מדעים דיגיטליים להייטק, אוניברסיטת תל אביב

סיכום פרויקט

מבוא ללמידת מכונה

[redacted]	
גיא דהאן	תמר שנירר

תאריך הגשה
6/19/2021

תקציר

במסגרת הקורס "מבוא ללמידת מכונה", התבקשנו לבצע פרויקט של ניתוח דאטה סט של פרטי הזמנות של עסק בתחום התיירות. מטרת הפרויקט היא יישום נושאי הקורס שלמדנו בהם לרבות מודלים של למידת מכונה, מטריקות למדידת ביצועי מודל ומושגי יסוד אחרים.

הפרויקט כלל חמישה חלקים: אקספלורציה, עיבוד מקדים, הרצת המודלים, הערכת המודלים וביצוע פרדיקציות.

במהלך הפרויקט ביצענו ניתוח מעמיק של הנתונים השונים בסט המידע שניתן לנו (ה-"דאטה סט"), הצגנו את המידע במגוון שיטות של ויזואליזציה וכן הצגנו את הקורלציות השונות בין הפיצ'רים השונים והשפעתם על ביטול או קיום ההזמנות במידע כפי שניתן.

בשלב הבא ביצענו עיבוד מקדים שכלל שיטות שונות להשלמת מידע חסר, עיבוד הנדרש לקבלת תוצאות מהימנות מהמודלים וכדומה.

בחרנו לבצע מימוש והשוואה של ארבעה מודלים, Random Forrest , MLP, LogisticRegression ו-KNN. כל אחד מהמודלים קיבל סקירה מעמיקה לגבי טיב ביצועיו על הדאטה, שכללה ניקוד לפי מספר שיטות, פלט של ROC יחד עם CV ל-Kfold שונים ועוד.

במהלך העבודה ביצענו ניהול של הגרסאות באמצעות Git ועקבנו אחרי מספר גרסאות בהן נבחנו ביצועי המודלים השונים על מספר גרסאות של עיבוד הדאטה – ביצוע PCA או לא, סטנדרטיזציה, קידוד לפי שיטות שונות של נתונים קטגוריאליים וכן התמודדות עם נתונים חסרים. השיטות כפי שהן מוצגות בפרויקט המסכם מייצגות את הממצאים שלנו לביצועים הטובים ביותר, ואך ורק הגרסאות הטובות ביותר הושארו עבור כל מודל.

בחרנו לעבוד בתצורה של Pipelines המובנים בחבילת Sklearn. זה אפשר לנו לשנות ולהתאים את הליך הבדיקה שלנו בשינוי של מספר שורות בודדות בלבד והקל על התהליך כולו.

חלק 1: אקספלורציה

בשלב הראשון של הפרויקט היה עלינו לבצע "היכרות" עם הנתונים. אחד האתגרים העיקריים באקספלורציה היה להבין את המהות של כל פיצ'ר ובפרט של הפיצ'רים האנונימיים. בהיעדר הבנת מהותם, הייתה חסרה לנו האינטואיציה לגבי האם ההתפלגות הערכים והשונות שלהם סבירים, ויתרה מכך האם הם מסוג משתנים קטגוריים או מספריים.

בתחילה השתמשנו בפקודות המתארות את הנתונים. לדוגמא: הפקודה describe נותנת לנו מידע סטטיסטי על כל פיצ'ר לרבות הממוצע, סטיית התקן, רבעונים וכדומה.

לצורך האקספלורציה, איחדנו באופן זמני את סט ה-label שניתן לנו יחד עם הדאטה, כלומר את העמודה שקובעת האם ההזמנה בוטלה או לא, זאת על מנת לייצר ויזואליזציות של פיצ'רים עם המידע לגבי הביטולים.

לאחר מכן, השתמשנו בספריות matplotlib ו- seaborn על מנת לבצע ויזואליזציות שונות לנתונים כגון היסטוגרמות של פיצ'רים, מטריצות קורלציה, דיאגרמת עוגה ודיאגרמות עמודות.

היה לנו חשוב עוד בטרם שלב הוויזואליזציה להמיר את הייצוג הטקסטואלי של פיצ'רים מסוימים לייצוג מספרי כגון: order_month, order_week, זאת לצורך עיבוד מהימן של הנתונים.

בשלב הבא של הניתוח, יצרנו פיצ'רים חדשים כדי ליצור ויזואליזציה יותר כוללנית, לדוגמה - משתנה המתאר את העונה בשנה שנוצר מתוך החודש.

עוד משתנה שיצרנו הוא משתנה בינארי המתאר האם ישנם ילדים בהזמנה, זאת מתוך התובנה שרוב ההזמנות אינן כוללות ילדים, ואלה שכן - הן בעלות ערכים קרובים.

ניצלנו את השלב הזה גם על מנת לבחון השערות ותהיות שונות שהיו לנו. למשל: האם יש קשר בין מספר ההזמנות עם ילדים לעונה בשנה, כיצד מתפלגת כמות ההזמנות של כל מדינה על גבי עונות השנה, כיצד מתפלג סוג ההזמנה, וכדומה.

חלק 2: עיבוד מקדים

<u>מידע חסר</u>

מטרת העיבוד המקדים היא להכין את הנתונים לשלב ה-modelling, על מנת שנוכל להתאים את הנתונים למודל וכן שיוכל להשיג ביצועים טובים יותר.

בשלב הראשון טיפלנו במידע חסר בשני אופנים:

1. הסרנו פיצ'רים שיותר ממחצית מהנתונים בו היו ריקים - הוסרו פיצ'רים בהם יותר ממחצית מהמידע חסר. זאת מתוך הנחה כי חוסר המידע עלול לגרום להטיה בתוצאות המודל.
2. עבור הפיצ'רים שנשארו, ביצענו השלמה באמצעות SimpleImputer של Sklearn לערך השכיח ביותר.

בעת מחשבה על אופן מילוי הערכים החסרים, התלבטנו האם לבצע מילוי של הערך הממוצע בכל עמודה, אך ממוצע עלול להכיל ערכים לא הגיוניים. לדוגמה: במידה וחסר ערך בכמות האנשים, השלמת הערך הממוצע עלול להביא להשלמה של ערך לא שלם, דבר שיפגע באותנטיות של הדאטה. סיבה נוספת למילוי הערך השכיח במקום הממוצע הוא שעם ממוצע אי אפשר להשלים ערכים לא מספריים.

<u>בחירת פיצ'רים</u>

כמו כן המשכנו בביצוע של Feature Selection, ובחרנו להסיר פיצ'רים שראינו בשלב העיבוד המקדים שהייתה ביניהם קורלציה גבוהה מדי (גדולה מ-55%), מתוך הנחה שהם תורמים לנו אינפורמציה דומה אך רק מוסיפים לממדיות של הבעיה, בחרנו מתוך כל זוג פיצ'רים עם קורלציה גבוהה את הפיצ'ר שיש לו תרומה יותר גדולה לחיזוי ה-label.

עמודות קטגוריאליות

ניסינו להריץ כמה סוגי קידודים - LabelEncoder, שהתגלה כלא יעיל משום שנתן משמעות לסידור המספרי של הפיצ'רים, כאשר לא הייתה ביניהם משמעות סידורית מסוימת (כמו בפיצ'ר של המדינות). לאחר מכן קידדנו את כל הפיצ'רים באמצעות OneHotEncoder שמשתמש בקידוד של משתנים קטגוריאליים לווקטורים בינאריים. אך בקידוד זה יש בעיה - הוא מייצר עמודות רבות כאשר מספר המופעים של המשתנה הקטגורי רב.

על כן, לבסוף בחרנו לקודד את הפיצ'רים הקטגוריאליים (פרט למדינה) באמצעות הפונקציה OneHotEncoder אשר מקודדת את הנתונים לקידוד בינארי.

עבור הפיצ'ר המכיל את המדינה ממנה בוצעה ההזמנה, החלטנו להתמודד אחרת, זאת בשל הכמות הגדולה יחסית של הערכים השונים שיש לפיצ'ר. חישבנו את אחוז הביטולים כתלות בכל אחת מהמדינות ויצרנו מהם פיצ'ר חדש - Country Cancel Ratio.

חלוקת הדאטה

חלוקת הדאטה הייתה אחת המחלוקות העיקריות בינינו בפרויקט - האם יש לבצע את החלוקה בשלב ה-modelling, וכך גם לדמות את ה-validation set בצורה הקרובה ביותר להרצת ה-test data, או שמא כבר בהתחלה לפני ביצוע האקספלורציה.

לבסוף הסכמנו כי מבחינה הגיונית, בעת ביצוע כל פעולות השלמת המידע, סטנדרטיזציה או כל פונקציה אחרת הדורשת fit, יש לפעול לאחר חלוקת המידע ולבצע אותה בהתאם ל-train בלבד. על כן בשלב העיבוד המקדים ברגע בו החלנו לבצע פעולות אלה בוצעה החלוקה.

הסרת **Outliers**

חיפשנו שיטה שלא תסיר יותר מדי תצפיות, ובכל זאת תוכל להסיר את התצפיות החריגות. לכן, קבענו סף של Z Score < 6 לאחר שגילינו שזה הסף האופטימלי, כלומר הוא אינו מסיר כמות גדולה של ערכים באופן יחסי, אך כן מסיר ערכים חורגים.

סטנדרטיזציה ו-**PCA**

החלנו את הפונקציה StandardScaler רק על הפיצ'רים הנומריים בדאטה. בשלב זה הדאטה שלנו היה מקודד, והכיל כ-34 פיצ'רים, מתוכם 15 מספריים רציפים והשאר בינאריים.

המשכנו לביצוע PCA test על ה-train data וראינו שהפיצ'רים המספריים במודל יכולים להיות מיוצגים ע"י ממד אחד בלבד ללא פגיעה משמעותית בשונות. זוהי תוצאה הגיונית מאחר שהנתונים מוזנים לפונקציה זאת בפרויקט שלנו לאחר סטנדרטיזציה. בפועל, ביצענו מספר וריאציות של הקוד יחד עם PCA וללא PCA, במודלים שונים, אך קיבלנו תוצאות ירודות, ועל כן השמטנו את הביצוע מהקוד הסופי - אם כי הפונקציה ותהליך הבחינה עדיין קיימים בפרויקט עצמו. ראו נספח ג'.

חלק 4: מודלים

מודל נאיבי

יצרנו pipelines לארבעה מודלים: רגרסיה לינארית, KNN, MLP ו-Random Forest. כאשר הוספנו את הסטנדרטיזציה למודלים שדורשים זאת.

את המודלים הרצנו על מגוון וריאציות של עיבודים מקדימים שעשינו, כפי שצוין בדו"ח זה. הדאטה כפי שהוא נבחר שיקף את התוצאות המרביות בהערכת המודלים.

ביצענו עבור כל מודל הערכה מעמיקה עבור ביצועיו, על שני הסטים של המידע (train ו-validation) שכללה:

- הדפסת Classification Report מתוך ספריית Sklearn
- חישוב CV Score עבור Kfolds 10, הצגת התוצאות בגרף
- ביצוע חיפוש להיפר פרמטרים וטיוב כלל המודלים
- ביצוע הערכה חוזרת של CV Scores והשוואתם
- ביצוע של ROC AUC עם Cross Validation עבור Kfolds 10 לכל מודל בנפרד

נעיר כי ביצוע ה-ROC לכל מודל הוא למעשה בחינה של התאמת המודל לדאטה, ואיננו משקף את טיב המודל על המודל המאומן שלנו בפועל, שכן בשיטה זו מודל מסוג זהה לכל אחד מ-4 המודלים מבצע למידה של חלוקות שונות מהדאטה וחיזוי. על כן יצרנו "העתקים" של כל מודל בכדי לבצע השוואה זאת, שכן למדנו שאחרת היינו למעשה מטים את התוצאות (ראינו זאת בפועל כאשר לא ביצענו הפרדה זאת) בכך שהיינו מציגים חלק מהמידע אשר קיים ב-Test עבור המודל. על כן בדיקה זו בעיקר לימדה אותנו שהמודל המתאים ביותר לבעיה הנתונה הוא אכן המודל שבחרנו בסופו של דבר - RandomForesetClassifier.

בדקנו גם את שיעור ההצלחה של מודל "טיפש", בכדי לוודא שאל מול אחוז ההצלחה בחיזוי ביטול אנחנו מתרחקים לפחות בכמה עשרות אחוזים מהאפשרות של סיווג על ידי חיזוי חיובי תמיד.

הערכת המודלים

תשובות לשאלות שניתנו בהגדרת הפרויקט

1. confusion matrix של המודל random forest:

כפי שאפשר לראות, יש למטריצה בציר ה-y יש לנו את הנתונים לגבי מה המודל חזה, ובציר ה-x יש לנו את הערכים האמיתיים. לכן, על האלכסון הראשי יש את הדוגמאות שהצלחנו לסווג נכון.

כאשר אנו מחלקים את הסכום של המספרים על האלכסון הראשי בסכום המטריצה כולה, נקבל את ה-accuracy של המודל.

2. Overfit

היה לנו חשד ל-overfit בהרצה הנאיבית של מודל ה-Random Forest זאת משום שהוא הצליח להגיע ממש ל-100% דיוק על ה-train data אך נתן תוצאות פחות טובות על ה-test data. הפער הזה מעיד על כך שהמודל "שינן" את הדוגמאות עליו הוא התאמן (נספח א'). לאחר כיוונון הפרמטרים של המודל ניכר כי ה-Overfit לא היה קיים יותר.

ביצענו אופטימיזציה של הפרמטרים על כל אחד מהמודלים, באמצעות הפונקציה :GridSearchCV, הפרמטרים של החיפוש נקבעו בהתאם לפרמטרים המקובלים בכל מודל, מתוך חשיבה הגיונית על גודל הדאטה וכמו כן מספר ניסיונות. זוהי הרצה כבדה ביותר שארכה מספר שעות, אך אכן פתרה למשל Overfit במודל ה-RFC. הצגנו בגרף מסודר את השיפור בביצועי כלל המודלים עקב חיפוש מעמיק זה. ראו נספח ד'.

כאמור הבנו כי שהמודל הטוב ביותר הוא ה-Random forest והפרמטרים המיטביים עבורו הם: מקסימום $\sqrt{34}$ פיצ'רים בכל עץ, 230 עצים, מינימום 2 פיצולים, מינימום דגימה אחת בכל עלה.

ציון ה-AUC (השטח היחסי שמתחת ל-roc curve) במודל הנבחר – Random Forest , הינו 0.924. כאמור ציון זה איננו משקף את טיב המודל שאימנו, אלא מחזק את בחירתנו במודל זה לחיזוי הבעיה הנתונה בדאטה לפרויקט זה.

לבסוף ביצענו השוואה של ציוני AUC עבור כל אחד מהמודלים, כאשר המודל בעל הציון הגבוה ביותר עבור המודל המאומן והמכוון שבחרנו הינו AUC של 0.92229 על ה-Validation Set.

חלק 5: ביצוע פרדיקציות

לבסוף, על מנת לבצע פרדיקציות על המודל, השתמשנו ב-pipeline המקבל בעצמו מספר צינורות המבצעים את כלל העיבוד מקדים למידע ומחילים עליו המודל הנבחר. לאחר ביצוע החיזוי מופעל צינור נוסף המבצע עיבוד של התוצאות לכדי פורמט ההגשה מבוקש ומייצא את ההסתברות לפרדיקציות לקובץ CSV.

סיכום

בפרויקט זה הצגנו את תהליך העבודה המסכם את לימודינו בקורס "מבוא ללמידת מכונה". אנו מרגישים כי המודל שסיפקנו בעבודתנו הגמורה הינו מודל המייצג השקעה רבה ומספר רב של גרסאות בהן בדקנו וריאציות רבות לעיבוד, הצגת המידע וביצוע פרדיקציות שונות עד לקבלת תוצאות חיזוי הגונות, עם AUC של 0.924 . נוסיף ונאמר כי גם שאר המודלים הציגו חיזוי ברמה סבירה ביותר, והתרשמנו רבות מהקלות בה ניתן לעבוד עם כלים כאלו (על אף שהם מורכבים ביותר) בכדי לקבל חיזוי ברמת דיוק שכזאת.

נספחים

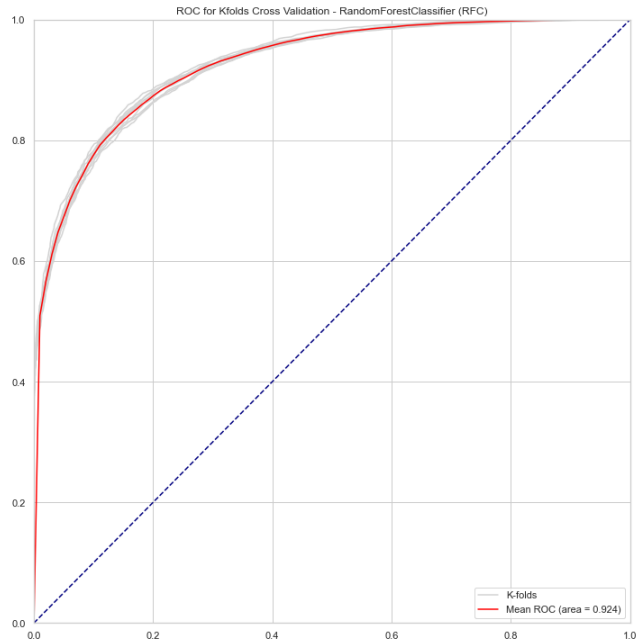
נספח א'

הצגת ה-OverFit עבור מודל לא מכוונן

```
RFC Train Data                precision    recall  f1-score   support

           0       1.00      1.00      1.00     37071
           1       1.00      1.00      1.00     21725

    accuracy                           1.00     58796
   macro avg       1.00      1.00      1.00     58796
weighted avg       1.00      1.00      1.00     58796

 80%|▌▌▌▌▌▌▌▌▌▌  | 4/5 [01:25<00:16, 16.68s/it]
RFC Test Data                precision    recall  f1-score   support

           0       0.84      0.93      0.88     18494
           1       0.86      0.71      0.78     11055

    accuracy                           0.85     29549
   macro avg       0.85      0.82      0.83     29549
weighted avg       0.85      0.85      0.84     29549
```

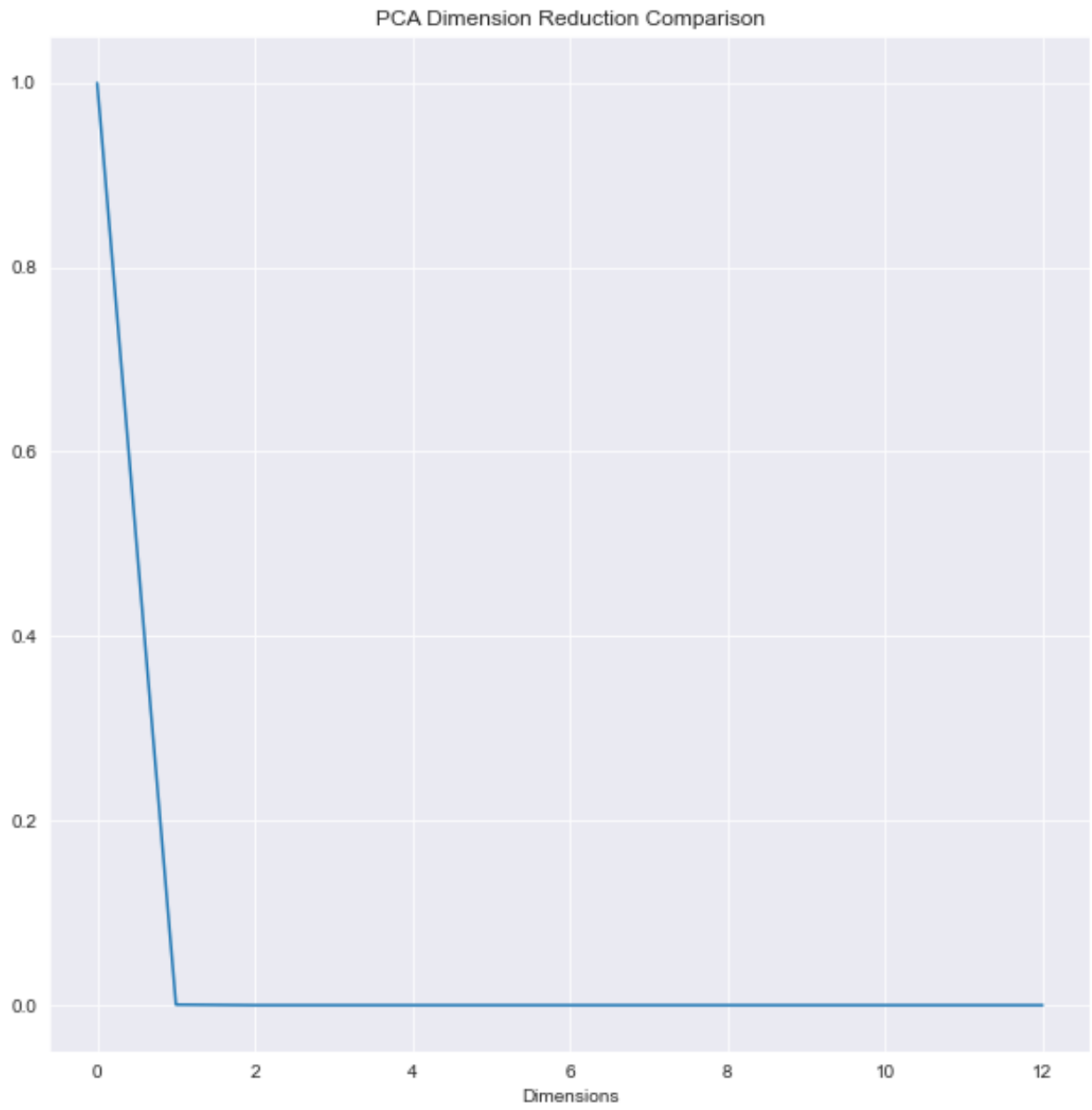
נספח ב'

דוגמה לגרף ROC של RFC



נספח ג'

גרף Explained Ratio PCA



נספח ד'

השוואת CV Scores של מודלים על פני Validation, Test, מכוונים ולא מכוונים.