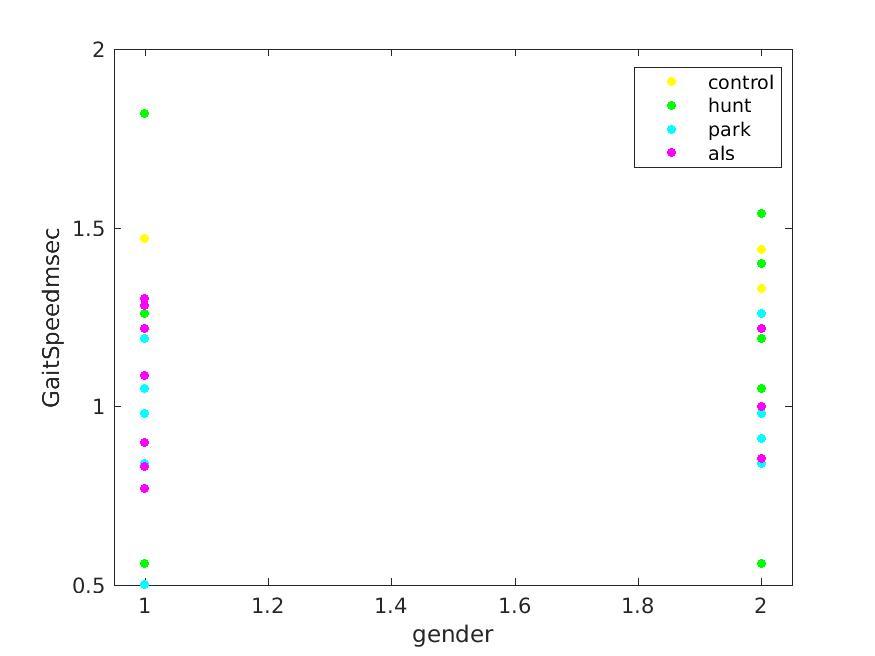
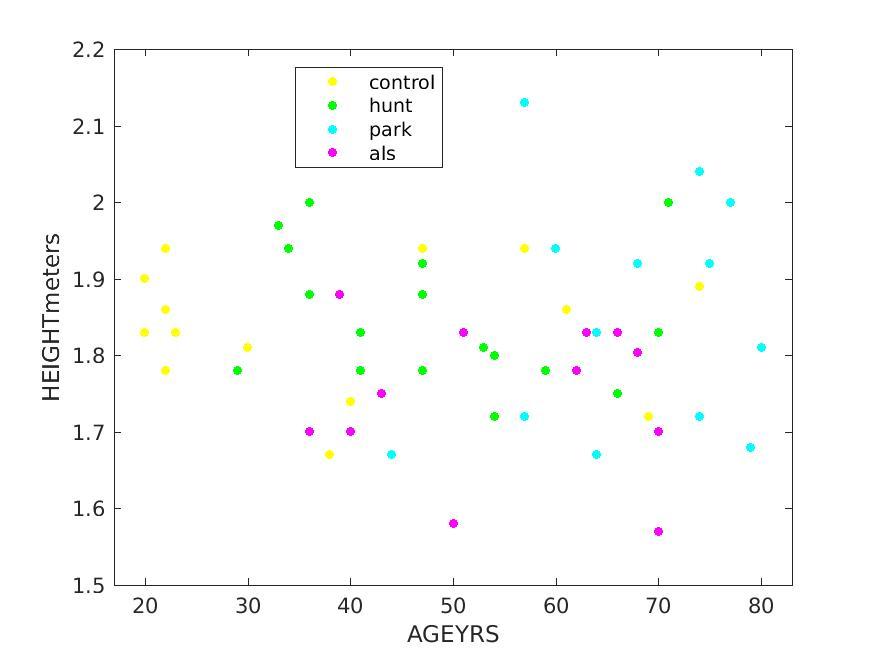
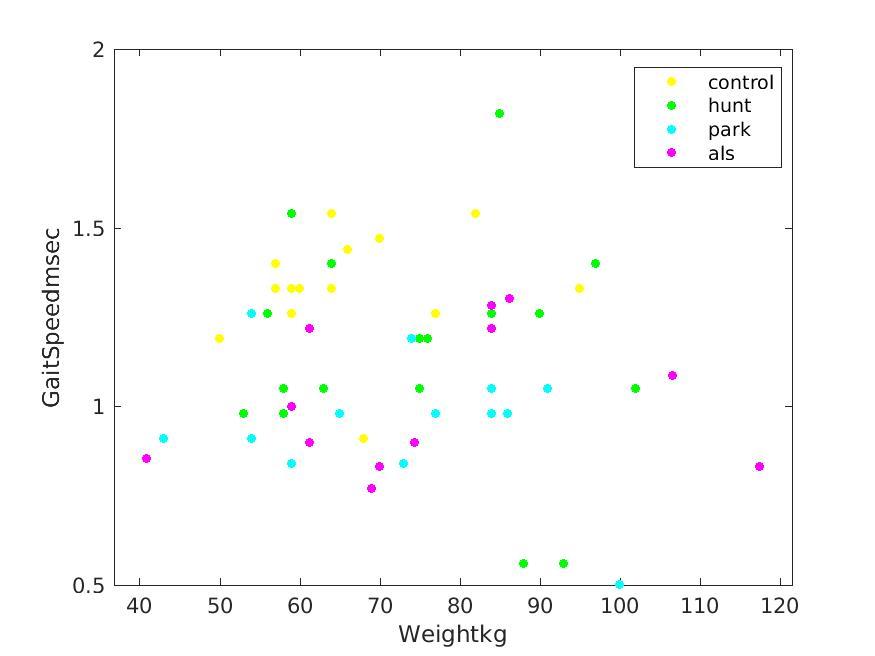
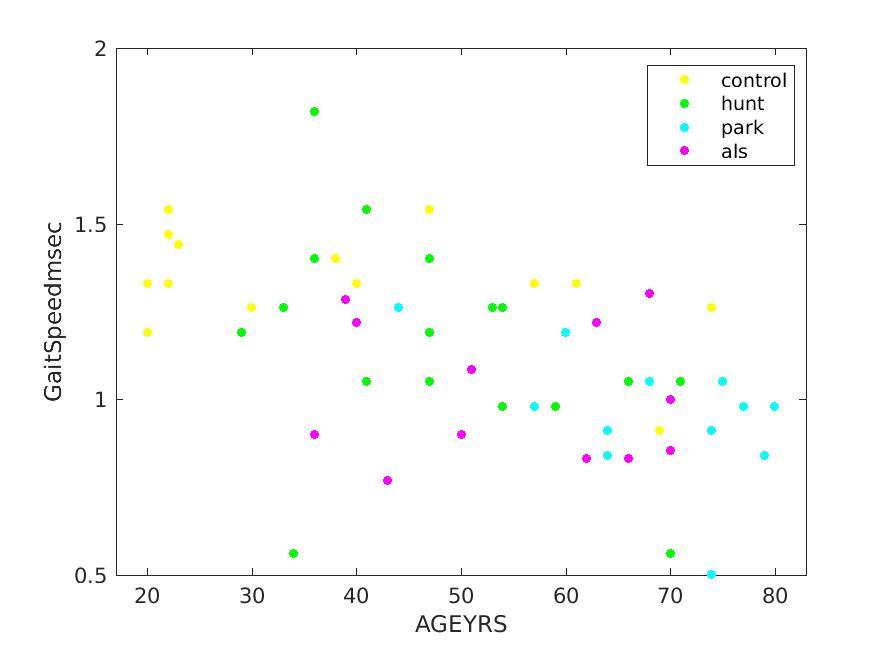
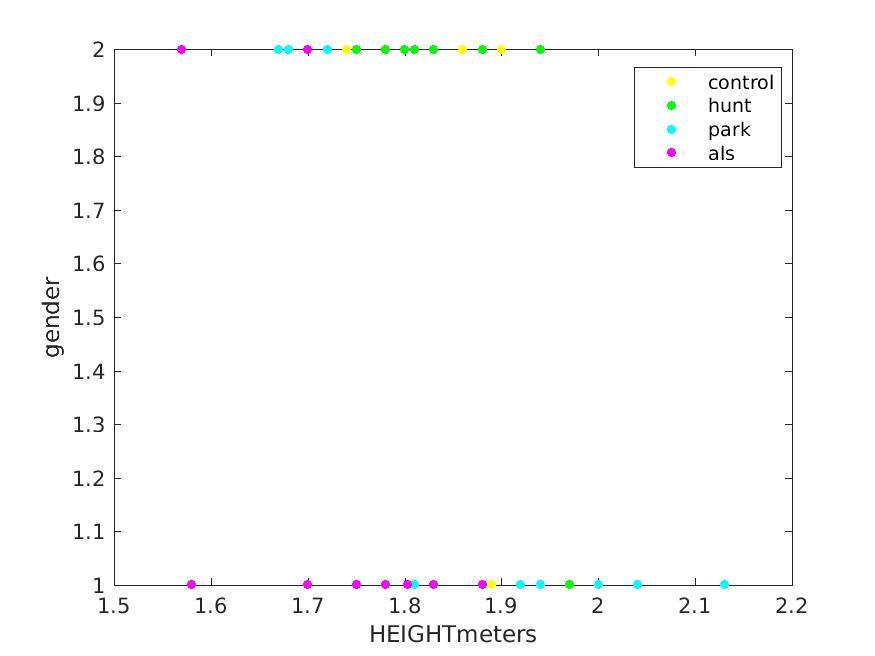
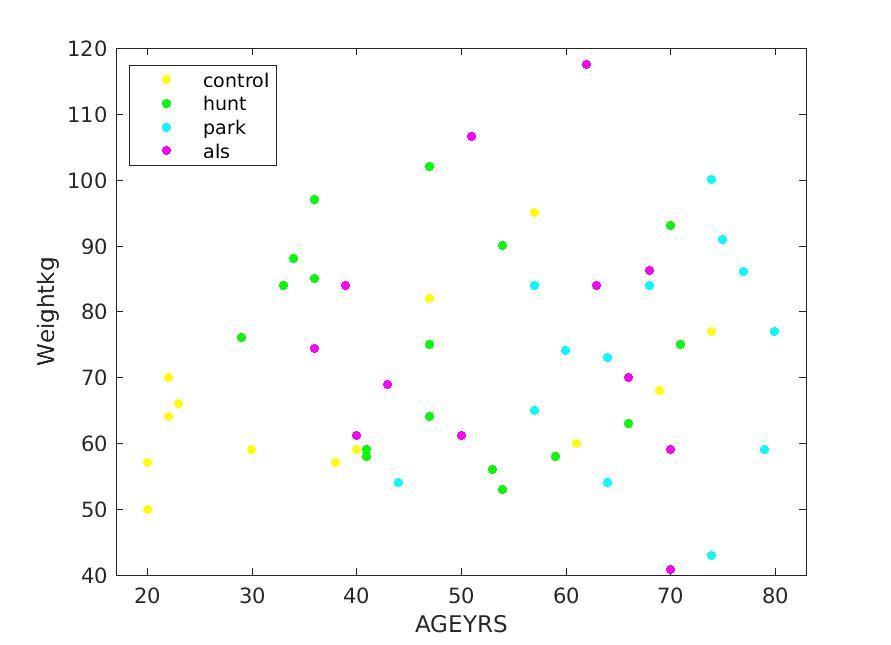
**All Data in Assets folder**

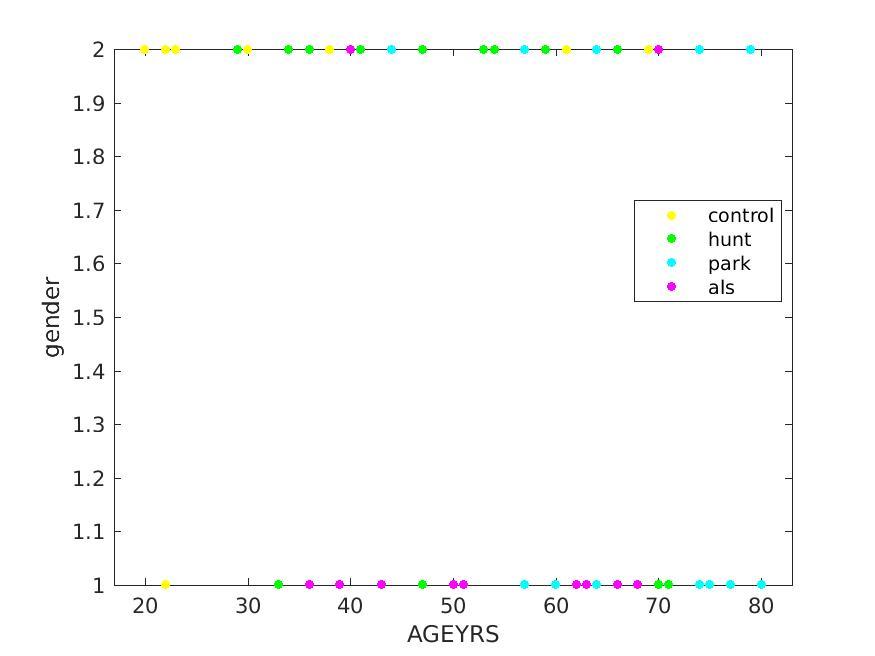
1. ***התמקדו ראשית רק בטבלה subjectdescriptionParTrainPerm :***

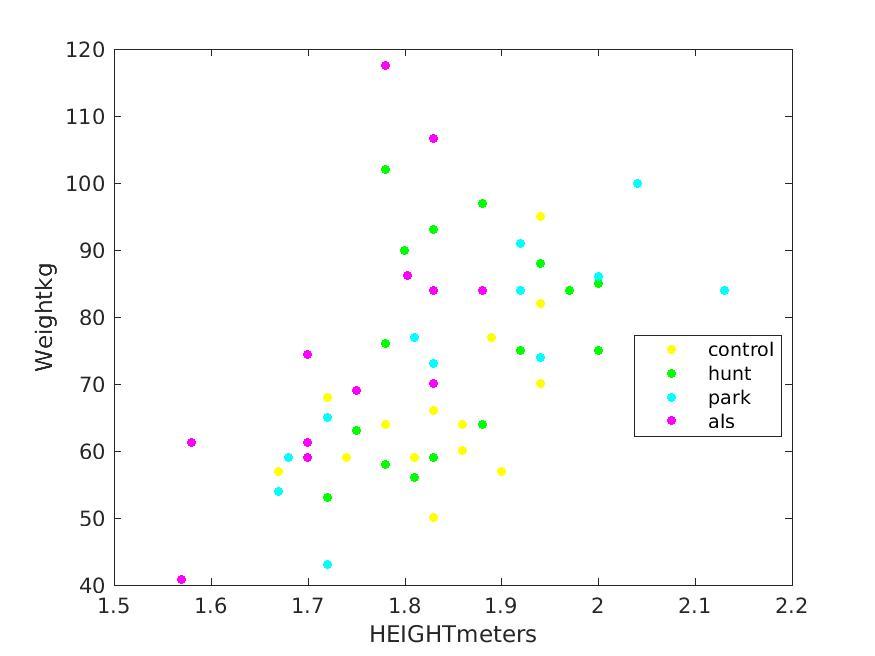
***השתמשו בפקודה scatter על מנת לצייר את הנתונים לפי זוגות משתנים כאשר קבוצות נבדקים שונות צבועות בצבעים שונים. בהתבסס על הגרפים שקיבלתם. הסבירו בקצרה אילו מחמשת הנתונים יהיו משמעותיים ובאילו קומבינציות כדאי להשתמש. מומלץ לצרף לדו"ח גרפים על מנת לנמק את ההסבר שלכם:***

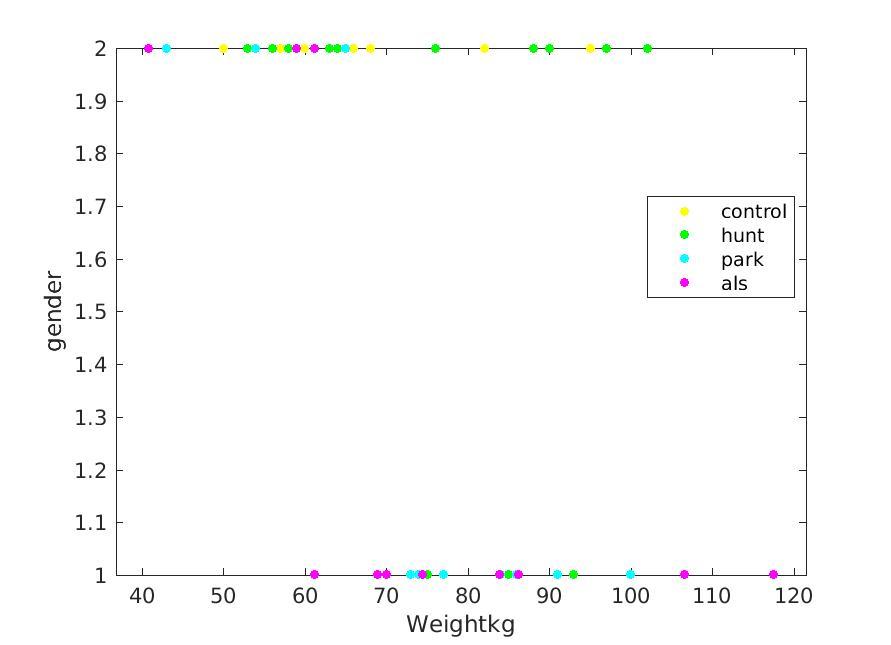
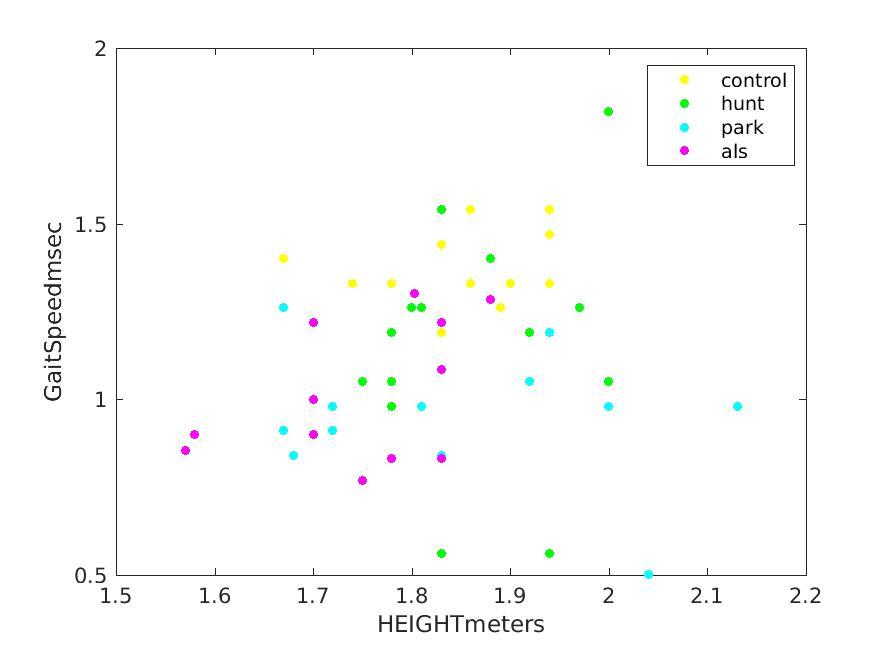












הסבר :

כפי שניתן לראות מהגרפים השונים מהירות וגיל הם הנתונים המשמעותיים ביותר הבאים לידי ביטוי גם בקומבינציות עם נתונים אחרים .

* **מהירות** - ניתן לראות כי קבוצת חולי הפרקינסון איטיים יותר משאר הקבוצות ומהירות קבוצת הביקורת קבועה . לעומתם , מהירות חולי הנטינגטון וALS מפוזרת לאורך הSCATTERS .
* **גיל** – ניתן לראות כי חולי הפרקניסון מבוגרים משמעותית משאר החולים. וכי קבוצת הביקורת רובם צעירים. גילאי שאר החולים שונים ומגוונים.

בנוסף לכך קומבינציות מסוימות מחלקות טוב יותר ( נראה לעין ) בין הקבוצות :

* **משקל וגיל** – קבוצת הביקורת צעירה ובעלת משקל נמוך ברובה .לעומתם פרקינסון מבוגרת יותר ובמשקלים שונים.
* **גיל וגובה**- ניתן לראות שמרבית מקבוצת הביקורת צעירה וגובהה בסביבות ה1.85 .
* **מהירות ומשקל** - ניתן לראות שקבוצת הפרקינסון מהירות זהה בסביבות 1 וכי קבוצת הביקורת במשקלים דומים ובמהירות 1.4 .

ישנם קומבינציות שלא עוזרות כמעט כלל כגון גיל ומין ומשקל גובה שבסהכ נראית קורלציה חיובית בין משתנים אלה בין כל הקבצות .

***חלקו את הנתונים ל- 5 קבוצות על מנת לבצע CROSS VALIDATION שימו לב***

***שאתם שומרים על מספר סביר של מקרים בכל קבוצה.***

ממומש בקוד. (ind=crossvalind('Kfold',group,5

***הפונקציה fitctree מקבלת טבלאות נתונים ורשימת labels תואמת ויוצרת מהם עץ החלטה. לפונקציה זו יש מספר משתנים שמייצגים היפרפרמטרים. עליכם למצוא את הקומבינציה האופטימלית של היפרפרמטרים. הציגו את העץ הזוכה עם פונקציית view והסבירו בבירור ובקצרה איך הגעתם להחלטה זו:***

הסבר:

בהתחלה ניסינו בצורה ידנית לשנות את ההיפר פרמטרים וראינו עד כמה תלוי היפר פרמטר אחד בערכו של האחר. לכן, בחרנו לבנות לולאה שלמעשה מגרילה בצורה רנדומלית כל פעם היפר-פרמטרים שונים. כל ריצה בלולאה מבוצעת 5 פעמים בלולאה נוספת עבור כל אחד מקבוצות הוולידציה ובודקת את אחוזי השגיאה הכוללים עבור קבוצת האימון ועבור קבוצת המבחן. לאחר סיום הריצה מצאנו את אחוזי השגיאה הנמוכים ביותר( 39.24%) ובחרנו בהיפר-פרמטרים של אותו העץ. על מנת לוודא שההיפר הפרמטרים אכן נכונים וזה לא מקרי, בדקנו בטבלה אחוזי שגיאה נמוכים אחרים ומצאנו שמרבית ההיפר-פרמטרים אכן חופפים.

ההיפר פרמטרים שנבחרו הם:

allsplits- PredictorSelection

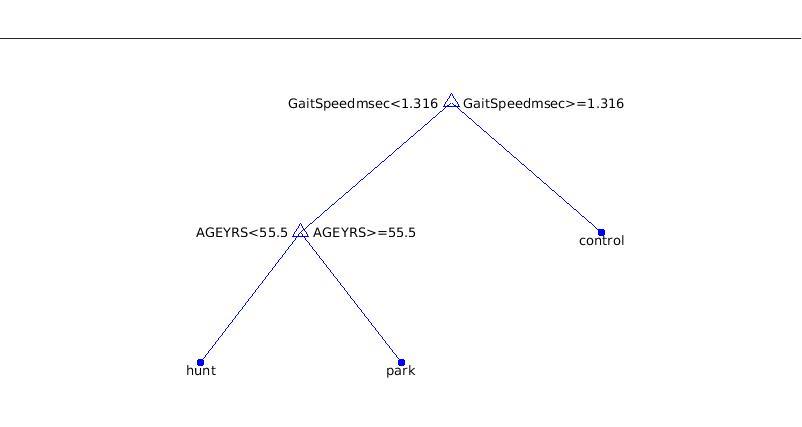
מספר חלוקות - 4

PRUNE - “אוף"

סט משקולות רנדומלי: 50 משקולות רנדומליות.

כל הדאטה נמצאת ב- “1bc\_data.mat” תחת המשתנה bestOptions. והעץ עצמו תחת המשתנה bestTREE.

העץ הזוכה הוא :



***כעת השתמשו במנגנון האוטומטי של הפונקציה לבחירת HYPERPARAMETERS ע"י***

***הקלט " OptimizeHyperparameters " והשוו את התוצאות לעצים הקודמים תארו***

***את ההבדלים אם יש כאלו, ונסו להסביר ממה הם נובעים:***

הסבר:

יש מספר הבדלים בין העץ שיצא לנו לעץ האופטימלי מהמנגנון האוטומטי:

דומה- allsplits- PredictorSelection

שונה- ה Prune שלנו נבחר על "אוף" ושלהם על "און"

מספר החלוקות המקסימלי הוא משמעותי שונה (50 באופטימלי לעומת 4 אצלנו) כמות המשקולות- העץ האופטימלי בוחר משקולות עבור כל הנתונים ואנחנו בחרנו בצורה אקראית ואימנו את העץ רק על סמך כמות הדאטה שיש לנו בקבוצת האימון.

ההבדלים נובעים ממספר בחינות:

* ניתן לראות שבסופו של דבר העץ שלנו מאוד מצומצם, כלומר רק 2 חלוקות עבור כל הדאטה. למרות שאיפשרנו לו 4 חלוקות התוצאה הסופית הייתה 2 חלוקות בלבד. העץ האוטימלי מאפשר המון חלוקות ובוחר לעשות גיזום על מנת לייעל את העץ. במודל האופטימלי 5 חלוקות. יכול מאוד להיות שהאימון שהתבצע על העץ האופטימלי עם כל הדאטה גרם ל-OVERFITTING , הדאטה מתאימה מאוד לקבוצת האימון. היות ולא ביצענו CV עבור האופטמיזטור, יכול להיות שמצא מודל טוב עבור הדאטה שאימן אבל עלול להיות ספציפי מדיי עבור דוגמאות נוספות.
* אנחנו אימונו את המודל בCV כלומר, 5 פעמים עם קבוצת ולידציה משתנה. ומיצענו את אחוזי השגיאה שלנו גם עבור קבוצת האימון וגם עבור קבוצת המבחן. לעומתנו, האופטמיזטור מוצא את ההיפר פרמטרים הטובים ביותר עבור כל הדאטה.

יכול להיות שבאופן מאוד מקרי, העץ שקיבלנו טוב מאוד עבור כל מבחני הCV בחלוקה הרלוונטית, אך לא יהיה טוב מספיק עבור חלוקה אחרת.

על מנת לבדוק את השערתנו בנוגע לשינוי, הרצנו את האופטמיזטור עבור 4/5 מהדאטה. קיבלנו עץ מעט שונה אבל עם היפר פרמטרים דומים לזה שקיבלנו עבור כל הדאטה.

***השתמשו בפונקציה predictorImportance כדי לדרג את המשתנים לפי החשיבות לקלסיפיקציה. הסבירו את תוצאות בהשוואה להשערות שלכם בסעיף a ואיך הן מתבטאות בעץ ההחלטה מסעיף d :***

preImp =(עבור העץ הטוב ביותר שקיבלנו)

0.0316 0 0 0 0.0751

preImpDT =(עבור עץ שבחר האופטימיזטור)

0.0144 0.0207 0 0.0121 0.0210

הסבר:

ניתן לראות, שהפרמטרים המהירות והגיל הם בעלי החשיבות הכי גדולה לקלספיקציה! בדיוק כמו ששיערנו בסעיף הראשון. גם עבור האופטמיזטור יש להם חשיבות אך הוא דירג אותם במקומות 1,3 ודווקא המין מבחינתו במקום ה-2. מכאן, שקיבלנו את אותם משתנים ששיערנו שיהיה החשובים ביותר. וגם האופטימזטור סימן אותם כחשובים מאוד.

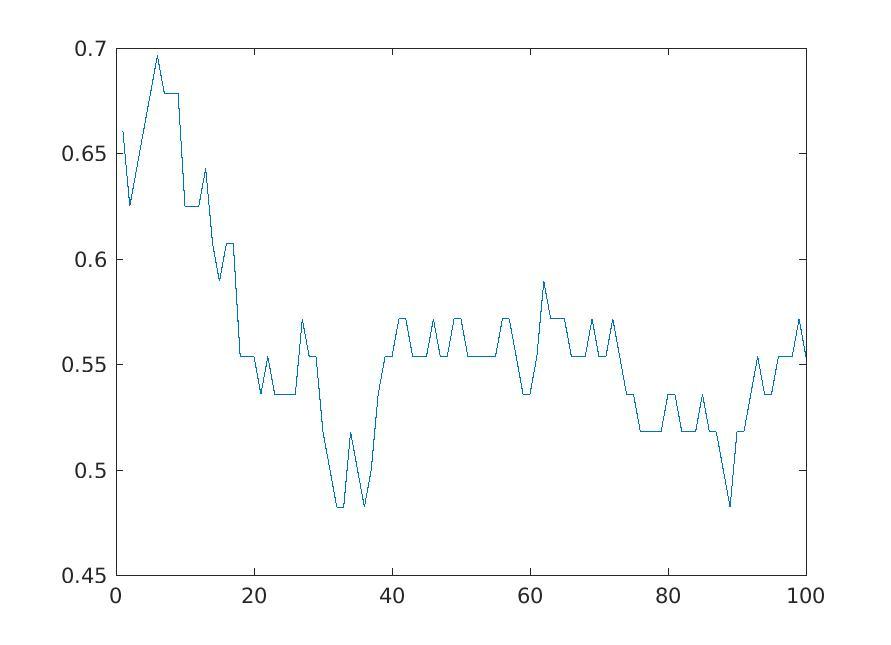
***כעת בדקו אם אוסף של עצים יכול להשיג תוצאות טובות יותר מאשר עץ בודד***

***השתמשו בפונקציה TreeBagger לייצר 100 עצים . ציירו את הטעות כפונקציה של***

***כמות העצים ע"י הפונקציה oobError השוו את התוצאות לסעיפים הקודמים :***

הסבר:

ניתן לראות שהטעות הנמוכה ביותר של העצים היא עבור TreeBagger של 32,33,36,89 עצים שעבור כולם הטעות היא 48.21%. התוצאה הזאת נמוכה מהתוצאה שהצלחנו להגיע אליה בעזרת האימון בסעיף ג’. דבר זה מחזק את התחושה שהעץ שקיבלנו כטוב ביותר הוא ב-OVERFITTING עבור קבוצת המבחן, אך זה מקרי ותלוי בחלוקה. תוצאות ה- TreeBagger דומה יותר לעץ האופטימלי שהוציא האופטימזטור. יחד עם זאת, כיצד המודל מגיע לפרדיקציה עצי RANDOM FOREST .

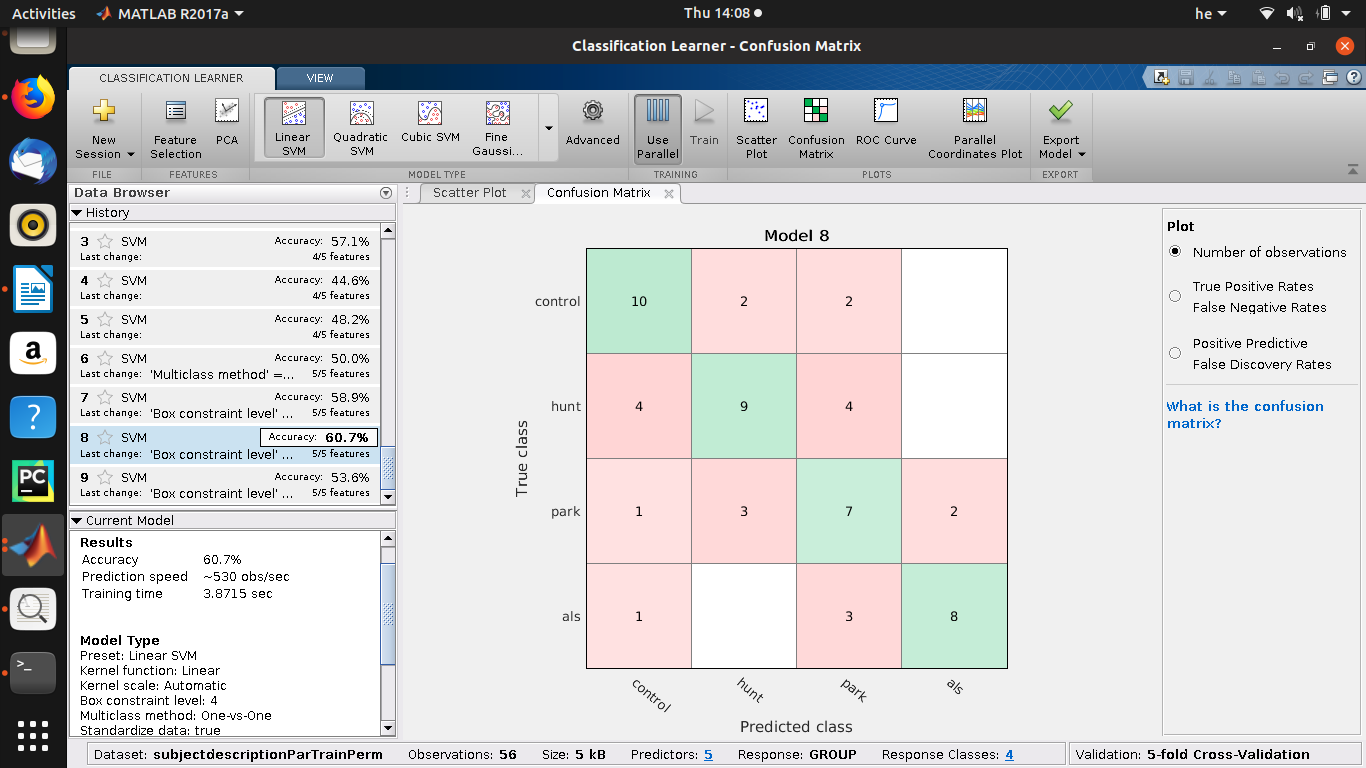


***העלו את הדטה עם 5 FOLD VALIDATION ,בדקו לפחות סוג אחד של SVM , סוג אחד של TREES וסוג אחד של ENSEMBLE. שחקו עם הפרמטרים שלהם בכפתור ADVANCED . מצאו את הקלסיפיר שמביא לכם את התוצאות הכי טובות והסבירו בגדול כיצד הוא עובד.***

הסבר:

הקלסיפיר שהגיע ל-60.7 אחוזי דיוק היה הSVM הליניארי עם BOX CONSTRAINT LEVEL השווה ל-4, כלומר איפשרנו לו לטעות מעט בקלסיפיקציה. ובכך שיפרנו את המודל בכ-2 אחוז. עצי החלטה וRANDOM FOREST נתנו תוצאות נמוכות יותר.

מה שעשינו היה למעשה לבחור פיצים לאימון. בהתחלה בחרנו לאמן רק עם המהירות והגיל וקיבלנו תוצאות נמוכות, בעיקר עבור קבוצת ההנטיגטון והALS לכן חזרנו ל-5 פיצרים. מאוחר יותר שינינו מעט את הADVANCED, שחלקן ביצענו כבר כאשר ביקשו הAPP לאמן את כל המודל האפשריים (ALL). אימנו במקביל ע"י PARALLEL ובסה"כ ניתן לאמן ולראות תמונה מעניינת גם ב SCATTER PLOT וגם ב CONFUSION MATRIX.



1. **חקרו את נתוני ההליכה במערך GateTrainDataCellPerm:**

***בחרו 3 נבדקים מכל קבוצה. לכל עמודה )חוץ מהראשונה שהיא זמן( ציירו )ב-( PLOT) את 12 העמודות של 12 הנבדקים שבחרתם( צבועים לפי הקבוצות) . ציינו אילו הבדלים אתם רואים במדדים השונים בין הקבוצות ואילו מדדים יוכלו לעזור לכם בקלסיפיקציה.***

הסבר:

המדדים למעשה חוזרים על עצמם בצורה כזו או אחרת כי פעם אחת באים לידי ביטוי בערך (משך זמן) ופעם אחת באחוזים (ביחס הפנה ועמידה למשך זמן הצעד הכולל, או זמן העמידה למשך הזמן הכולל של צעד+עמידה וכדומה. לכן, ה PLOTS יוצאים קצת כאלמנטים חוזרים:

**מצורף בתיקייה ( קבצי cellperm):**

מסודר לפי המדדים:

* משך צעד שמאל 1cellperm.jpg
* משך צעד ימין 2cellperm.jpg
* משך הנפה שמאל 3cellperm.jpg
* משך הנפה ימין 4cellperm.jpg
* משך הנפה שמאל באחוזים 5cellperm.jpg
* משך הנפה ימין באחוזים 6cellperm.jpg

משך עמידה שמאל 7cellperm.jpg

* משך עמידה ימין 8cellperm.jpg
* משך עמידה שמאל באחוזים 9cellperm.jpg
* משך עמידה ימין באחוזים 10cellperm.jpg
* משך עמידה על שני רגליים 11cellperm.jpg
* משך עמידה על שני רגליים באחוזים 12cellperm.jpg

מהגרפים ניתן לראות שהנטיגטון וALS בעלי משך צעד ימין ושמאל ארוכים (עמודות 2,3, גרפים 1,2), ניתן לראות שונות גדולה במשך הנפה שמאל וימין באחוזים (עמודות 6,7 וגרפים 5,6) עבור כולם מלבד קבוצת הביקורת. בהנטיגטון שונות גדולה ובALS חריגים (קושי לסיים תנועה). הנטינגטון יותר זמן בעמידה שמאל באחוזים (עמודה 11, גרף 10), ושונות גדולה בעמידה על 2 רגליים באחוזים בין הקבוצות (עמודה 13, גרף 12).

**בחרו 6 מתוך העמודות חשבו להם ממוצע על פני הזמן לכל נבדק:**

ממומש בקוד. בנוסף ביצענו T-TEST על מנת לוודא שבחרנו נכון וקיבלנו שעמודות 2,3,6,7,9,11,12 הם עם הכי הרבה T-TEST סגניפיקנטי בין הקבוצות. אנחנו סימנו את 2,3,6,7,11,13 שזה כמעט אותו הדבר. לאחר דיון משותף החלטנו להתקדם עם העמודות: 2, 3, 6, 7, 11, 12 (aveAll) .

**כעת הפעילו את העצים האופטימליים שקיבלתם בסעיפים 1c,d,f על 6 עמודות אלו והשוו תוצאות:**

אחוז השגיאה עלו משמעותית עבור כל אחד העצים האופטימליים שהיו בשאלה הראשונה.

עבור העץ שאנחנו בנינו עבור המדדים הכללים על הנבדק- קיבלנו אחוז שגיאה גבוה של 58.31% ועבור העץ האופטימלי שבנה האופטימזטור 50.68% שגיאה. אחוזי שגיאה נמוכים בהרבה בשאלה הקודמת מעידים על כך ש-2 העצים הללו לא מתאימים כלל לפיצ’רים החדשים.

בTREEBAGGER התקבל 57.14% עבור 12 עצים, לכן גם הוא אינו מתאים לפיצ’רים החדשים.

עץ שנבחר על ידינו אחוז שגיאה 58.31:

TREE2B1.JPG

עץ של אופטימיזטור אחוז שגיאה 50.68:

TREE2B2.JPG

מרובה עצים:

TREEBAGGER2.JPG

***כעת הוסיפו את הממוצעים כעמודות נוספות לטבלה subjectdescriptionParTrain .***

***הפעילו את העצים האופטימלים שקיבלתם בסעיפים 1c,d,f על הדטה החדש, ציירו אותם השוו תוצאות, ציינו האם חל שיפור בקלסיפיקציה. הסבירו את התוצאות.***

הסבר:

חל שיפור משמעותי לאחר הוספת הפיצ’רים שבעזרתם הוחלט על הפיצרים האופטימליים למודל. בעץ שאנחנו מצאנו קיבלנו 41.42% דיוק, בעץ האופטימלי 40.33% דיוק שזה שיפור של 10-20% מהסעיף הקודם. ובTREEBAGGER הכי טוב שהתקבל היה 46.43% עבור 5 עצים.

הפיצרים האלה מתאימים למודלים שבנינו בשאלה ה-1 ולכן התוצאות השתפרו משמעותית. יחד עם זאת, התוצאות לא יותר טובות מהתוצאות שהתקבלו בשאלה הראשונה למרות הוספת הפיצרים לכן יהיה עלינו לבחון באיזה פיצרים יכולים להשתמש והאם צריך לבנות מודל חדש לגמרי עבור הפיצרים החדשים.

עץ שנבחר על ידינו אחוז שגיאה 41.42:

TREE2C1.JPG

עץ של האופטמיזטור אחוז שגיאה 40.33:

TREE2C2.JPG

מרובה עצים:

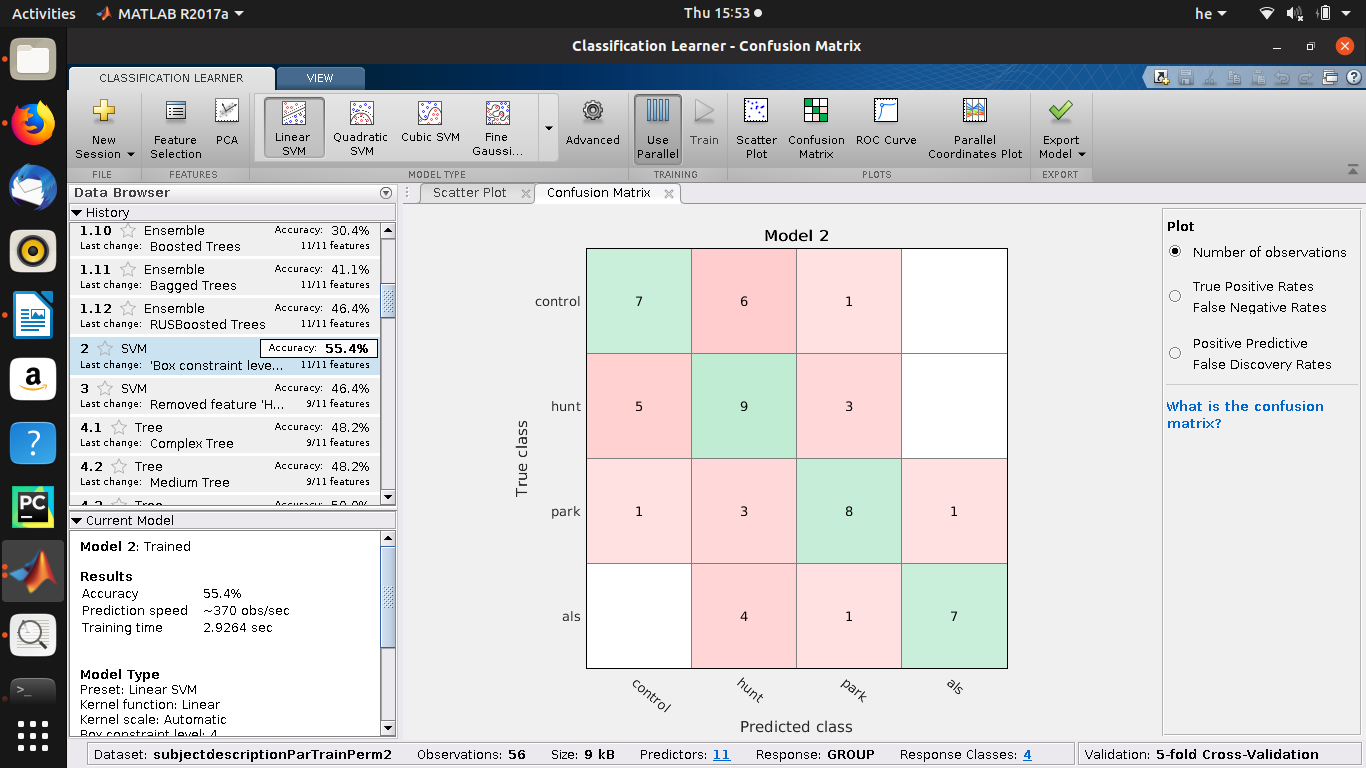
TREEBAGGER3.JPG

***כעת שחקו עם classification Learner App על הטבלה המעודכנת ומצאו קלסיפייר***

***אופטימלי. שמרו אותו ע"י EXPORT MODEL :***

הסבר:

לאחר שניסינו מספר אופיציות ומודלים שונים התוצאה הכי טובה שהתקבלה הייתה SVM עם CONSTRAINT 4 – קיבלנו 55.4% דיוק.

נשמר בקובץ: כ-SVM2CMODEL (בקובץ – DATA1.MAT\_2)

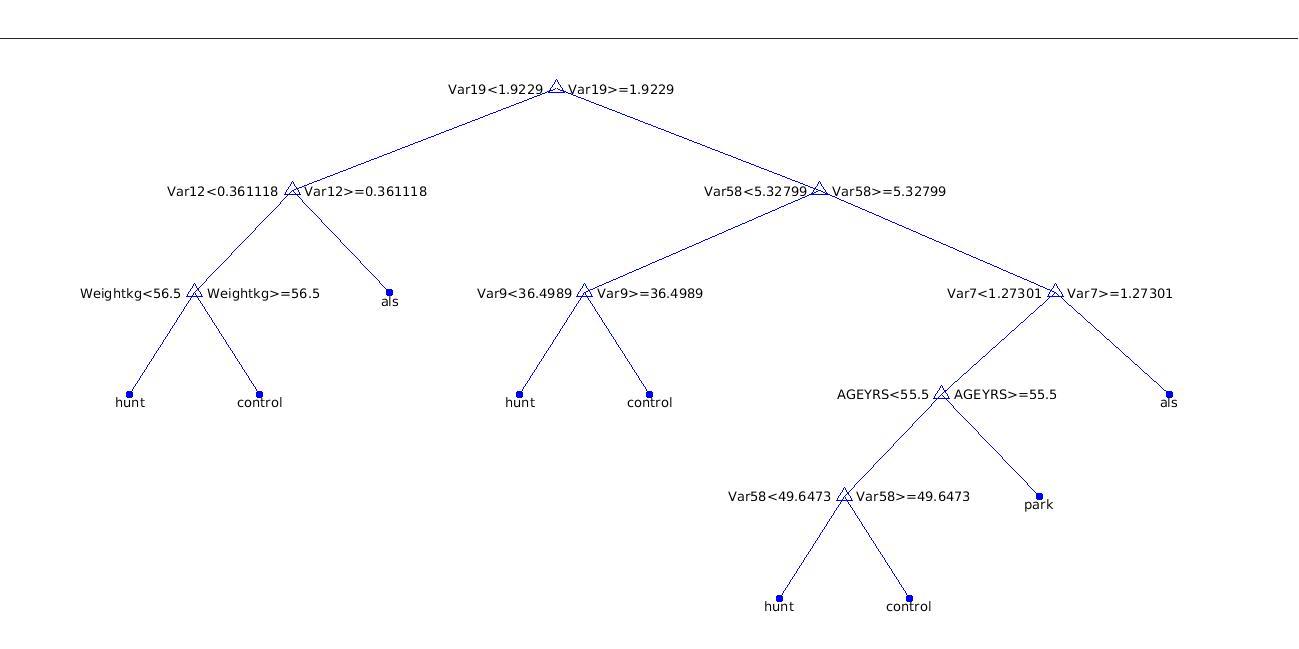
***3. הוסיפו 2 מדדים נוספים לבחירתכם המחושבים ממערך ההליכה. ניתן להשתמש בסטיית***

***התקן של עמודה, ב SKEWNESS ו KURTOSIS . וניתן לבחור מאיזה עמודות להשתמש.***

***השתמשו ב classification Learner App על מנת למצוא מודל קלסיפיקציה אופטימלי. שמרו מודל זה. הסבירו אותו:***

הסבר:

על מנת למצוא מהם המדדים הנוספים שיעזרו לנו לשפר את המודל מדדנו את הערכים השונים (ס"ת, שונות, SKEWNESS ו KURTOSIS ) עבור כל עמודה בכל נבדק. לאחר מכן ביצעו T-TEST בין כל הקבוצות בכל עמודה ובדקנו מהו המדד שהוא הכי הרבה פעמים סגניטיפיקנטי בין הקבוצות השונות. קיבלנו תוצאות טובות עבור ס"ת והשונות ולכן בחרנו במדדים אילו שיתווספו לפיצרים שעליהם נעבוד.השתמשנו במדד לס"ת ולשונות (משך הנפה ימין באחוזים, משך עמידה שמאל באחוזים בהתאמה) – וקיבלנו את העץ MEDIUMTREE עם 71.3% דיוק. העץ הוא אמנם פשוט קלסיפיקציה פשוטה בהגדרתה אך הוספת הפיצרים הועילה מאוד לחלוקות בעץ ככה שיכל לסווג נכון יותר את הקבוצות השונות. יפה לראות, שהוספת 2 פיצרים בלבד העלתה את אחוזי הדיוק ביותר מ-15%. אם היה למודל קושי להפריד בין הקבוצות הוספת הפיצרים עזרה לו לסדר את הדאטה בצורה טובה יותר.



היות ובהוראות התבקשנו להשתמש במדדים ספציפיים ביצעו את המודל רק עבור ס"ת - משך הנפה ימין באחוזים, משך עמידה ימין באחוזים – קיבלנו שמודל ה-ENSEMBLE (TREEBAGGER) הוא המודל הטוב ביותר עם 62.5% דיוק. TREEBAGGER הוא מצויין כאשר עוסקים במספר רב של פיצרים. לעץ יש נטייה להגיע לOVERFITTING כך שלאחר האימון התוצאות בקבוצת המבחן לרוב עלולות להיות לא טובות, במיוחד שמדובר במספר גדול של פיצרים. לכן המודל הזה הוא מודל טוב כאשר יש לנו כבר 13 פיצרים באימון. מכאן, המשכנו לבצע את הבונוס מתוך הנחה שהפיצרים הנוכחים לא מובילים אותנו לתוצאות טובות מאוד, וכי עלינו להוסיף פיצרים נוספים. אם כך, נשתמש בשלב הבונוס בעיקר בTREEBAGGER.

***בונוס:***

תחילה השתמשנו ב-59 (11+48) וניסינו למצוא את ההיפרפרמטרים האופטימליים- קיבלנו שגיאה מאוד קטנה עבור קבוצת האימון 13.44% אך שגיאה גדולה מאוד עבור קבוצת המבחן 52.04% (בקרוס ולידציה). לאחר מכן ניסינו TREEBAGGER, בעיקר על מנת לאשש את השערתנו מהשאלה הקודמת שמודל מסוג זה יהיה יעיל במספר רב של פיצרים, הרבה יותר מעצים בודדים.

קיבלנו תוצאות יפות מאוד- שגיאה של 33.93% עבור 46 עצים. הבנו שאנחנו בכיוון הנכון, הוצאנו מעט פיצרים על מנת שנוכל לאפיין ולעבוד איתם (subjectdescriptionParTrainPerm6) ובנוסף ביקשנו להוציא חריגים- לכן השתמשנו בפונקציית isoutlier- והוצאנו את הנבדקים- 14,26,31. לאחר מספר בדיקות שביצענו בחרנו להוציא במקומם את 10,21,30,35,36. שגם הם הוציאו חריגים מספר פעמים בבדיקות שביצענו. (הגענו למסקנה זו לאחר בדיקה עם ה-APP).

בדקנו עוד המון מדדים נוספים:

* הערך המקסימלי בכל עמודה-לדוגמא-קושי בתחילת תנועה
* הערך המינימלי בכל עמודה –לדוגמא- הליכה מאוד מהירה ולא יציבה
* הפרש הערך המקסימלי והמינימלי- לדוגמא- אצל קבוצת הביקורת ערך זה הכי מינימלי
* ההפרשים בין עמודות (כמו לדוגמא משך זמן עמידה רגל ימין מול רגל שמאל..)בהקשר זה חושבו מדדים שהתעלמנו מהם מאוחר יותר.
* סכום בין עמודות (מאותה הסיבה)
* הפרש אבסולוטי עמידה רגל ימין לעמידה רגל שמאל- ראינו שונות גדולה בין הקבוצות במדד זה.
* זמן הנפה כולל ממוצע- קבוצות חולי פרקינסון, הנטיגנטון ALS וקבוצת הביקורת מראים שונות רבה גם בסרטונים שצפינו בהם וגם במדדים שהעבירו לנו. לדוגמא חולה ALS מתקשים לשלוט ברגליים, לרוב נעזרים במישהו ומשך ההנפה גדול. לעומתם, חולי פרקינסון הולכים בצעדים קטנים מאוד בלי כמעט הנפה. לכן, בחרנו במדד זה. כנ"ל לגבי משך צעד.
* ממוצע ההפרש האבוסלוטי במשך צעד ימין ושמאל באחוזים- בהליכה רגילה מראים דומיננטיות מסויימת ברגל אחת. הקבוצות השונות מראות מדדים שונים מקבוצת הביקורת.
* שונות זמן הליכה כולל- השלמת כל מעגל הצעד- מתוך הנחה שקבוצת הביקורת תהייה בעלת השונות הנמוכה ביותר והקבוצות האחרות יהיו בעלי שונות גבוהה בגלל הקושי בהליכה שלהם.

בסופו של דבר הוספנו רק את 5 המדדים האחרונים שפירטנו עליהם.

לאחר מכן, אימנו 1200 פעמים את מודל עץ רגיל וTREEBAGGER עם היפר-פרמטרים משתנים- את כמות הנבדקים השארנו קבועה אך שנינו את:

* הפיצרים- כולל כמות משתנה
* NLEARN- מספר הלמידות
* מספר החלוקות
* בחירות הPREDICTORS
* PRUNE
* משקולות.

יצאנו מנקודת הנחה שבמקרה ההליכה ההבדלים המשמעותיים ביניהם נעוצים בפיצרים לא מוגדרים היטב, לפחות לא על ידינו ולכן ניסינו לחשוב על פיצרים שיכולים לשפוך אור עוד יותר ויותר. מצד אחד לא רצינו להגיע לOVERFITTING באימון, ומצד שני רצינו שיהיה למודל את היכולת הטובה לזהות בין הקבוצות. לאחר כשעתיים התקבלו התוצאות הבאות:

במודל העץ הרגיל:

20.72% שגיאה- עם עוד עשרות מודלים שונים עם אחוזי שגיאה דומים. (bestOptions)

במודל הTREEBAGGER:

**14.75% שגיאה – כלומר 85.25% דיוק!** עם עוד עשרות מודלים עם אחוזי דיוק גדולים מ-80%. (bestOptions2) לצערנו שמרנו את הכל מבלבד את המודל עצמו. לאחר מכן ניסינו להריץ שוב והגענו למודל של **17.85% שהם 82.15% דיוק.** המודל נמצא בקובץ "model8215.mat”

נ.ב.

ברור לנו שיכול להיות שיש כאן מזל וחלוקה מאוד ספיציפית שמקרה הובילה לאחוזים יפים במודל מסויים. יחד עם זאת, בדקנו את אותו מודל על קבוצות ולידציה משתנות והאחוזים תמיד נעים בין 70-82% דיוק. כך שלדעתנו המודל הזה הוא מודל טוב.

אופציה נוספת שהתחלנו לנסות, ומפאת המבחנים ומשימות נוספות לא סיימנו, היא לבדוק את הדאטה על כמה מודלים שונים עם אחוזי דיוק גבוהים ולקחת את התשובה שחזרה על עצמה הכי הרבה פעמים. עבור שימוש ב-3-5 מודלים הכי טובים התקרבנו לאיזור ה-80 אחוז דיוק. לדעתנו אפשר לפתח את זה עוד יותר.