

למידה חישובית – פרויקט מסכם

בפרויקט הנ"ל כתבנו תוכנה שמנסה ללמוד את הדטה המצורף וזאת על ידי מספר שלבים המעריכים את איכות הלמידה והאלגוריתמים:

1. השוואה בין אלגוריתמים על ידי כווננם בהתאם לחומר הנלמד בקורס
2. הצגת גרפים מפורטים הממחישים את השפעתם של פרמטרים שונים על הלמידה
3. הצגת גרפים הממחישים את השפעתם של פיצ'רים חשובים וכמותם על הלמידה
4. התמודדות עם overfitting על ידי הרצת אלגוריתם K-Fold cross validation

הערה: בתרגיל השתמשנו הן בספרייה SKLEARN והן באלגוריתמים שמומשו על ידינו במהלך קורס זה.

מהלך הפרויקט

בעת הרצת התוכנית (המקבלת כפרמטר קובץ xls. בלבד) למשתמש תהיה אפשרות לבחור בין מספר פעולות שברצונו לבצע:

1. הרצת השוואה בין אלגוריתמים שונים והצגתם בגרפים מפורטים (השוואה אל מול אחוזי ההצלחה של האלגוריתם)
2. הרצת אלגוריתם K-Fold cross validation להתמודדות עם overfitting
3. הרצת האלגוריתם הטוב ביותר ומתן ציון (בין 0-100) עבור הדטה של המשתמש
4. יציאה מהתוכנה

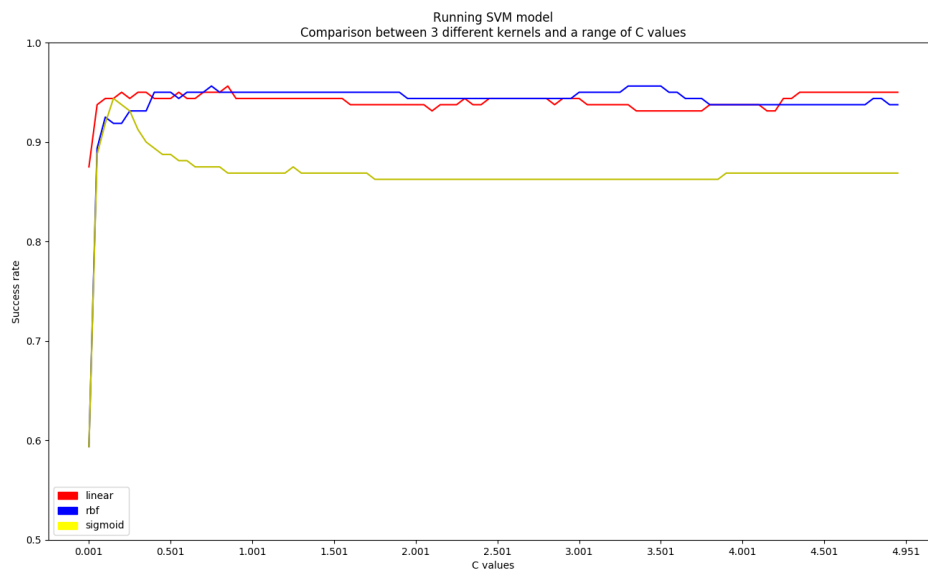
שימו לב, הרצת התוכנית תתבצע דרך ה-CMD באופן הבא:

```
python test.py <Input_xls_file>
```

הרצת השוואה בין אלגוריתמים שונים והצגתם בגרפים מפורטים

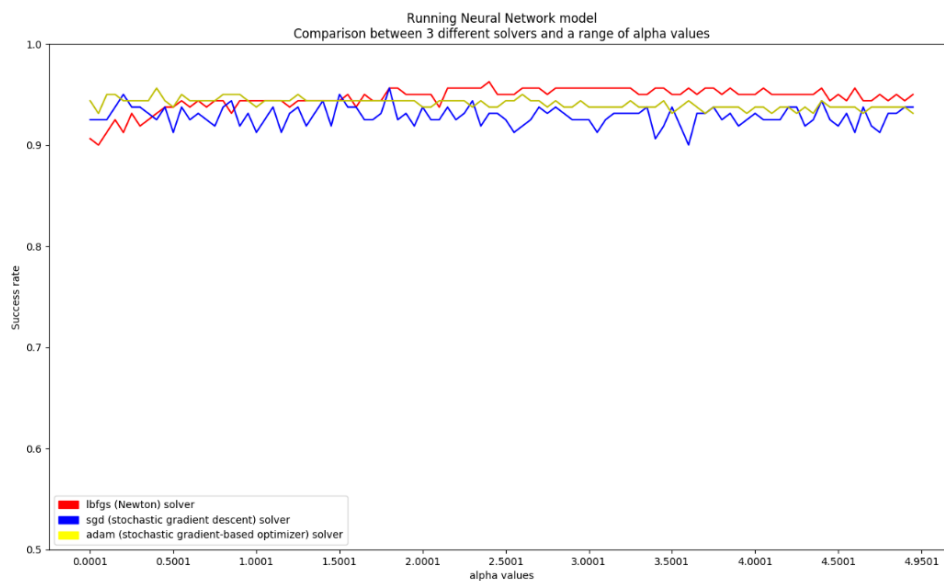
1. SVM (Support Vector Machine)

- ❖ באלגוריתם זה השונו בין הקרנלים הבאים: Linear, Sigmoid, Gaussian (rbf).
- ❖ באלגוריתם זה השונו בין ערכי מקדם התיקון בטווח הבא: בין 0.001 ועד 5 בקפצות של 0.05.



2. Neural Network

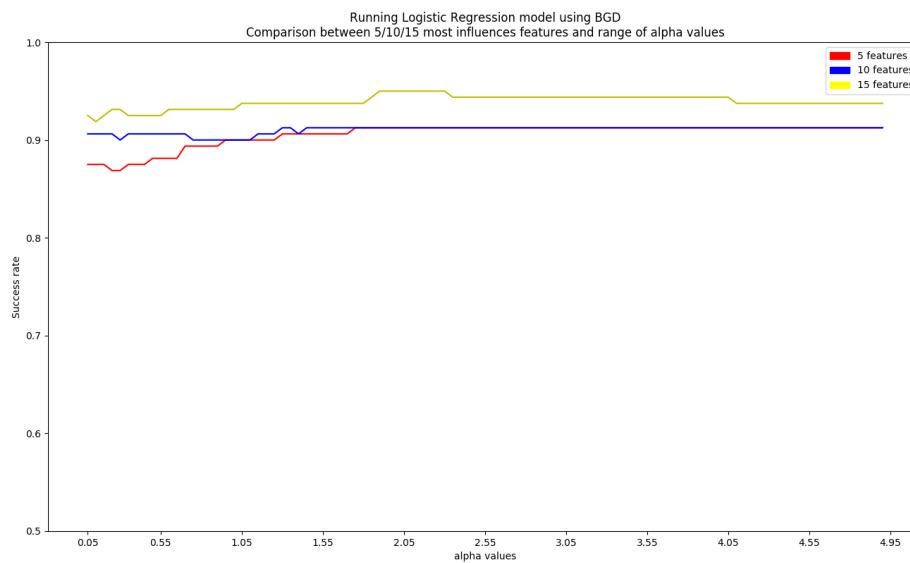
- ❖ באלגוריתם זה השונו בין מספר אלגוריתמים למציאת מינימום בהם רשת הניורונים משתמשת: lbfgs (Newton), sgd (SGD), adam (SGD optimizer).
- ❖ באלגוריתם זה השונו בין ערכי α בטווח הבא: בין 0.0001 ועד 5 בקפצות של 0.05.



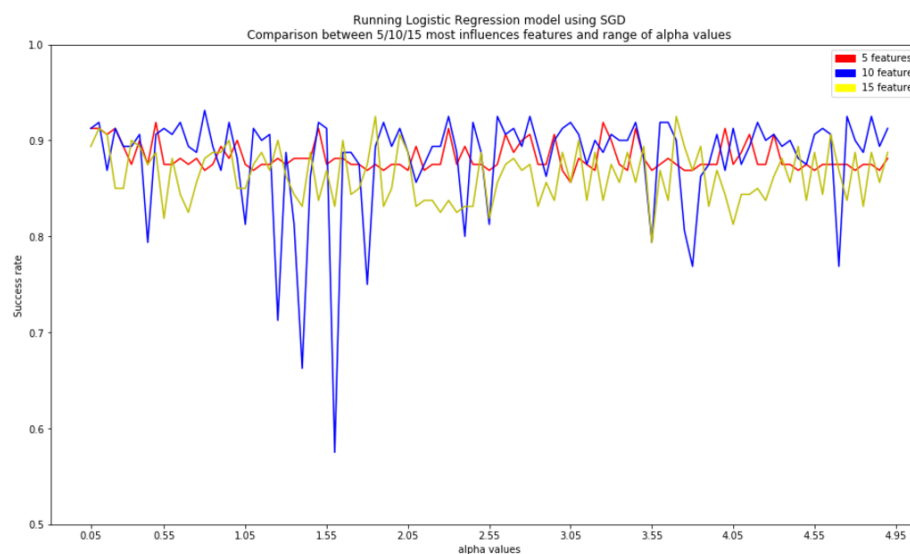
3. Logistic Regression using batch/stochastic gradient descent

- ❖ באלגוריתם זה השונו בין 5, 10, 15 הפיצ'רים המשפיעים ביותר אשר חושבו על ידי Mutual Information.
- ❖ באלגוריתם זה השונו בין ערכי α בטווח הבא: בין 0.05 ועד 5 בקפיצות של 0.05.

Batch Gradient Descent



Stochastic Gradient Descent



הרצת אלגוריתם K-Fold cross validation להתמודדות עם overfitting

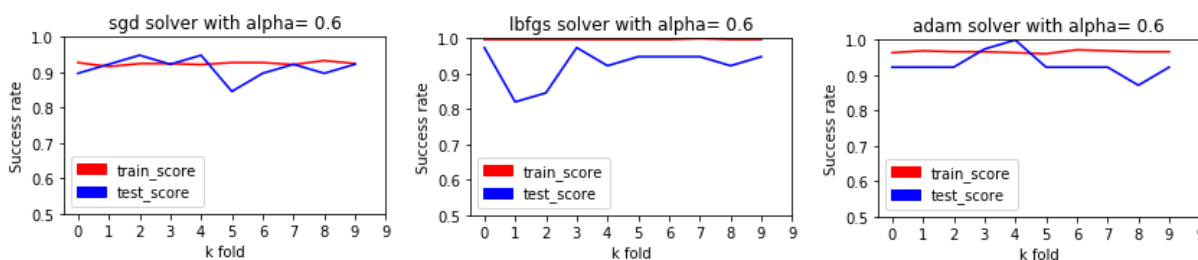
בפרויקט זה הרצנו את אלגוריתם K-Fold cross validation כאשר $K=10$ על מנת לאפשר התמודדות עם overfitting.

הרצנו את האלגוריתם על הדטה הנתון על מנת למצוא את האלגוריתם היציב ביותר עבור 10 הפרמוטציות האפשריות לחלוקת הדטה לtrain ו- test.

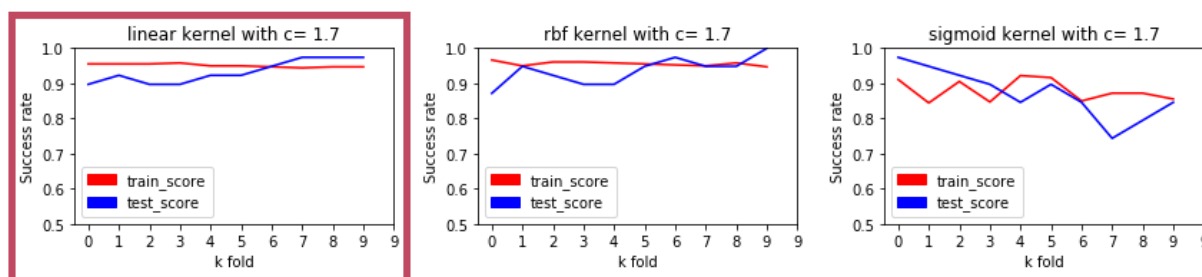
הערות:

- את האלגוריתם הרצנו על 2 האלגוריתמים הבאים: SVM ו- Neural Network עם הפרמטרים שהוצגו בהשוואה בסעיף הקודם.
- כאן נציג מספר מצומצם של גרפים להשוואה (בריצה ניתן לראות את כולם), כאשר הפרמטר להשוואה הוא אחוז ההצלחה של כל אלגוריתם.

1. Neural Network



2. SVM (Support Vector Machine)



המסקנה שלנו מהרצת K-Fold cross validation על האלגוריתמים הנ"ל היא שאלגוריתם SVM עם הפרמטרים הבאים, Linear Kernel & $C=1.7$, הוא היציב ביותר עבור הדטה שקיבלנו.

הרצת האלגוריתם הטוב ביותר ומתן ציון (בין 0-100)

בחירה בפעולה זו תריץ את האלגוריתם שנמצא כיציב ביותר בעזרת אלגוריתם K-Fold cross validation (בסעיף הנ"ל) על הדטה שהשתמש יכניס. השיטה תציג את אחוז הצלחת החיזוי ואת מספרי האינדקסים של הדוגמאות שנפלו.