature number 10



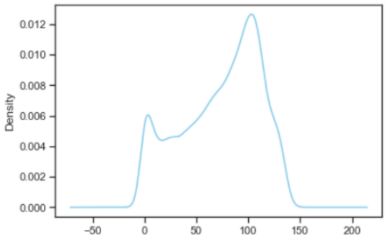
Density plot for feature number 11



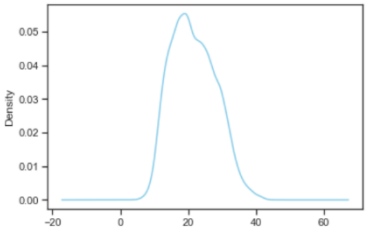
Density plot for feature number 12



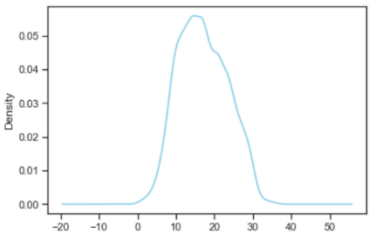
Density plot for feature number 15



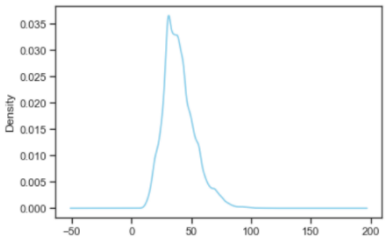
Density plot for feature number 16



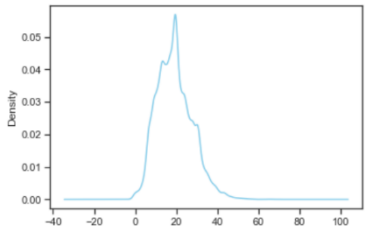
Density plot for feature number 17



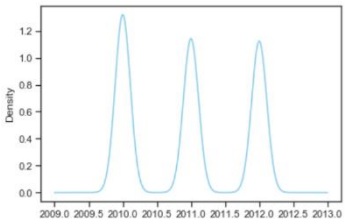
Density plot for feature number 20



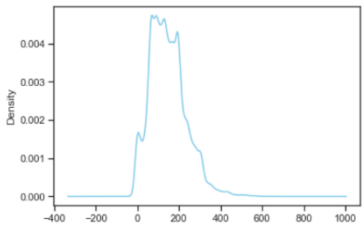
Density plot for feature number 21

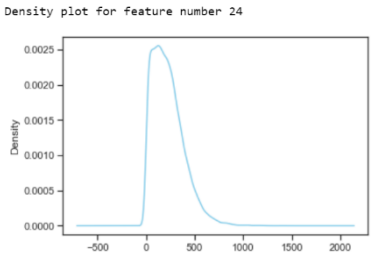


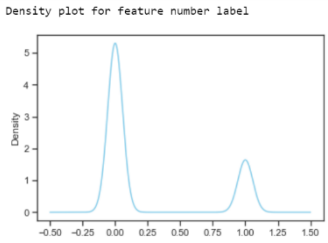
Density plot for feature number 22



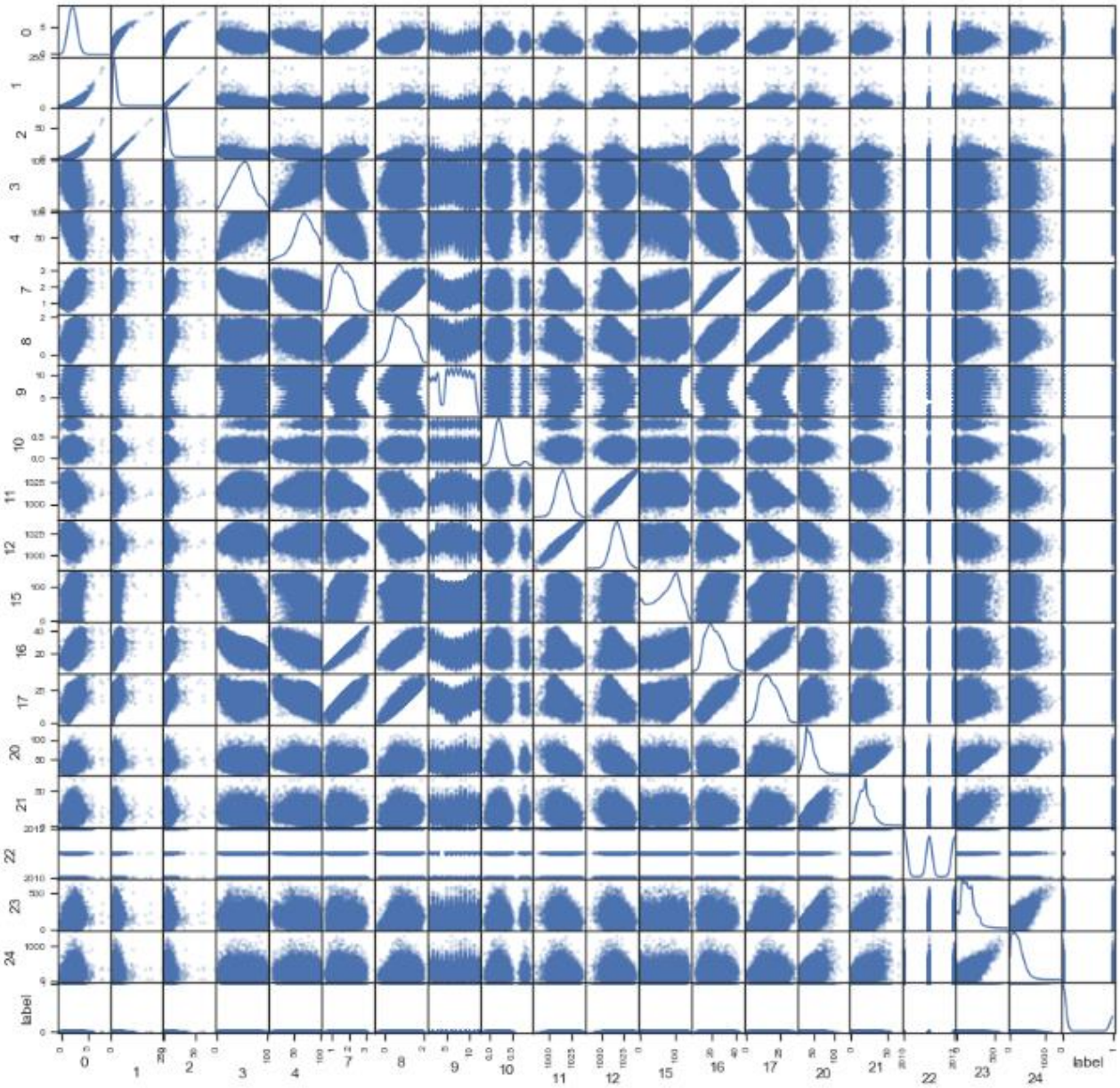
Density plot for feature number 23



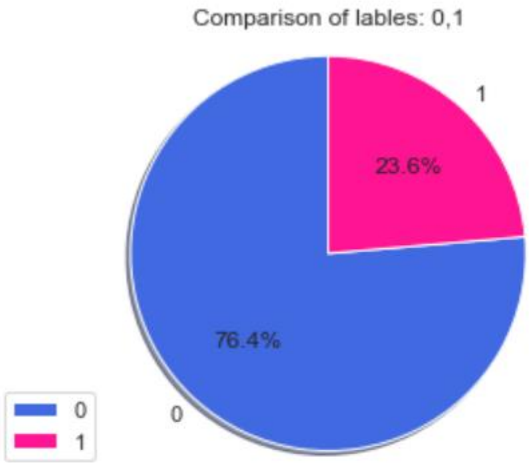




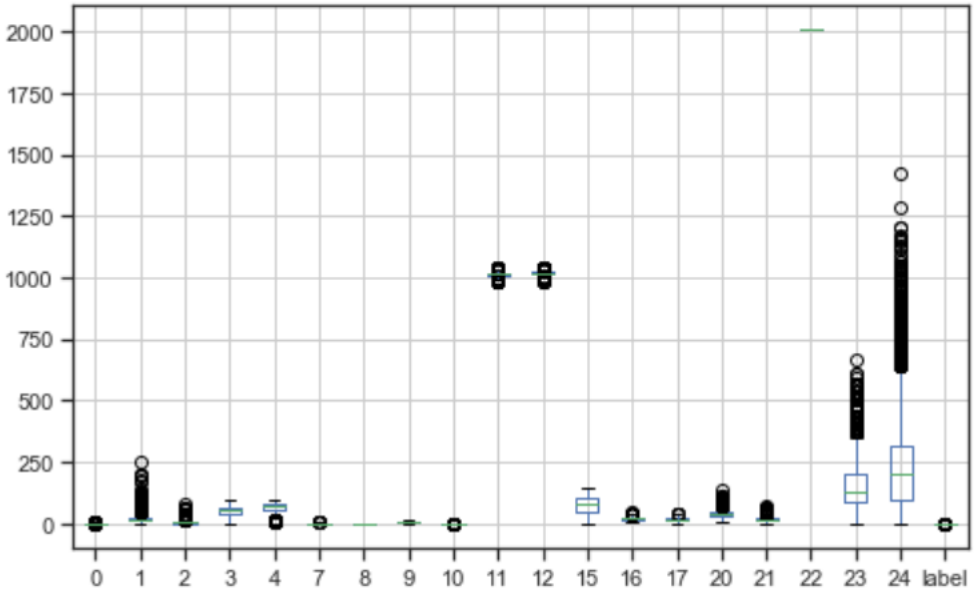
<u>נספח 5-</u> הצגת גרפי צפיפות בין כל זוג פיצ'רים נומריים.



נספח 6- גרף המשווה בין פרופורציות התיוגים '0' ו-'1'.

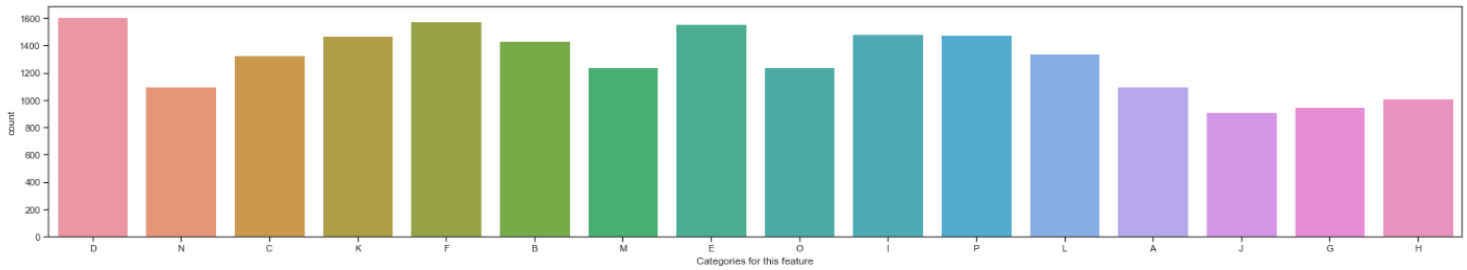


נספח 7- תרשים המכיל גרף Box Plot עבור כל פיצ'ר נומרי.

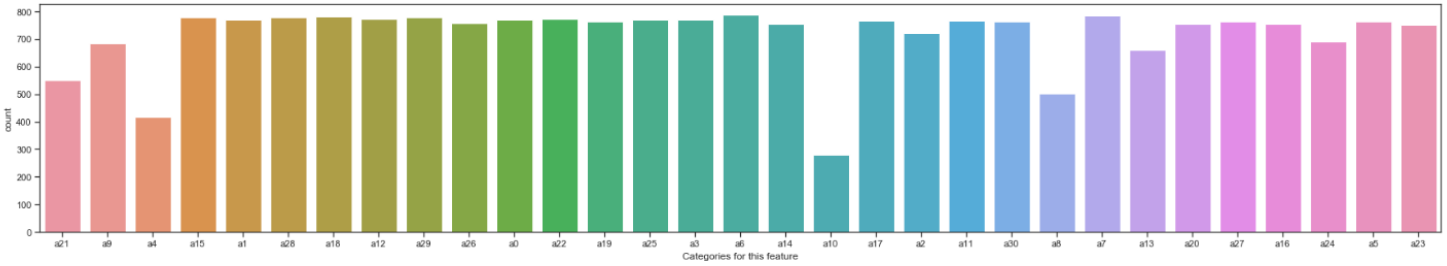


נספח 8- תרשימי היסטוגרמה עבור כל עמודה קטגוריאלית.

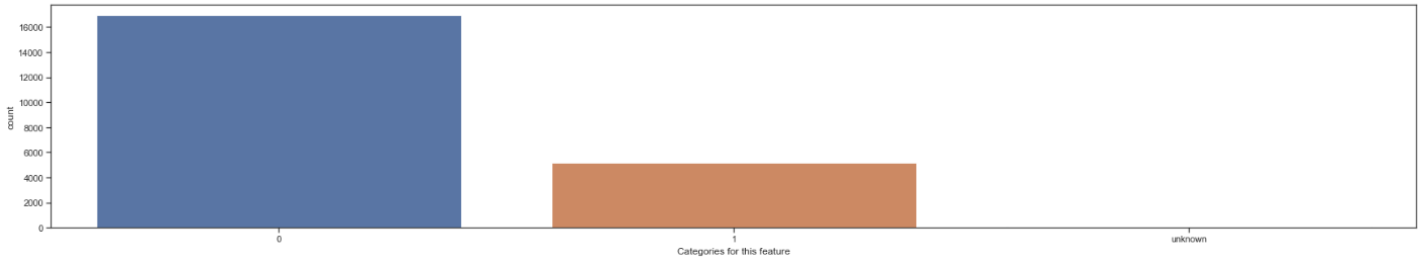
```
This is histogram for categorial feature number: 5
```



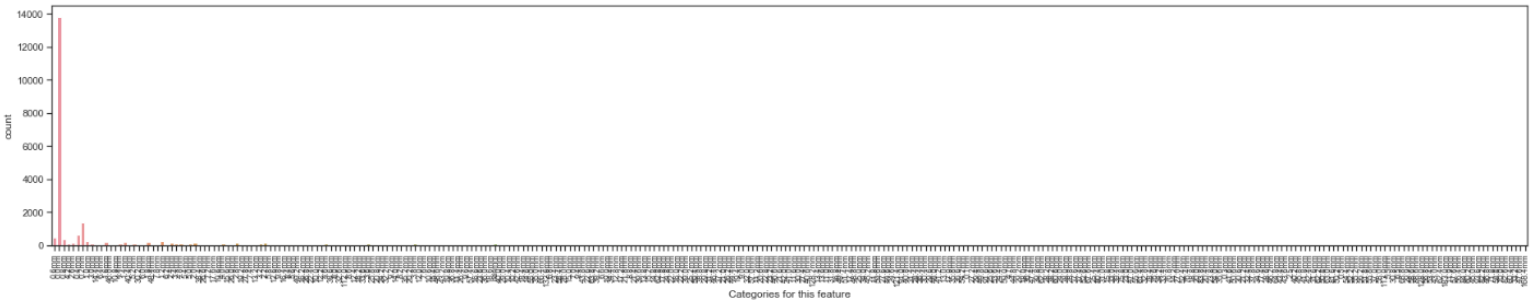
This is histogram for categorial feature number: 6



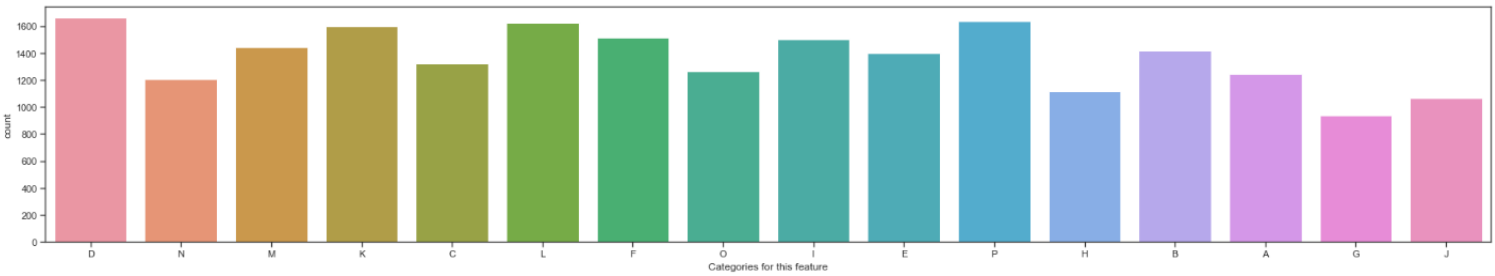
This is histogram for categorial feature number: 13



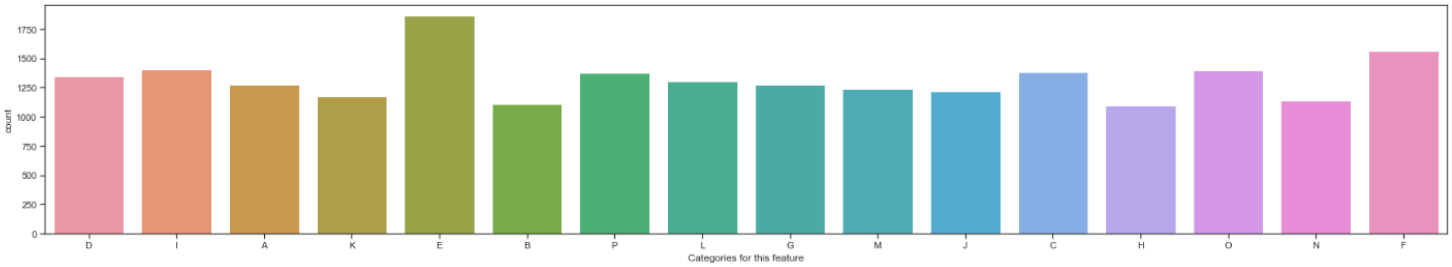
This is histogram for categorial feature number: 14



This is histogram for categorial feature number: 18

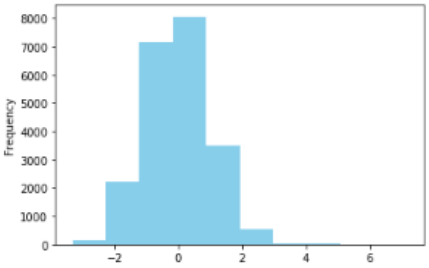


This is histogram for categorial feature number: 19

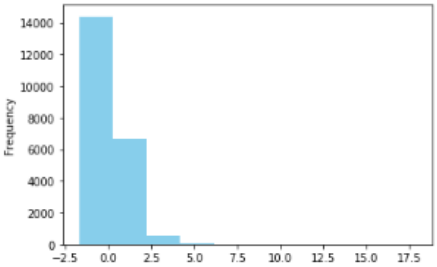


נספח 9- תרשימי היסטוגרמות לכל פיצ'ר נומרי בנפרד לאחר שביצענו נרמול נתונים על ידי Z-Score. כלומר נצפה לראות סקאלות שונות מנספח 3 כך שהתוחלת תהיה 0 וסטיית התקן תהיה 1 (נורמלית סטנדרטית).

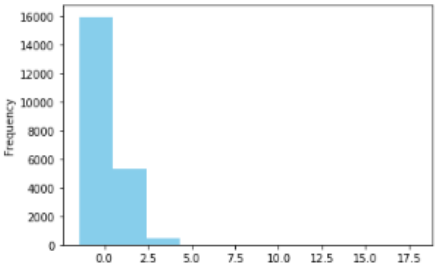
Histogram for feature number 0



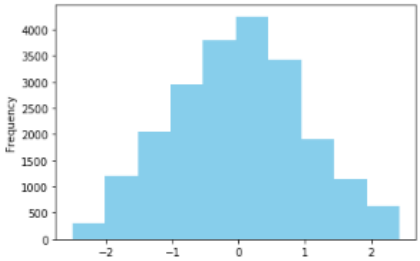
Histogram for feature number 1



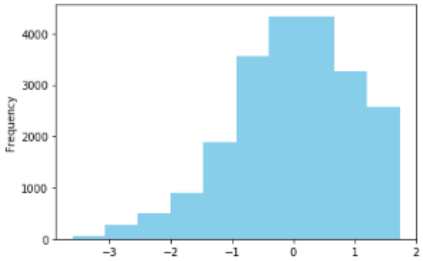
Histogram for feature number 2



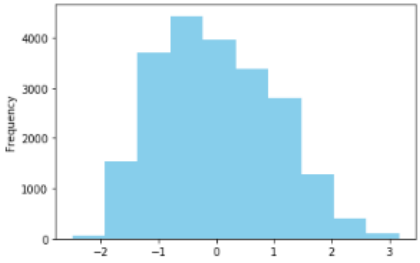
Histogram for feature number 3



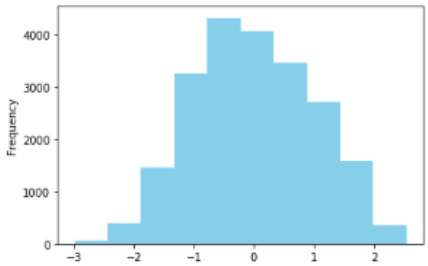
Histogram for feature number 4



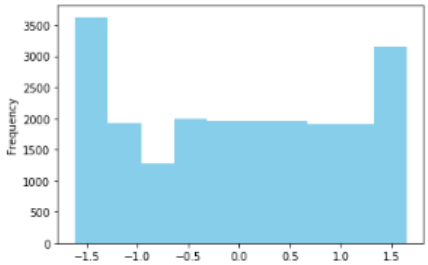
Histogram for feature number 7



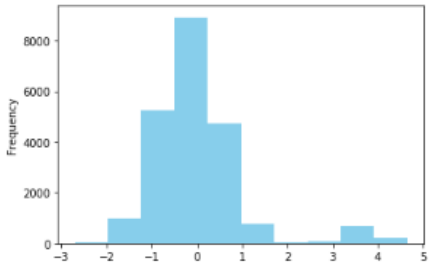
Histogram for feature number 8



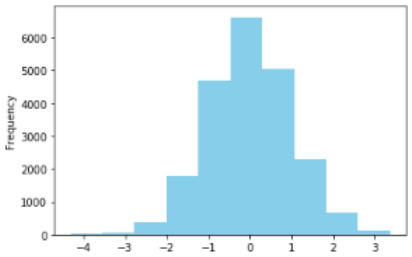
Histogram for feature number 9



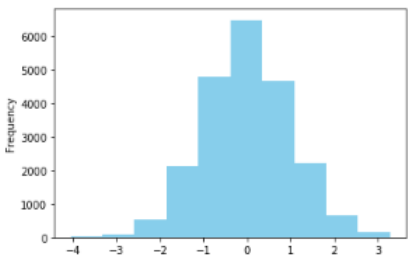
Histogram for feature number 10



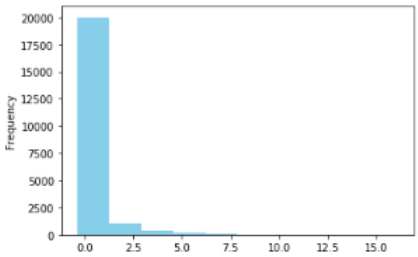
Histogram for feature number 11



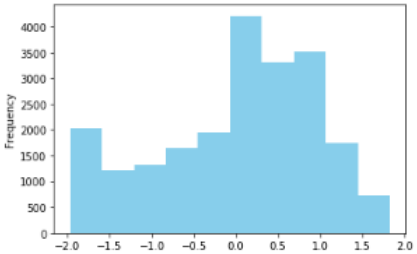
Histogram for feature number 12



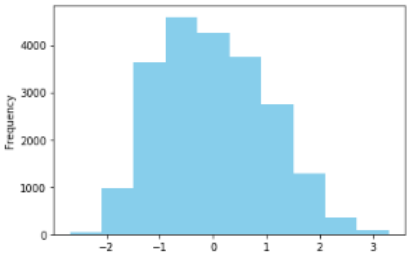
Histogram for feature number 14



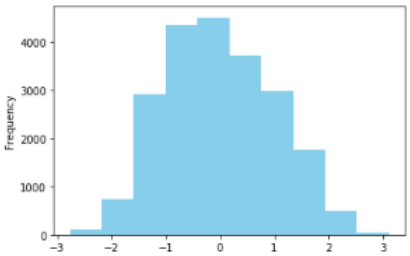
Histogram for feature number 15



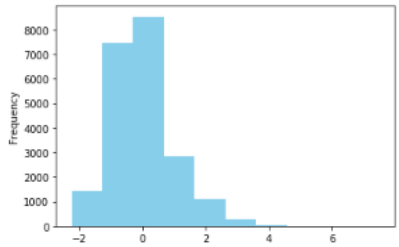
Histogram for feature number 16



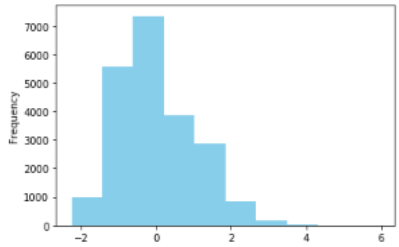
Histogram for feature number 17



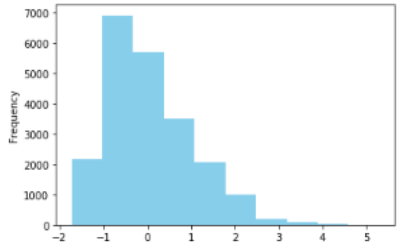
Histogram for feature number 20



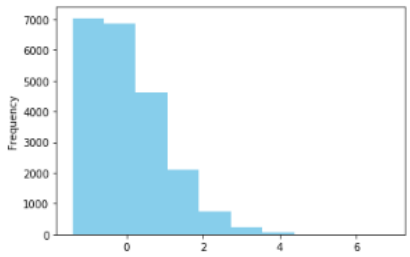
Histogram for feature number 21



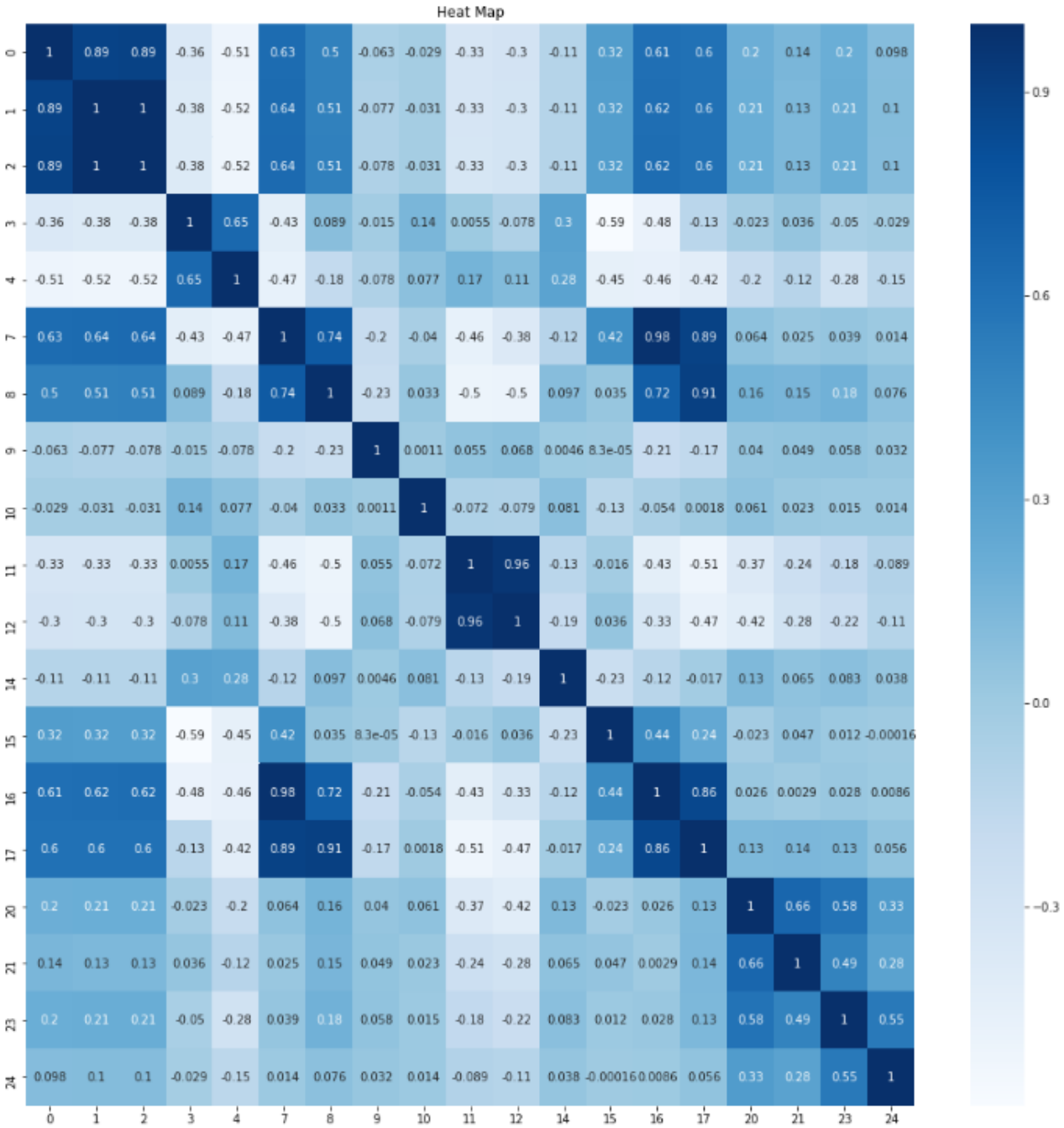
Histogram for feature number 23



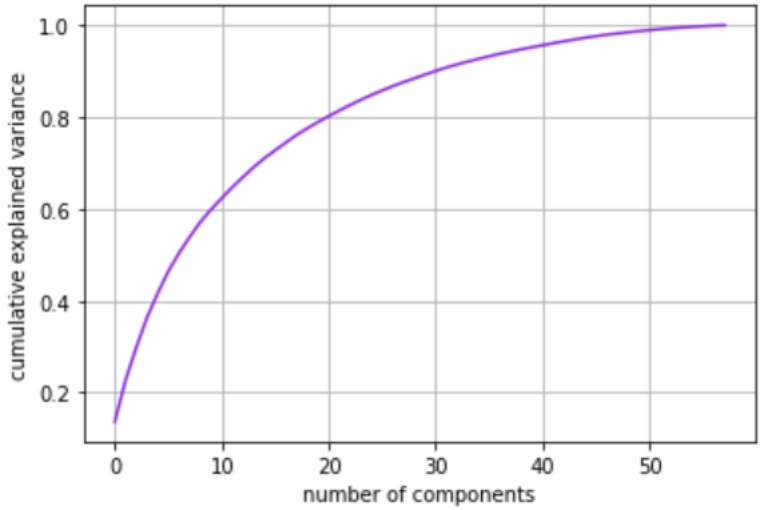
Histogram for feature number 24



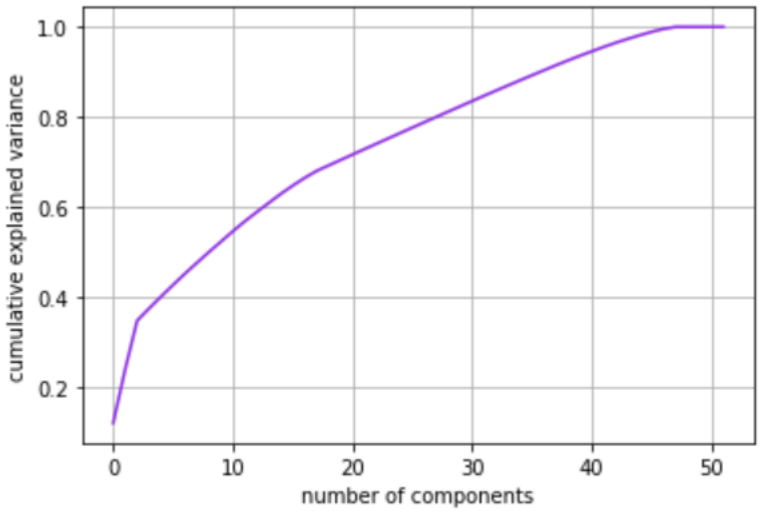
<u>נספח 10-</u> תרשים Heat Map בחלק העיבוד המקדים לאחר הסרת חריגים, נרמול נתונים ומילוי ערכי Null בעמודות הנומריות.



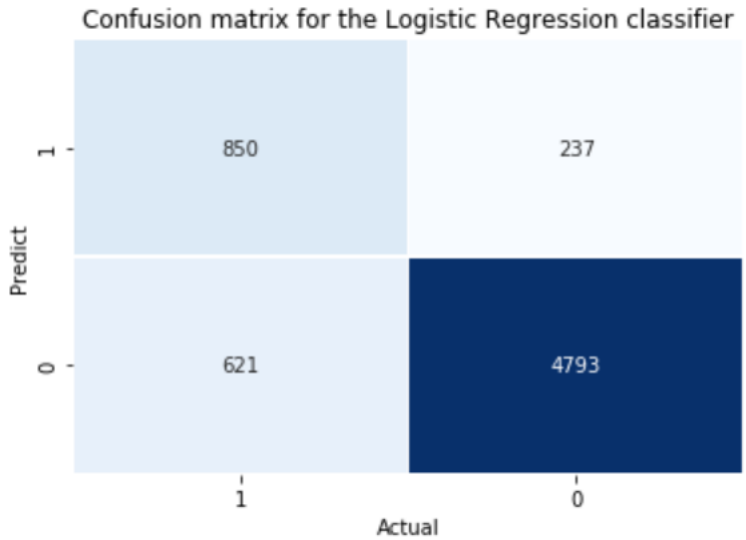
נספח 11- תרשים שונות מוסברת מצטברת ב-PCA עבור כלל העמודות הנומריות.



נספח 12- תרשים שונות מוסברת מצטברת ב-PCA עבור כלל העמודות הקטגוריאליות.

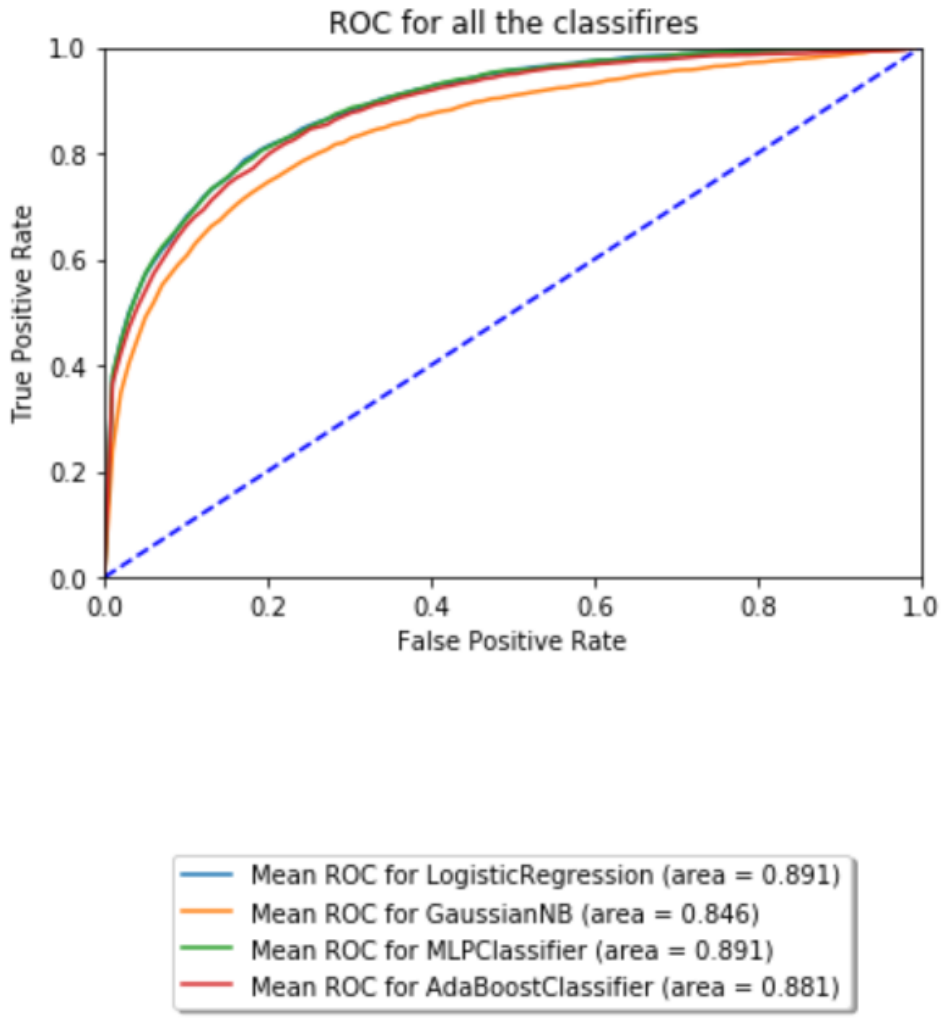


נספח 13- Confusion Matrix עבור המודל של Logistic Regression.



נספח 14- פלט ROC לכל K-Fold עבור כל אחד מ-4 המודלים שנבחרו עבור K=5.

```
The chosen k is: 5
```

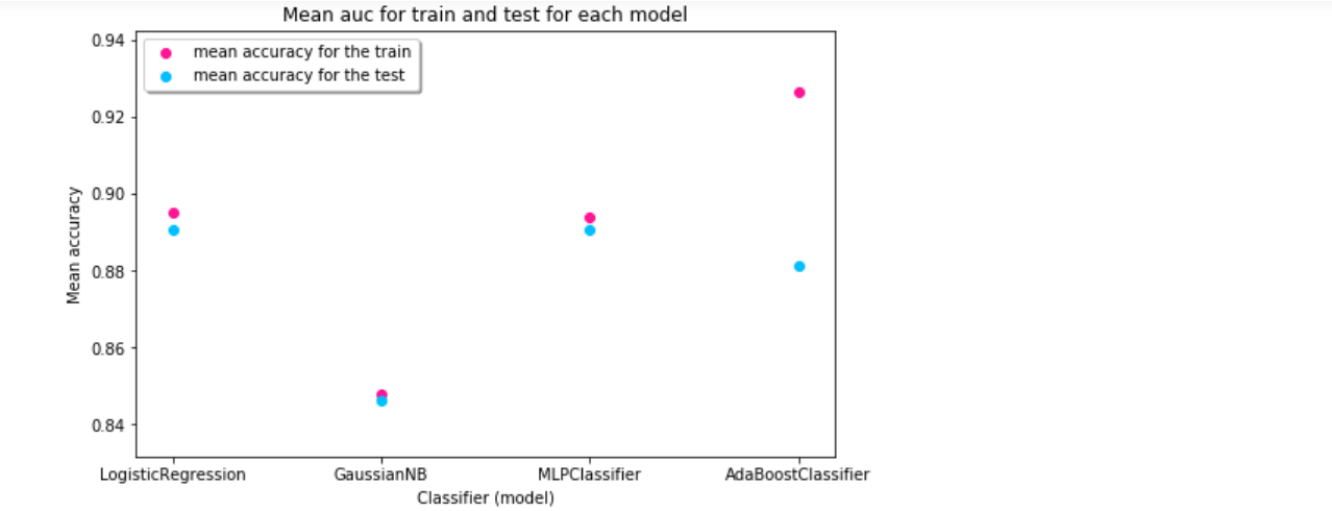


נספח 15- ביצוע בדיקות עבור המדד המשוקלל לצורך ווידוא כי אכן מתאים.

```
1* for tp=565 fp=0 tn=4663 fn=0 the assuracy is: 1.0
2* for tp=0 fp=208 tn=0 fn=643 the assuracy is: 0.0
3* for tp=565 fp=208 tn=4663 fn=643 the assuracy is: 0.604
4* for tp=565 fp=643 tn=4663 fn=208 the assuracy is: 0.756
5* for tp=565 fp=208 tn=4663 fn=0 the assuracy is: 0.962
6* for tp=565 fp=0 tn=4663 fn=643 the assuracy is: 0.619
7* for tp=0 fp=208 tn=4663 fn=643 the assuracy is: 0.577
8* for tp=565 fp=208 tn=0 fn=643 the assuracy is: 0.142
```

1* As we expected all classifications are correct and therefore the accuracy is 1.
2* As we expected, there is no correct classification so the accuracy is 0.
3* This is the accuracy with not manipulation.
4* We changed 3* so fn and fp were interchanged. That is, fn is the lowest and so is the punishment and therefore the accuracy is greater.
5* We changed 3* so fn=0. That is, there is no penalty at all for misclassification and therefore as expected accuracy increases.
6* We changed 3* so fp=0. That is, There are penalties but the number of incorrect classifications is small and therefore as expected the accuracy is smaller.
7* We changed 3* so tn=0. That is, we reduced the amount of correct classifications and therefore, as expected, the accuracy is smaller.
8* We changed 3* so tp=0.That is, we reduced the amount of correct classifications much more and therefore, as expected, the accuracy is much smaller.
This new measure does get accurate as expected in our tests and so we will use it.

נספח 16- תרשים המציג את פער הביצוע בין מדד הדיוק של סט האימון למדד הדיוק של סט הבחינה בצורה וויזואלית עבור כל אחד מ-4 המודלים שבחרנו.




```
If the gap is over than 0.03 the model is overfitted.
The performance gap for LogisticRegression classifier is: 0.004153059933663994
The performance gap for GaussianNB classifier is: 0.0013132089423177895
The performance gap for MLPClassifier classifier is: 0.003377052888832055
The performance gap for AdaBoostClassifier classifier is: 0.04502395860203934 ---> Above 0.03 -Overfitted
```

נספח 17- פירוט בחירת ההיפר פרמטרים עבור כל מודל.

- **Gaussian naïve bayes:** נחשב כמודל פשוט ומהיר למימוש מסווג כך שהוא מניח כי לפיצ'רים השונים אין השפעה ביניהם. למודל זה יש 2 היפר פרמטרים:
  - ✓ Priors: אחראי לקבוע את ההסתברויות הפריוריות עבור כל פיצ'ר. כיוון שאנו לא יודעים את משמעות הפיצ'רים בחרנו בערך הדיפולטיבי None כך שהמודל יסווג אוטומטית תוך התאמתו לנתונים.
  - ✓ Var smoothing: מדבר על השונות הגדולה ביותר של כל הפיצ'רים אשר נוספו לחישוב היציבות. הכנסנו עבורו את ערך ברירת המחדל 1e-9 ובנוסף ערכים שונים והטוב ביותר הוא 0.7.

- **Logistic regression:** מודל זה הינו מסווג אשר ידוע בחיזויו עבור תיוגים עם 2 ערכים כפי שנתון לנו בבעיית סיווג בינארית זו. המודל יעשה זאת תוך חיפוש מתאם בין משתנה מסביר למשתנה מוסבר. למודל זה יש כמה היפר פרמטרים:
  - ✓ Penalty: נועד לציון הנורמה בה ההענשה תתבצע כך שנבחר l1 כטוב ביותר.
  - ✓ C: יהיה הפוך מעוצמת הרגולריזציה כך שנבחר 0.3 כטוב ביותר.
  - ✓ Solver: ישמש כאלגוריתם בבעיית האופטימיזציה כך שבחרנו שיהיה liblinear שכן הוא יכול להיות עם כל קומבינציה של ההיפר פרמטרים האחרים
  - ✓ שאר הערכים נקבעו כברירת מחדל וביניהם:

    'class_weight': None, 'dual': False, 'fit_intercept': True, 'intercept_scaling': 1, 'l1_ratio': None, 'max_iter': 100, 'multi_class': 'auto', 'n_jobs': None, 'random_state': None, 'tol': 0.0001, 'verbose': 0, 'warm_start': False

- **ANN:** מודל זה ילמד ויתאמן על ידי רשת נוירונים. למודל זה כמה היפר פרמטרים וביניהם:
  - ✓ Activation: ישמש לבחירת פונקציית האקטיבציה עבור השכבות החבויות, כך שהכי טוב שנבחר הוא logistic.
  - ✓ Hidden layer sizes: ישמש לבחירת אופן מבנה רשת הנוירונים עבור השכבות, כך שהכי טוב שנבחר הוא (100,100).
  - ✓ Batch size: משמש לאופן ההתייחסות ל-epoch שלאחריו יתבצע עדכון משקולות, כך שהכי טוב שנבחר הוא 10. לפרמטר זה יתרון כאשר הוא קטן כך שהמודל יכול להתכנס יותר מהר.
  - ✓ Learning rate init: משמש לקביעת קצב הלמידה של המודל, כך שהטוב ביותר שנבחר הוא 1e-5.
  - ✓ Max iter: משמש לקביעת מספר האיטרציות המרבי, כך שהטוב ביותר שנבחר הוא 1250.

- **Adaptive Boosting:** מודל זה משתמש בהיפר פרמטר base estimator כברירת מחדל באופן כזה שהוא מכיל מספר עצי החלטה כך שהעומק המקסימלי שלהם הוא 1. במודל זה כל עץ החלטה ילמד מדגימות ופיצ'רים אקראיים באופן כזה שינסה לתקן את הטעויות של העצים הקודמים. היפר פרמטרים נוספים של המודל הם:
  - ✓ n estimator: מספר עצי ההחלטה המירבי לפיו המודל ילמד, כך שהטוב ביותר שנבחר הוא 400.
  - ✓ Learning rate: קצב למידה שיצמצם את תרומתו של כל מסווג (עץ החלטה), כך שהטוב ביותר שנבחר הוא 0.5.
  - ✓ Algorithm: בחירת האלגוריתם עבור עבור המודל, כך שהטוב ביותר שנבחר הוא SAMME.R.
  - ✓ Random_state: נבחר כערך ברירת המחדל עבור ה-seed כלומר כ-None.

<u>נספח 18-</u> תוצאות עבור מדד הדיוק המשוקלל לכל מודל שנבחר.

```
The accuracy with this new measure for the LogisticRegression is: 0.628
The accuracy with this new measure for the GaussianNB is: 0.452
The accuracy with this new measure for the MLPClassifier is: 0.631
The accuracy with this new measure for the AdaBoostClassifier is: 0.619
```