



למידה ייצוג וראיה ממוחשבת – מטלה 1



מגישים:

אורן גולדפישר 307967612

חן גרינברג 204640262

תאריך:

28/12/2020

תוכן עניינים

3.....	שיטת כיוון הפרמטרים
4.....	פרמטרים
5.....	גרפים
7.....	המודל הנבחר וערכי הפרמטרים הנבחרים
8.....	טעויות מקסימליות בכל קטגוריה

שיטת כיוון הפרמטרים

למרות שדרך זו צורכת יותר זמן חישוב, בחרנו בכל זאת להתנסות בשיטה זו ולהגיע לפרמטרים האופטימליים בעזרת קרוס וולידציה.

קרוס הוולידציה הופעלה בשלב האחרון בתהליך. הגדרנו סט של פרמטרים אשר אותם בחרנו לאפטם וחילקנו אותם לשני חלקים:

1. טעינת התמונה (כולל בחירת גודל התמונה) + חילוף פיצ'רים על ידי HOG

2. החלק של המודל (SVM)

הגדרנו רשימה של ערכים עבור כל פרמטר בחלק הראשון, לאחר מכן הרצנו בלולאה מכוננת את הפרמטרים מהחלק הראשון בצורה כזאת שהם יוצרים קומבינציה ייחודית בניהם בכל איטרציה.

קומבינציה ייחודית היא סט ערכים של הפרמטרים שבחרנו לאפטם בחלק הראשון (כל איטרציה בתוך הלולאה יוצרת לנו קומבינציה חדשה). ברגע שיש לנו קומבינציה, אנחנו טוענים את התמונות, גוזרים אותן בגודל של פרמטר גודל התמונה ולאחר מכן מפעילים תהליך של חילוף פרמטרים על ידי HOG. בסוף השלב הזה, נוצר לנו הדאטה סט.

על הדאטה סט הזה אנחנו מפעילים שיטת קרוס ולידציה על מנת למצוא את הפרמטרים האופטימליים עבור אלגוריתם ה-SVM (החלק השני).

את התוצאות של הקרוס ולידציה שמרנו בתוך מילון, כאשר המפתח הינו הקומבינציה הייחודית של ההיפר פרמטרים והערך הוא אובייקט הקרוס ולידציה עם התוצאות עבור אותם היפר פרמטרים שנמצאים במפתח.

הדרך בה בחרנו את הערכים האופטימליים עבור כל פרמטר בשלב הראשון, היא מציאת הקומבינציה הטובה ביותר לפי ערכי הקרוס ולידציה. את הקומבינציה הזו הגדרנו כקומבינציה הנבחרת.

בעבור כל פרמטר שבדקנו, שמרנו את כל שאר הערכים להיות קבועים לפי הקומבינציה האופטימלית ובכל פעם שינינו רק את הערך של המשתנה שאותו אנחנו רוצים לאפטם. זאת על מנת לוודא שרק השינוי של הפרמטר שאותו אנחנו מאפטמים ישפיע על תוצאת המודל.

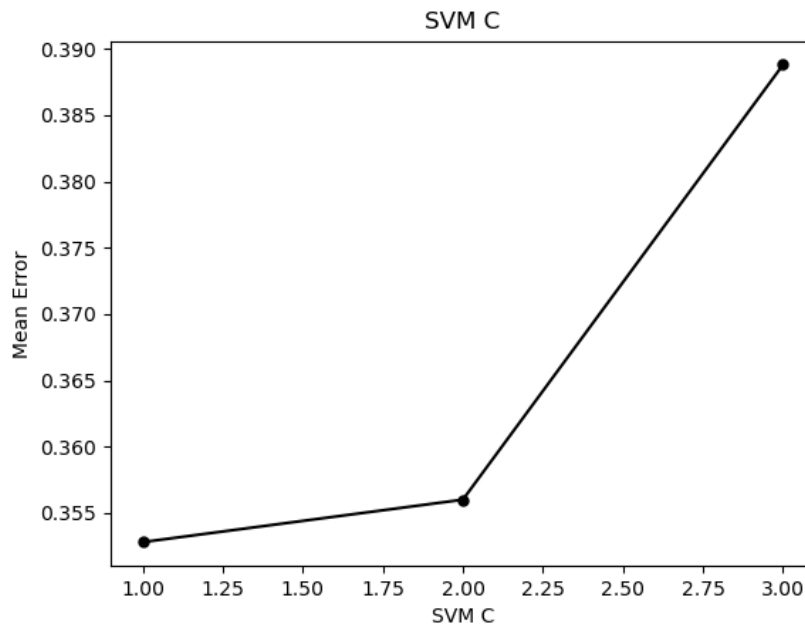
פרמטרים

1. C – הקנס על טעות במודל ה-SVM "קבוע ענישה". ערך C גבוה פירושו העדפת הסיווג הנכון על פני שוליים רחבים, וערך C נמוך מעדיף הכללה (שוליים רחבים), גם במחיר שדוגמאות האימון, אולי לא יסווגו נכונה.
2. Kernel – ברירת המחדל של המודל היא יצירת מפריד לינארי, אך לא תמיד זה המפריד המתאים ביותר למודל. נוכל לעבור למרחב ייצוג אחר בעזרת מכפלה פנימית, אשר תביע במימד גבוה יותר בעיה אשר שקולה לבעיה המקורית. היפר פרמטר זה מייצג את המודל לפיו ה-SVM יסווג את התמונות.
3. (S) Picture Size – אנו קולטים בתחילת התוכנית 10 קבוצות של תמונות כדי לאמן מודל בעזרתן. כדי שלמודל לא יהיו סטיות בעקבות גדלי תמונה שונים, אנו יוצרים גודל קבוע (אורך ורוחב) לכלל התמונות. היפר פרמטר זה מייצג את האורך והרוחב שהתמונות ישונו אליהם בזמן הכנת הנתונים.
4. HOG Bins – אלגוריתם HOG מקבל תמונה ויוצר היסטוגרמה שלה אשר מיוצגת כווקטור ערכים. על מנת ליצור את ההיסטוגרמה הוא משתמש בבני אוריינטציה בתאים של התמונה. היפר פרמטר זה מייצג את מספר בני האוריינטציה בתא.

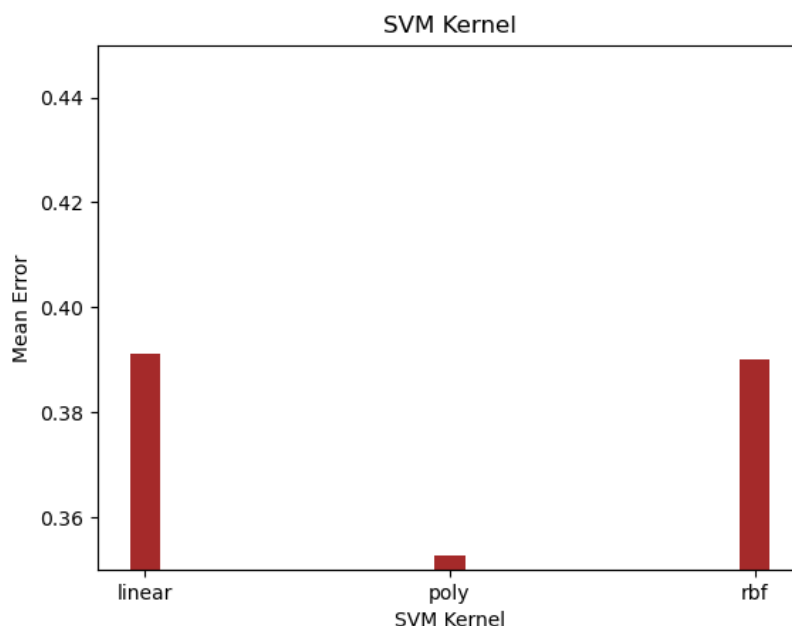


גרפים

1. C – בדקנו 3 ערכי C שונים, 1, 2 ו-3, כפי שניתן לראות, השגיאה המזערית התקבלה בעבור הערך הנמוך ביותר, כלומר המודל שאף להחמיר כמה שפחות עם הערך של C, ולכן הערך הנבחר הוא C שווה ל-1.

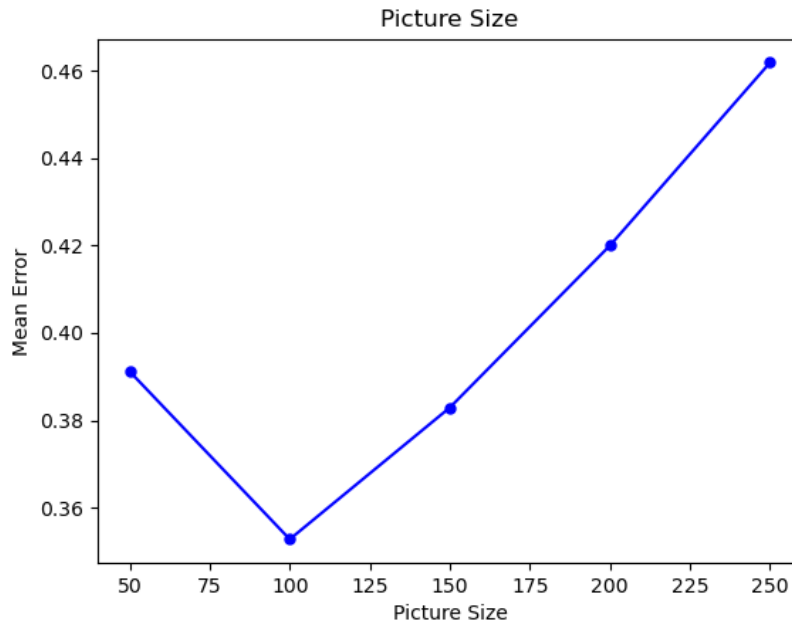


2. Kernel – בדקנו שלושה סוגי kernel: פולינומיאלי, לינארי rbf תוך הצבה של הפרמטרים האופטימליים שהגענו אליהם עד כה. כפי שניתן לראות, kernel שהוביל אותנו לשגיאה הנמוכה ביותר הינו הפולינומיאלי, עם 34% אחוזי שגיאה, לכן, אנחנו נבחר במודל הפולינומיאלי.

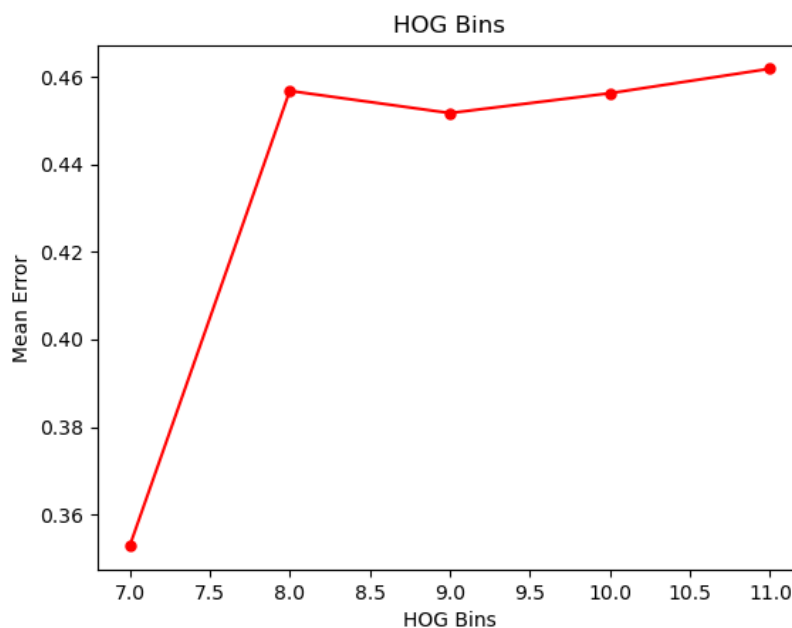




3. Picture Size (S) – ביצענו בדיקה על 5 ערכי S שונים, כאשר ההפרש בניהם היה 50 בכל פעם, כפי שאפשר לראות, אחוז השגיאה המינימלית התקבל כאשר גודל התמונה היה שווה ל100, לכן התמונות שלנו הן בגודל 100x100.



4. HOG Bins – בחנו 5 אופציות עבור מספר ביני האוריינטציה בטווח שבין 7 ל11 כולל, כאשר ההפרש בין הקפיצות הוא 1. ניתן לראות שאחוזי השגיאה המינימליים התקבלו במצב של 7 בינים ולכן נבחר בערך הזה.





המודל הנבחר וערכי הפרמטרים הנבחרים

לאחר האפטום בשיטת הקרוס ולידציה, הגענו להיפר פרמטרים הבאים:

Picture Size (S) - 100

Orientation Bins – 7

Kernel – Poly

C – 1

confusion matrix

```
Error rate for non linear SVM: 0.2772727272727272
[[12  0  1  1  0  2  1  0  0  0]
 [ 0 14  0  0  0  0  3  0  0  0]
 [ 1  0 18  0  4  1  0  0  1  0]
 [ 0  0  0 22  0  1  0  1  1  0]
 [ 0  2  1  0 16  1  4  1  0  0]
 [ 0  0  0  0  0 20  1  0  0  0]
 [ 1  4  0  0  3  0 15  0  2  0]
 [ 1  0  3  0  0  0  1 18  2  0]
 [ 2  2  0  0  2  0  1  4 14  0]
 [ 0  0  2  0  0  2  0  1  0 10]]
```

כפי שניתן לראות, ברגע שהרצנו את המודל הנבחר על התיקיה הטסט, קיבלנו שגיאה ממוצעת של 27.72% אחוז.

עפ"י מטריצת המבוכה, ניתן לראות כמה נתונים מעניינים:

- ניתן לראות כי במחלקת Gerenuk, כל השגיאות של המודל סווגו כ-Hedgehog, כנראה משום שבשני המקרים מדובר בחיות ולעיתים האלגוריתם לא הצליח להבדיל בניהן.
- Hawksbill סווג ב-45% מהטעויות שלו כ-Hedgehog, ככל הנראה צורת השריון דומה לצורה של הגב הקוצני של Hedgehog.
- המחלקה של Headphone עם אחוזי שגיאה הנמוכים ביותר, האלגוריתם הצליח לסווג 20 מתוך 21 התמונות של הטסט, כלומר 5% שגיאה.
- 40% מטעויות הסיווג של Hedgehog סווגו כ-Gerenuk, ככל הנראה מאותה הסיבה ש-Gerenuk סווג כ-Hedgehog כמה בכמה תמונות.
- המחלקה עם אחוזי השגיאה הגבוהים ביותר היא Ibis, עם 44% שגיאה, כאשר המחלקה סווגה בתור כל מיני מחלקות שונות. ב-36% מהשגיאות סווגה כ-Helicopter, כנראה מכיוון שצורתו יחד עם הכנף המסובבת הזכירה את צורת Ibis יחד עם הרגליים הדקות.












תוצאות המודל הלינארי

```
Error rate for linear SVM: 0.34545454545454546
[[14 0 0 1 0 1 1 0 0 0]
 [ 0 13 0 0 0 0 4 0 0 0]
 [ 2 0 13 0 5 1 2 1 0 1]
 [ 0 1 2 21 0 0 1 0 0 0]
 [ 1 2 0 0 15 1 6 0 0 0]
 [ 0 1 1 1 1 13 3 0 0 1]
 [ 0 4 0 0 3 0 15 0 3 0]
 [ 0 1 2 1 5 0 1 14 1 0]
 [ 2 2 1 0 3 0 0 2 15 0]
 [ 0 0 1 0 1 1 0 1 0 11]]
```

טעויות מקסימליות בכל קטגוריה

מחלקה	שגיאה מקסימלית	שגיאה שנייה מקסימלית
41 – Garfield		
42 – Gerenuk		
43 – Gramophone		



		44 – Grand Piano
		45 – Hawksbill
		46 – Headphone
		47 – Hedgehog
		48 – Helicopter



		<p>49 – Ibis</p>
		<p>50 – Inline Skate</p>