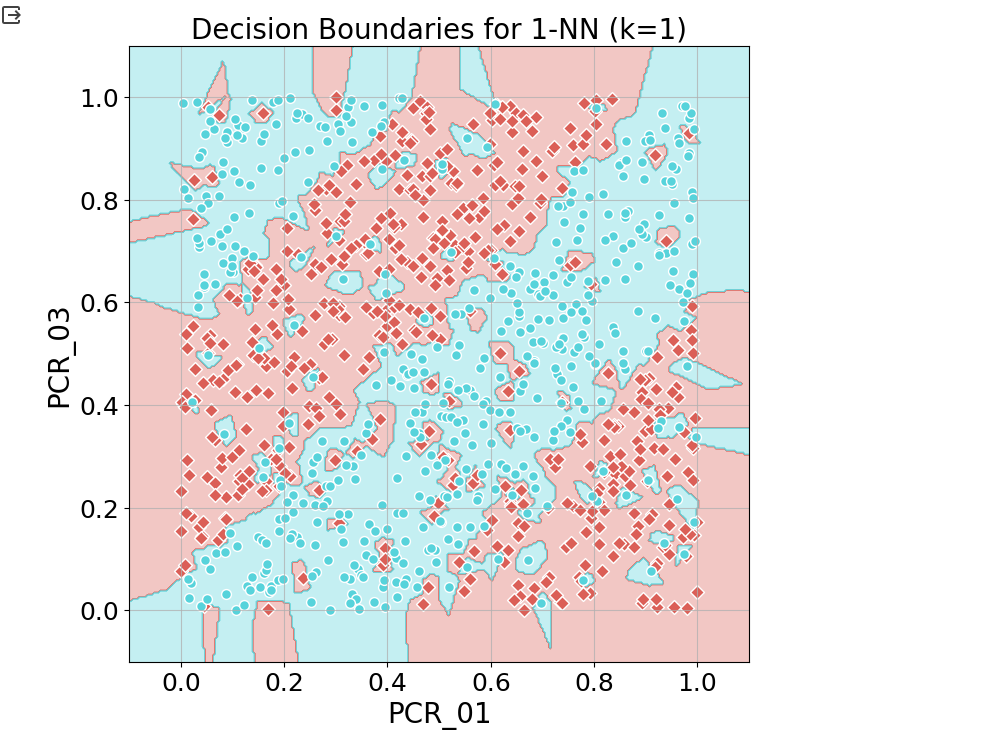
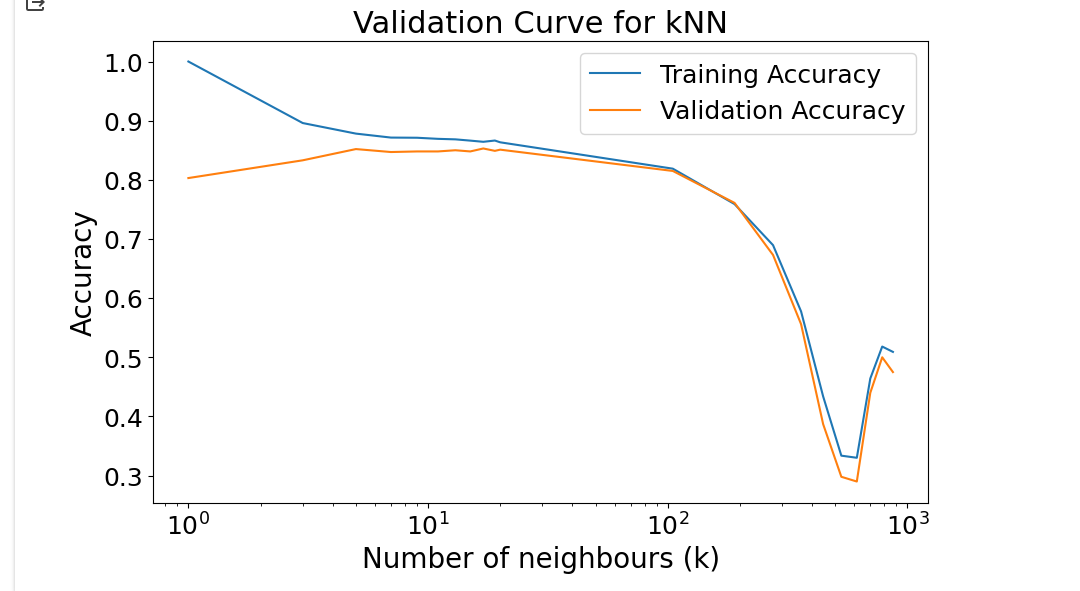
**HW2-Wet**

(Q1)

(Q2)

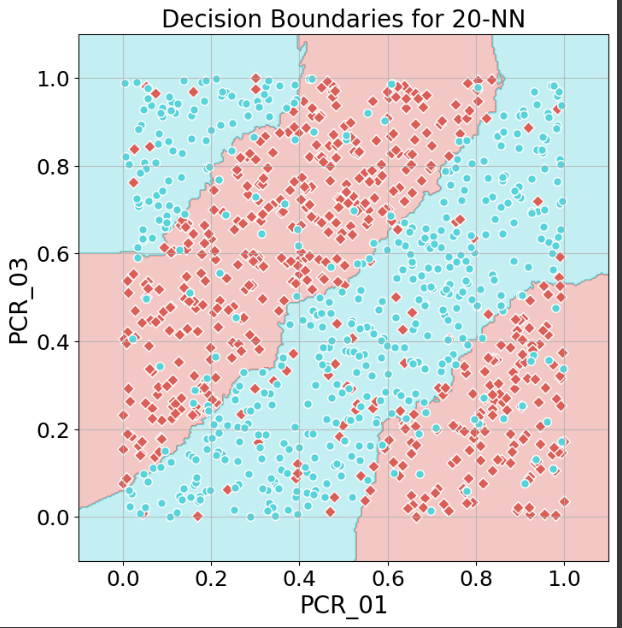


הk הטוב ביותר הוא k=17   
עם מידת דיוק ממוצעת על קבוצת האימון של 0.864 ומידת דיוק ממוצעת על קבוצת הולידציה ממוצעת של 0.853.

ערכי k נמוכים גורמים לתופעת הoverfitting. ניתן לראות כי עבור ערכי k נמוכים מידת הדיוק על קבוצת האימון היא גבוהה ומידת הדיוק על קבוצת הוולידציה נמוכה יותר(כפי שראינו בתרגול ההפרש הזה קשור לoverfit). תופעה זו קורית משום שעבור ערכי k נמוכים, אלגוריתם הknn מחשב את התיוג של נקודה מסוימת לפי מספר שכנים קטן. כתוצאה מכך, האלגוריתם רגיש לנקודות רעש, וניתן לראות גם בplot הקודם שנקודות רעש יוצרות מסביבן אזורי החלטה קטנים שיחזירו תיוג שגוי.

ערכי k גבוהים גורמים לתופעת הunderfitting. ניתן לראות כי עבור ערכי k נמוכים הן מידת הדיוק על קבוצת האימון והן ומידת הדיוק על קבוצת הוולידציה נמוכות. תופעה זו קורית משום שבמצב זה התיוג של נקודה מסוימת מושפע מיותר מדי שכנים, גם משכנים שרחוקים מדי. כתוצאה מכך, ההשפעה של הנקודות הרלוונטיות (הקרובות יותר) בחישוב התיוג יורדת ומקבלים תוצאות שגויות יותר בהרבה מקרים. ראינו בתרגול שאם השגיאה על קבוצת האימון גדולה יחסית, התופעה מקושרת לunderfit ולכך שהמודל שמקבלים הוא פשוט מדי ולא מצליח ללכוד את הקשרים שקיימים בקבוצת האימון.

(Q3)



A black background with white text

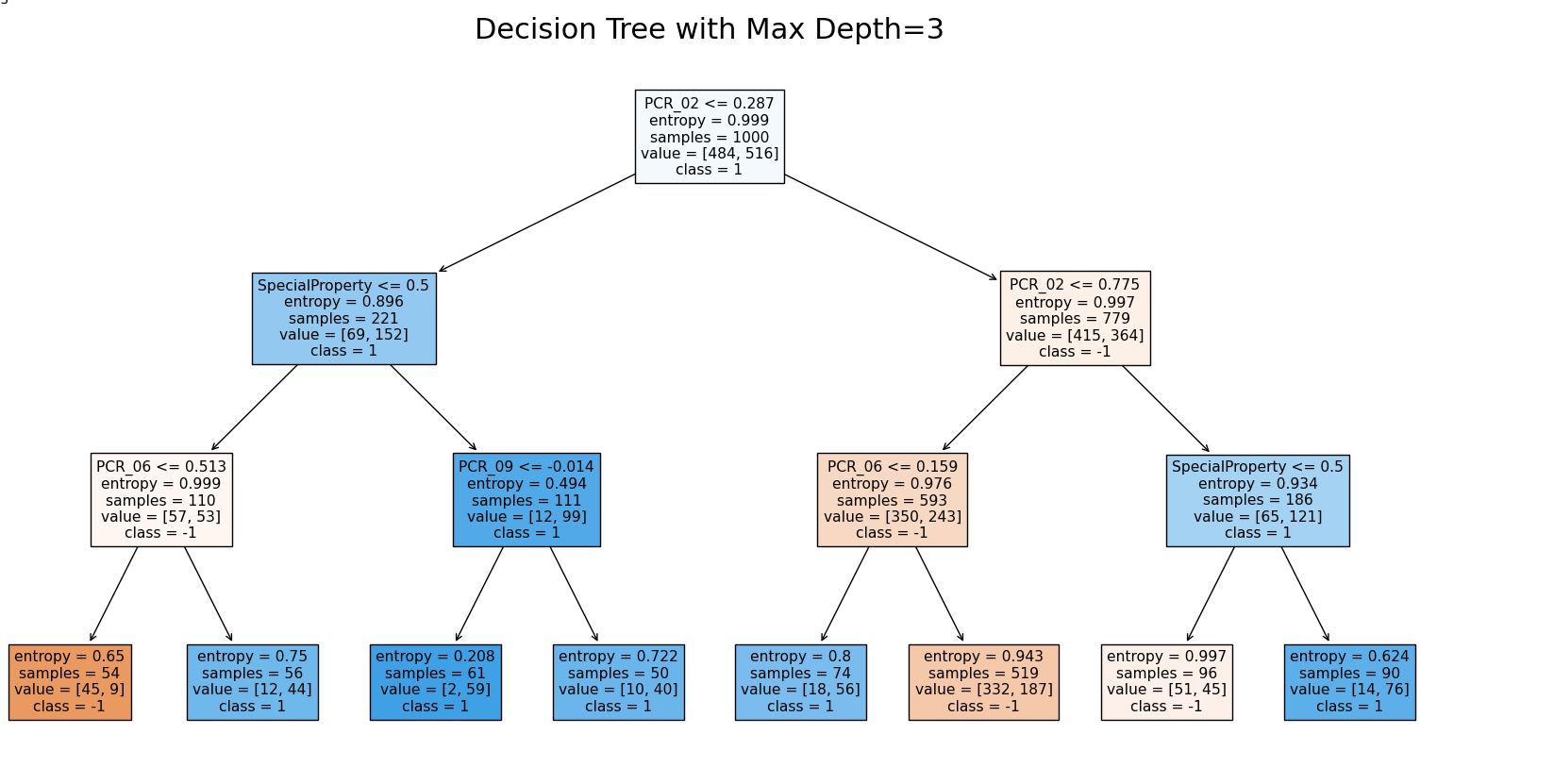
Description automatically generated

(Q4)

גבולות ההחלטה של k=1 מתאימים באופן מושלם לקבוצת האימון מה שמרמז על overfitting ושונות גבוהה יותר במידה והיינו מאמנים מסד נתונים אחר . גבולות ההחלטה של k=17 ״חלקים״ יותר ומצליחים לתפוס את ארבעת אזורי ההחלטה העיקריים של הדאטה , מה שמרמז על יכולת הכללה טובה יותר איזון טוב יותר בין variance לbias.

(Q5)

מידת הדיוק של עץ בעומק 3 על קבוצת האימון היא 0.703



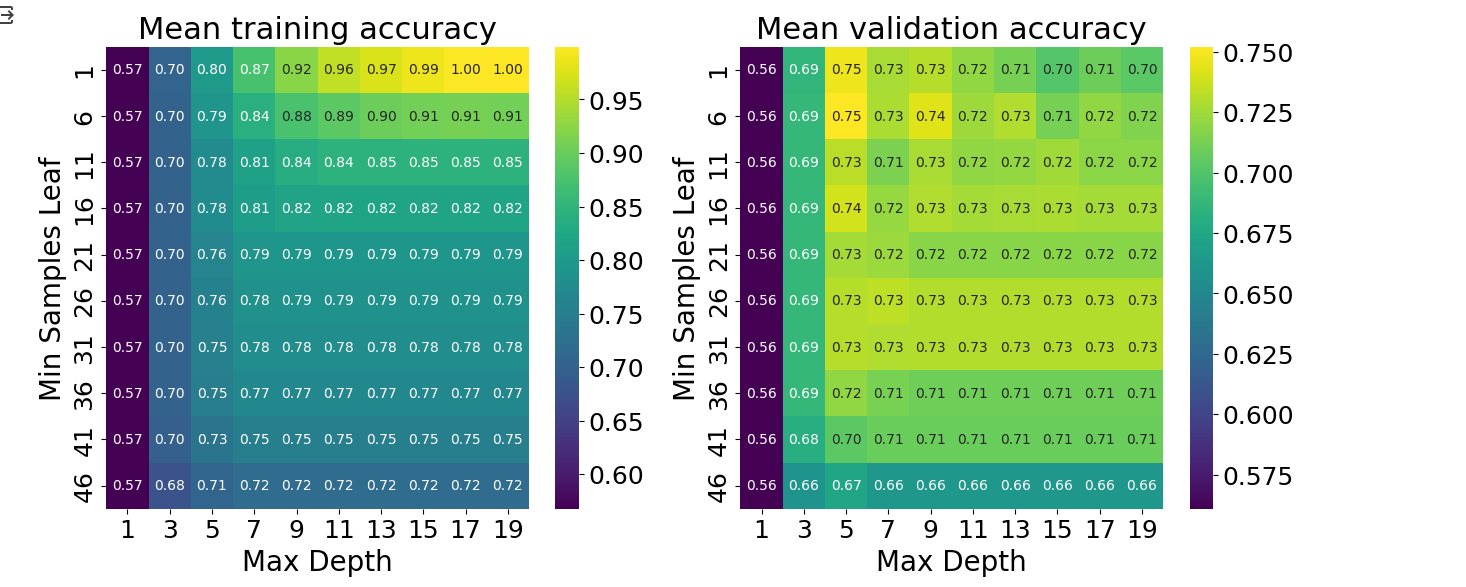
(Q6)

לאחר מספר ניסיונות, החלטנו לבצע grid search בטווחים:

- גובה העץ בטווח של בין 1-20, עם קפיצות של 2 בין כל שני ערכים

- מספר דגימות בעלה בטווח של בין 1-50 עם קפיצות של 5 בין כל שני ערכים

להלן הheap maps המציגות את הgrid search:



* קומבינציה אופטימלית: גובה עץ 5, מס׳ עלים מינימלי בעלה 6, עם מידת דיוק על קבוצת הולידציה של כ0.752.
* קומבינציה שגורמת לunder fitting-

גובה עץ 1 וגם מס׳ דגימות בעלה 46. ניתן לראות שמידת הדיוק על קבוצת האימון נמוכה וגם מידת הדיוק על קבוצת הוולידציה נמוכה, מה שמאפיין את תופעת הunderfitting.

סיבה לתופעה זו: גובה עץ 1 לא מאפשר לתפוס את הקשרים המורכבים בדאטה- ניתן לחלק את הדאטה לכל היותר ל2 קבוצות. במצב כזה המודל פשוט מדי ולא מתאים לדאטה הנלמד.

באופן דומה, מס׳ דגימות גבוה בעלה עוצר את תהליך יצירת העץ, גם כאשר האנטרופיה בעלה מסוים יחסית גבוהה עדין. כתוצאה מכך, לא המודל לא מצליח ללמוד קשרים עדינים יותר בדאטה, שמאפיינים קבוצות קטנות של דגימות.

* קומבינציה שגורמת לover fitting:

גובה עץ 19 ומס׳ דגימות בעלה 1. ניתן לראות שמידת הדיוק על קבוצת האימון גבוהה מאד (מושלמת) ואילו מידת הדיוק על קבוצת הוולידציה נמוכה בהרבה, מה שמאפיין את תופעת הoverfitting.

סיבה לתופעה זו: גובה עץ גדול מאפשר לחלק את הדאטה לקבוצות יותר מדי קטנות (למשל עבור גובה עץ 19 ניתן לחלק את הדאטה ל קבוצות. כתוצאה מכך, יש רגישות -יתר לרעש בדאטה והתאמה יותר מדי מושלמת לדאטה סט אחד , כך שיכולת ההכללה כבר נפגעת.

באופן דומה, מס׳ דגימות קטן בעלה, מאפשר למודל לייצר קבוצות אפילו עבור דגימות בודדות, שעשויות להיות רעש, ולכן גם במקרה נוצרת התאמת- יתר ופגיעה נוספת ביכולת ההכללה.

(Q7)

במהלך הgrid-search בחנו 10 ערכים לכל היפר-פרמטר ולכן סה״כ בחנו 100 קומבינציות.

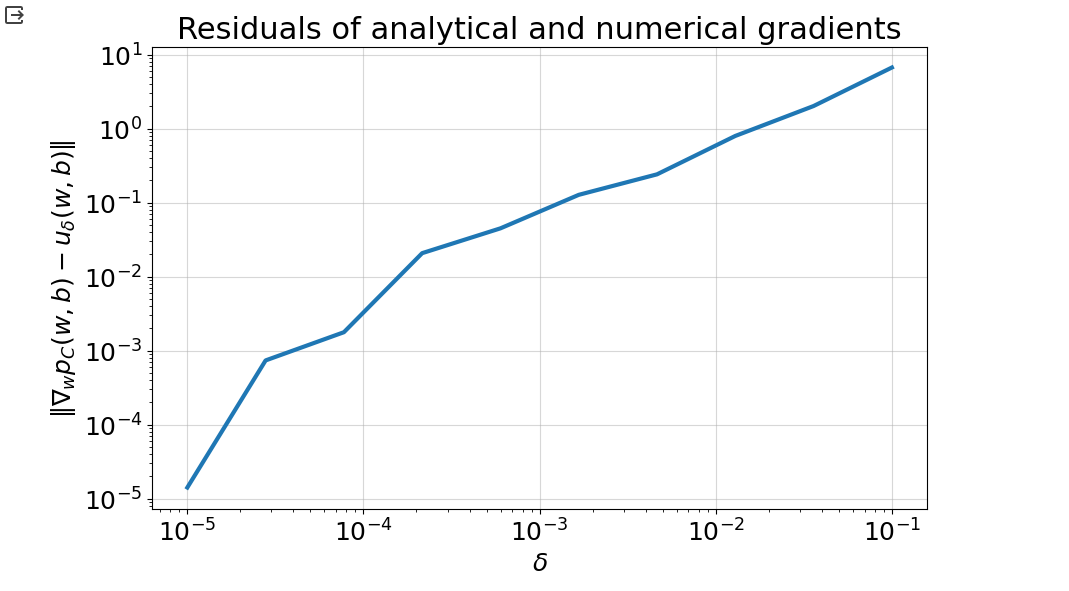
במידה והיינו מכווננות פרמטר נוסף, מספר הקומבינציות היה גדל פי מס׳ הערכים שהיינו בוחנות עבור פרמטר זה. למשל אם היינו בוחנות 10 ערכים אפשריים עבור הפרמטר השלישי , היינו מקבלות 1000 קומבינציות.

באופן כללי, בכל פעם שמוסיפים היפר -פרמטר נוסף למרחב החיפוש , מס׳ הקומבינציות שאנו בוחנים גדל באופן מעריכי (עם כל היפר-פרמטר מכפילים את מס׳ הקומבינציות במס׳ האפשרויות עבור ההיפר פרמטר החדש).

(Q8)

אימנו עץ החלטה עם max\_depth = 5 וmin\_samples\_leaf=6 וקיבלנו מידת דיוק על קבוצת הטסט של 0.76.

(Q9)



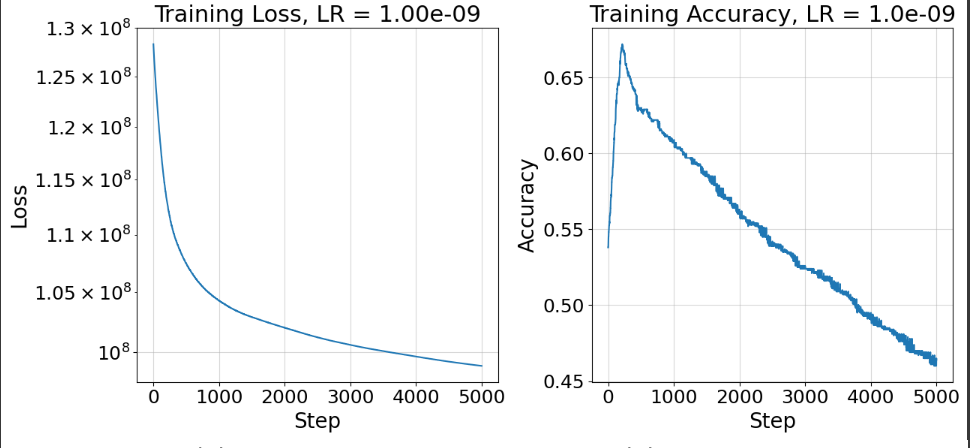
ככל שדלתא קטן ניתן לראות שגם ההפרשים קטנים ומתכנסים ל0

לא בטוחה מה עוד לכתוב

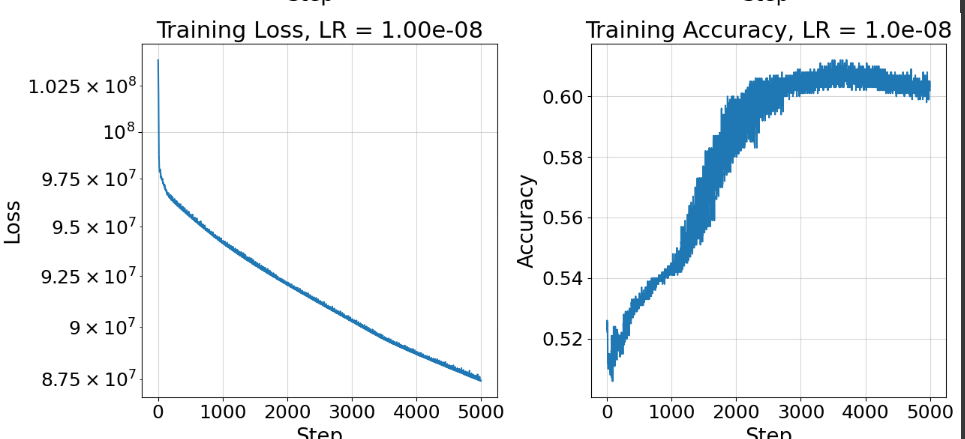
(Q10)

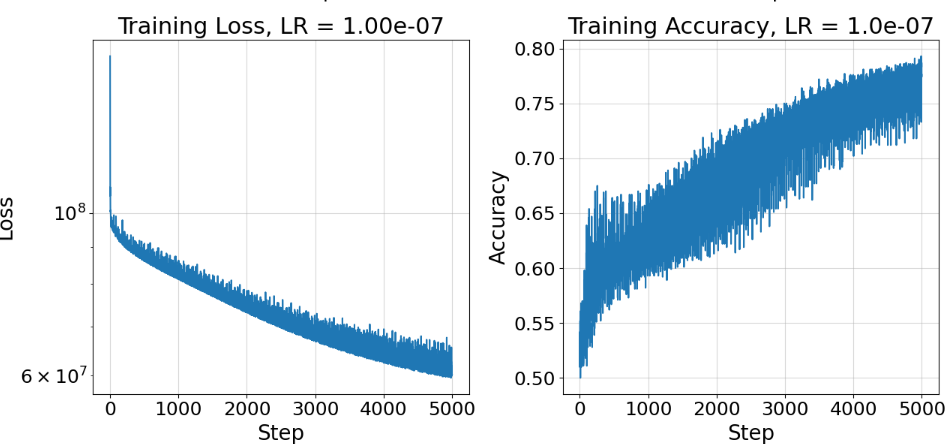
דברים שצריך לשים לב:

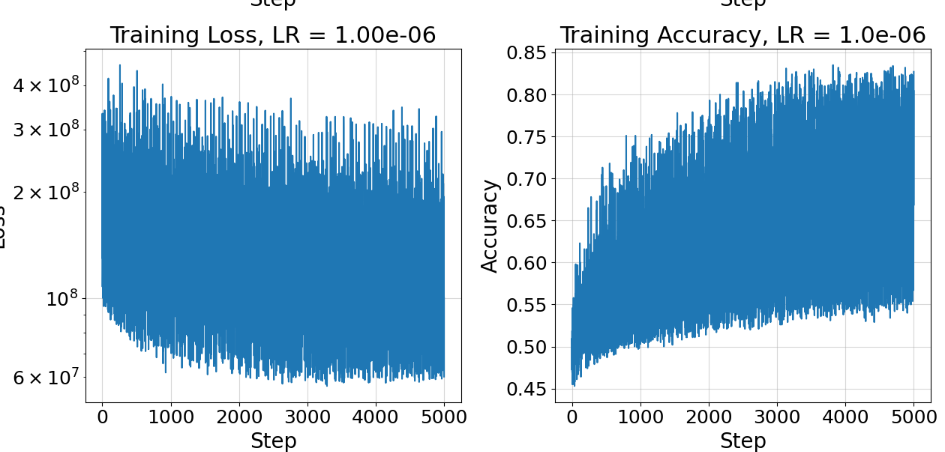
* הדאטה לא פריד לינארית, וזה נראה כאילו למפריד לינארי יהיה מיטיבי לא יהיה אחוזי הצלחה מספקים.
* c גדול מאד – יש עונשים כבדים על טעויות של מיסקלפיקציה , וזה גם גורם לזה שהערכים של פונק׳ המטרה מאד גדולים. בכיתה למדנו שכשC גדול מאד, אנחנו מתכנסים לhard svm , אבל במקרה פה שהדאטה לא פריד זה מוביל אותנו לזה שאין פתרון כ״כ
* קצב הלמידה קטן מאד ולכן ההתכנסות איטית
* למה המידת הדיוק על האימון נראית ככה אני לא בטוחה...
* באופן עקרוני הtrain loss אמור לרדת יחסית במהלך האימון והaccuracy אמור לעלות
* אני גם לא בטוחה איך לקשר את התמונה של המסווג למה היה מצופה? זה לא נראה שבכלל קיבלנו את ה margin הכי טוב שאפשר כי פחות שמנו דגש על זה
* יש פה גם קשר לbatch size
* מינימיזציה על hinge loss בכלל ולא על accuracy

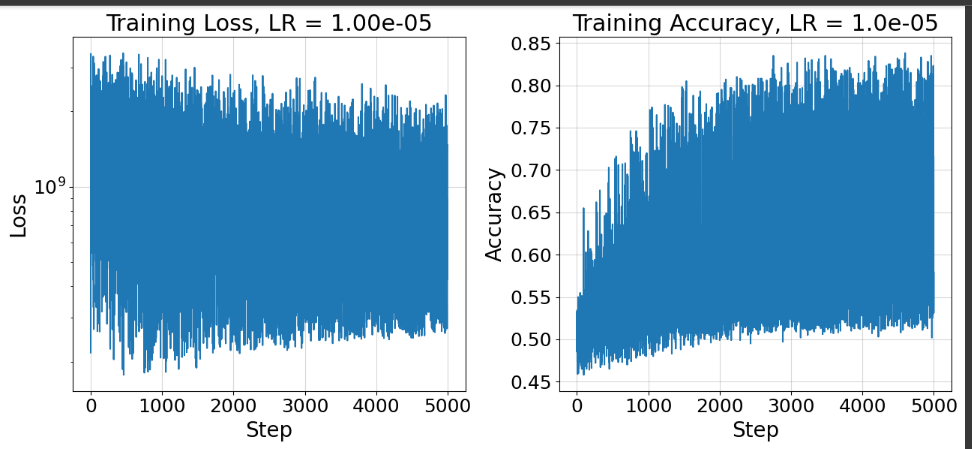


(Q11)







****

**על בסיס הplots בחרנו בlearning rate של 10 בחזקת -7.**

**ראשית כל, עבור LR=10^-5 ו LR=10^6-, אין התכנסות בכלל של פונקציית הloss ,יש התבדרות לגמרי ולכן גודל הlr גדול מדי.**

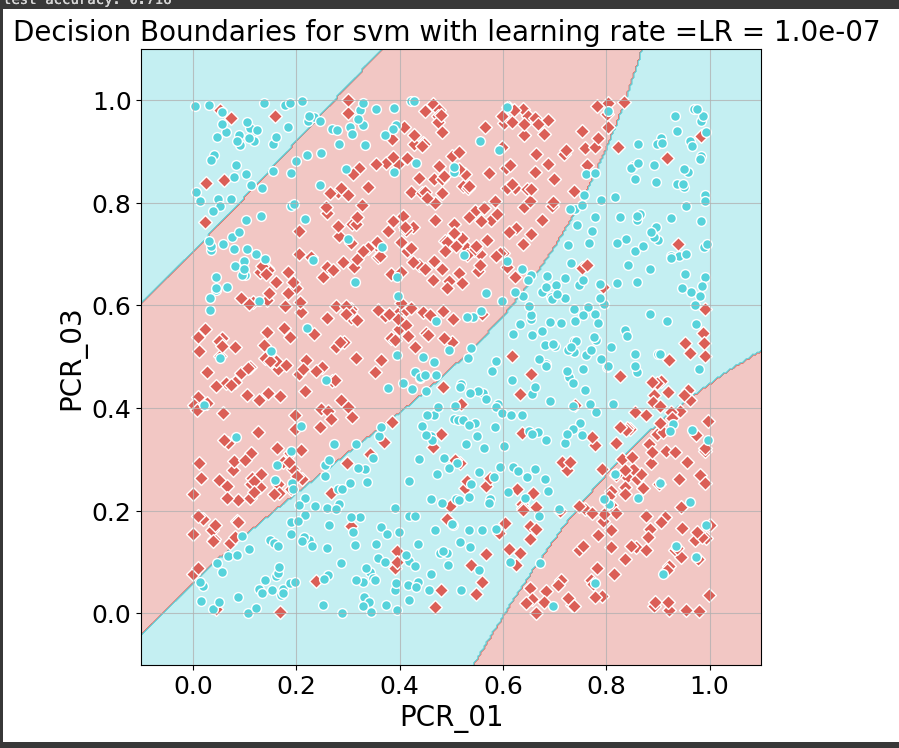
**עבור lr=10^-9 נראה שהוא קטן מדי (כן יש ירידה בפונ׳ הloss אבל בהשוואה לערך הפונקציה עבור קצבי למידה אחרים, הירידה איטית . בנוסף אין שיפור בaccuracy (למרות שאין התחייבות שזה יקרה כי אנחנו עושים מינימיזציה לhinge)**

**ההתלבטות היא בין קצבים lr=10^-8 וlr=10-^-9 , כי נראה שערך המינימום בפונ׳ הloss יורד בקצב מספק ויש שיפור בaccuracy עם תהליך הלימוד.**

**החלטנו לבחור בסופו של דבר ב10^-7 , על אף שהוא רועש יותר, מאחר ונראה כי הוא מביא את הloss לערך המינימום מהר יותר ב5000 צעדים, ובסופו של דבר הaccuracy מגיע סביב 0.7-0.8 (על אף שזה מעט רועש, אבל בסופו של דבר גם זה תלוי בגודל הbatch שעשינו, וזה כן נשאר בטווח הערכים הזה ולא מתבדר לגמרי).**

**12.**

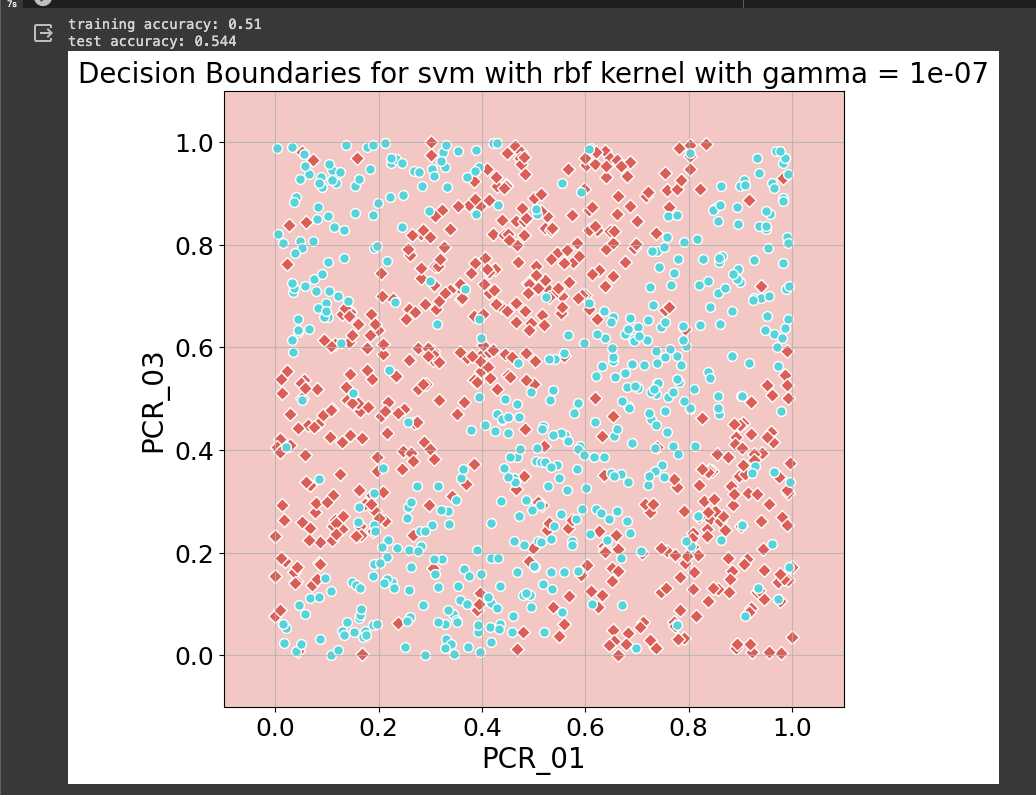
****

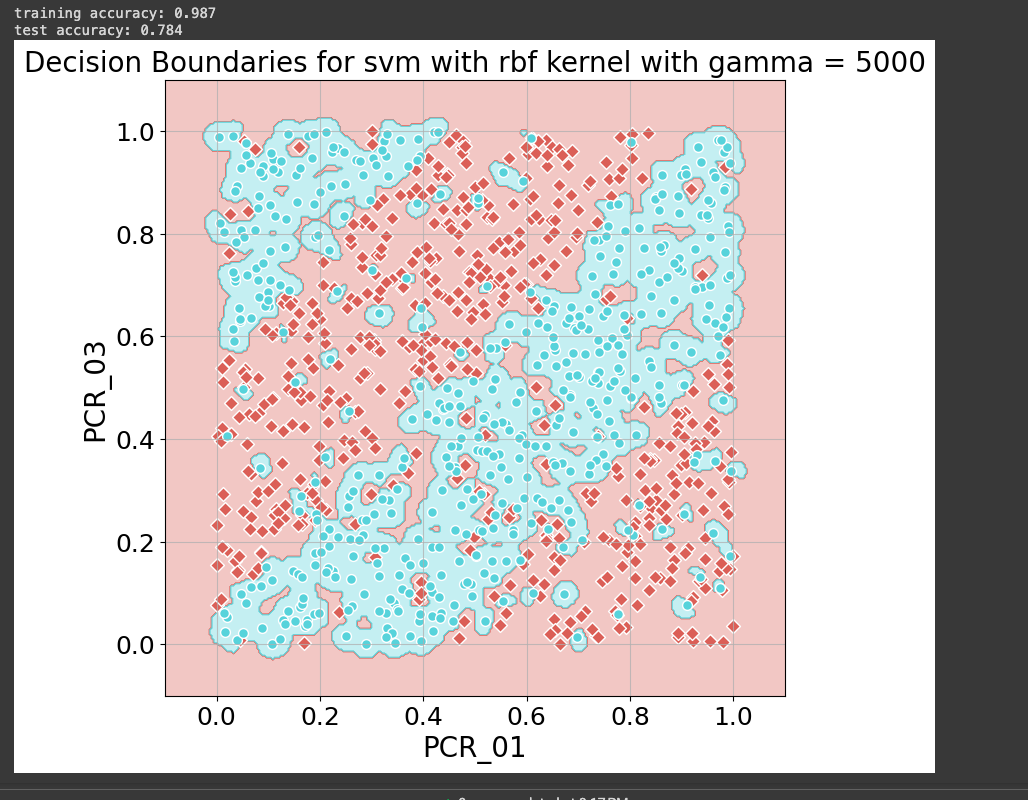
****

1. **.**
2. **להשלים**

**ב.** הוכחנו קודם כי כאשר אנחנו מקבלים שכלל הפרדיקציה הוא ובמקרה שלנו כלל ההחלטה הוא - מאחר שאין דוגמאות בקבוצת האימון שעליהם מקדמי האלפא האפס, הפרדיקציה של כל נקודה כלומר נקבעת על פי תיוג של רוב הנקודות בדאטה סט. נשים לב, שככל שגמא שואף ל0 כמעט ואין חשיבות למרחק של הנקודה x משאר הדוגמאות ולכן ככל שגמא קטן לכיוון האפס אנחנו מקבלים מודל שמחזיר פרדיקציה קבועה (-1 או 1 על פי תיוג הרוב שיש בדאטה סט). אפשר גם להגיד שככל שגמא שואף ל0 אנחנו מקבלים מודל knn שההחלטה שלו מתקבלת על פי m שכנים (m דוגמאות בקבוצת האימון), כאשר שככל שגמא קטן ככה ״מס השכנים״ מתקרב לm.

14. גבולות ההחלטה של המודל אכן מתאימים לכלל ההחלטה הנידון בסעיף 13 ב, כפי שניתן לראות קיבלנו את המסווג ה״טריוואלי״ שפולט פרידקציה קבועה לכל הנקודות.

****

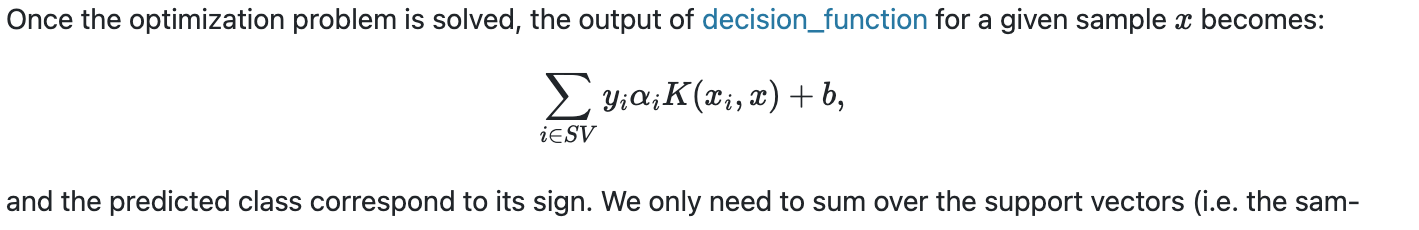
**15.**

נשים לב כי המודל שקיבלנו דומה למודל 1-nn בכך שהוא מתאים את עצמו לקבוצת האימון בצורה מושלמת (יוצר סביבות קטנות סביב נק׳ חריגות). על פי כלל הפרידקציה, מאחר וגמא גדול מאד, נקבל שההשפעה של נק׳ support vector מקבוצת האימון תבוא לידי ביטוי רק במידה והנקודה שרוצים לחזות את התיוג שלה קרובה אליה מאד. לכן, במובן הזה, המודל אמור להיות דומה מאד למודל knn.

עם זאת, ניתן לראות כי לעומת מודל הknn, המודל הזה נותן פרדיקציה קבועה (במקרה שלנו הצבע האדום מייצג פרדיקציה -1) לרוב הנקודות, ויודע לתת פרדיקציה 1 רק שהנקודה נמצאת ממש קרוב לנקודה עם תיוג 1 . כלומר המודל לא יודע לתת פרדיקציה 1 לנקודה שהשכן הקרוב שלהן הוא כחול כאשר המרחק בין הנקודות לא מספיק קרוב.

אנחנו חושבות שתופעה זו קורית, מאחר וכאשר גם גמא גדול וגם המרחק גדול נקבל כי הוא כמעט 0, ומבחינה נומרית -> המחשב עשוי להתייחס אליו כאפס.

במצב כזה, כאשר לנקודה אין אף שכן מספיק קרוב מבחינה נומרית, הפרידקציה שלה נקבעת לפי גורם הb. כפי שניתן לראות כלל החלטה של המודל שאימנו לפי הדוקומנטציה של sklearn הוא:





במקרה שלנו הדפסנו וראינו כי :

ולכן הגיוני שהחיזוי שלנו במקרים של נקודות ללא שכנים מספיק קרובים יוצא שלילי (שכן החיזוי נקבע ע״פ הסימן של b שהוא שלילי.