EX11

roi hezkiyahu

19 5 2022

```
#imports
library(tidyverse)
library(tidymodels)
library(glmnet)
library(glue)
library(microbenchmark)
library(caret)
library(ROCit)
```

Q1

שאלה 1

בשאלה זו נשתמש בנתונים בשם bank הקיימים כקובץ csv במודל.

- y - (כן/לא) האם לקח/ה הלוואה
- age - (נומרי) גיל
- housing - (כן/לא) האם ללקוח/ה יש משכנתא
- university - (כן/לא) האם הלקוח/ה עשה/עשתה תואר באוניברסיטה
- contact - (סלולרי/קווי) המספר אליו נערכה השיחה
- duration - (נומרי) אורך השיחה בשניות
- previous - (נומרי) מספר הפניות הקודמות שנעשו ללקוח/ה לפני קמפיין זה
- marital - (רווק/ה, נשוי/אה, גרוש/ה) מצב משפחתי

נרצה לבנות מודל רגרסיה לוגיסטית החוזה הלוואה, תוך שימוש ברגלוריזציית Lasso.

תוכלו להשתמש בשבעת המשתנים המסבירים הנתונים: גיל, האם ללקוח/ה יש משכנתא, האם עשה/עשתה תואר באוניברסיטה, המספר אליו נערכה השיחה, אורך השיחה בשניות, מספר הפניות הקודמות ומצב משפחתי (ללא אינטראקציות, עם אפקטים לינאריים בלבד למשתנים הרציפים).

א. נרצה לבחון כמה ערכי λ.

I. השתמשו בחבילה glmnet וצרו גרף המתאר את המקדמים של המודל כפונקציה של נורמת L1.
מה ניתן ללמוד מגרף זה?

II. בחרו שלושה ערכי λ כך שמתקבלים שלושה מודלים שונים.

III. דווחו את המקדמים המתקבלים לכל אחד מהמשתנים עבור כל אחד משלושת ערכי λ שבחרתם.
הסבירו בקצרה כיצד המקדמים משתנים כאשר λ גדל.

ב. השתמשו ב Cross Validation תוך שימוש ברגלוריזציית Lasso על מנת לבחור מודל,

I. חשבו את הערך הλ הגדול ביותר אשר ממזער את ממוצע הטעות

(1) פעם אחת ישירות בעזרת שימוש בחבילה glment.

(2) פעם נוספת בעזרת שימוש בשלושת הוקטורים cvm, cvsd ו-lambda הנמצאים באובייקט המתקבל מהרצת cv.glmnet.

- cvm : הטעות הממוצעת ב Cross Validation .
- cvsd : סטיית התקן של cvm.
- lambda : ערכי λ שנבדקו.

(3) עבור ערך λ זה, כמה מקדמים הם לא אפסים?

II. חשבו את ערך הλ הגדול ביותר אשר הטעות שלו בטווח של סטיית תקן אחת מהטעות המינימלית,

(1) פעם אחת ישירות בעזרת שימוש בחבילה glment.

(2) פעם נוספת בעזרת שימוש בשלושת הוקטורים cvm, cvsd ו-lambda הנמצאים באובייקט המתקבל מהרצת cv.glmnet.

- cvm : הטעות הממוצעת ב Cross Validation .
- cvsd : סטיית התקן של cvm.
- lambda : ערכי λ שנבדקו.

(3) עבור ערך λ זה, כמה מקדמים הם לא אפסים?

III. דווחו מהם המודלים בהם הייתם שוקלים להשתמש.
האם המודל שנבחר בתרגיל 10, שאלה 1 ד הוא אחד מהם?

a

```
bank <- read_csv("bank.csv") %>% mutate(y = ifelse(y == "yes",1,0)) %>% select(-1,-2)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

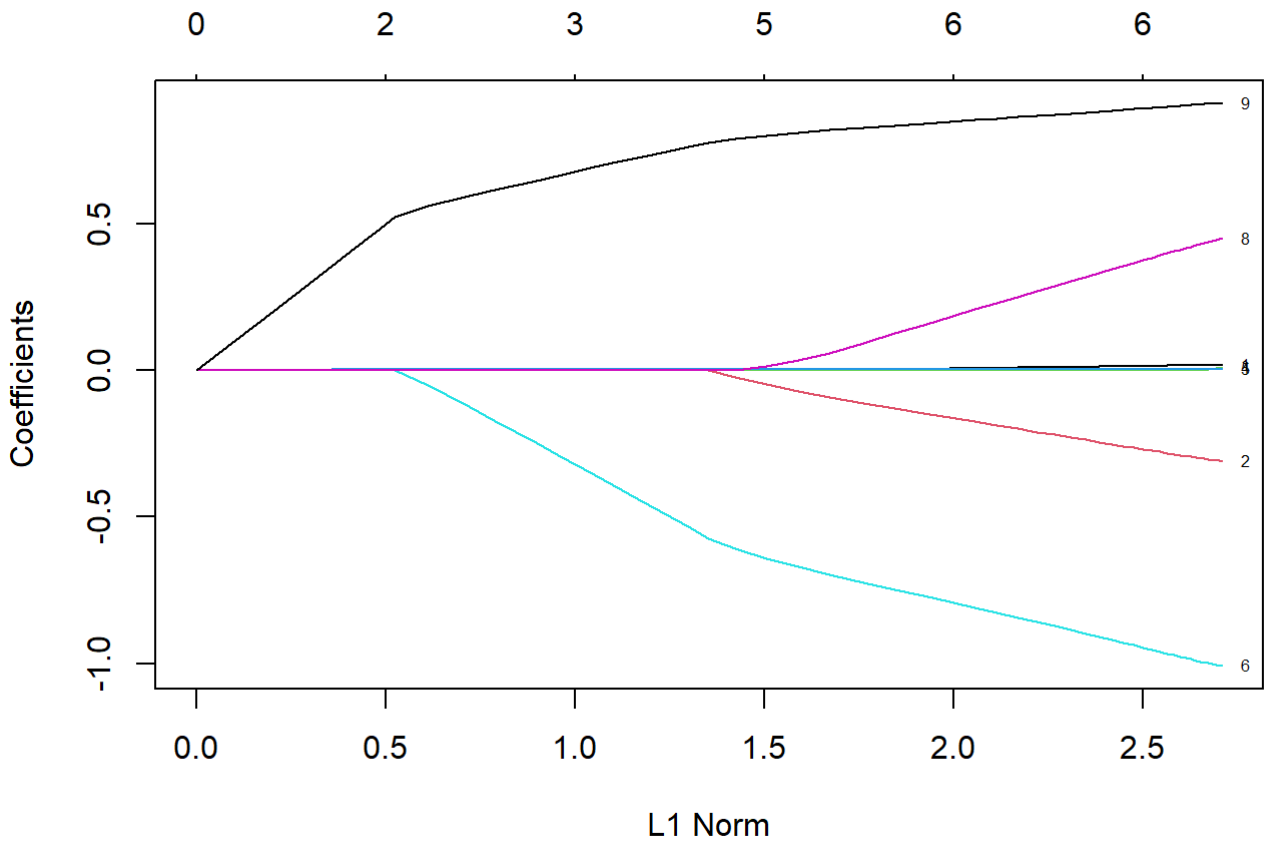
```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   ...1 = col_double(),
##   age = col_double(),
##   university = col_character(),
##   housing = col_character(),
##   duration = col_double(),
##   contact = col_character(),
##   marital = col_character(),
##   previous = col_double(),
##   y = col_character()
## )
```

```
X <- model.matrix(y~ 0+.,data = bank)
y <- bank$y
lasso <- glmnet(X,y,family = "binomial",alpha = 1)
lasso
```

```
##
## Call:  glmnet(x = X, y = y, family = "binomial", alpha = 1)
##
##    Df  %Dev   Lambda
## 1   0  0.00 0.128100
## 2   1  3.49 0.116800
## 3   1  5.91 0.106400
## 4   1  7.75 0.096930
## 5   1  9.24 0.088320
## 6   1 10.46 0.080470
## 7   1 11.47 0.073320
## 8   2 13.32 0.066810
## 9   2 14.93 0.060870
## 10  2 16.21 0.055470
## 11  2 17.26 0.050540
## 12  2 18.12 0.046050
## 13  2 18.84 0.041960
## 14  2 19.45 0.038230
## 15  3 20.16 0.034830
## 16  3 20.84 0.031740
## 17  3 21.41 0.028920
## 18  3 21.91 0.026350
## 19  3 22.33 0.024010
## 20  3 22.70 0.021880
## 21  3 23.01 0.019930
## 22  3 23.27 0.018160
## 23  3 23.50 0.016550
## 24  3 23.69 0.015080
## 25  3 23.86 0.013740
## 26  4 24.03 0.012520
## 27  5 24.21 0.011410
## 28  5 24.36 0.010390
## 29  6 24.51 0.009470
## 30  6 24.68 0.008629
## 31  6 24.83 0.007862
## 32  6 24.95 0.007164
## 33  6 25.05 0.006527
## 34  6 25.14 0.005948
## 35  6 25.21 0.005419
## 36  6 25.27 0.004938
## 37  6 25.32 0.004499
## 38  6 25.37 0.004099
## 39  6 25.40 0.003735
## 40  6 25.43 0.003403
## 41  6 25.46 0.003101
## 42  6 25.48 0.002826
## 43  6 25.50 0.002575
## 44  6 25.51 0.002346
## 45  6 25.52 0.002137
## 46  6 25.53 0.001948
## 47  6 25.54 0.001775
## 48  6 25.55 0.001617
## 49  6 25.55 0.001473
## 50  6 25.56 0.001342
## 51  6 25.56 0.001223
```

```
## 52  6 25.57 0.001114
## 53  6 25.57 0.001015
## 54  6 25.57 0.000925
## 55  7 25.57 0.000843
## 56  7 25.58 0.000768
## 57  7 25.58 0.000700
## 58  7 25.58 0.000638
## 59  7 25.58 0.000581
## 60  7 25.58 0.000530
```

```
plot(lasso, label = T)
```



we can learn from this graph which features enter the model for different L1 norm, we can see that features university,contact,martial,previous are the stronger features

```
model_1 <- glmnet(X,y,family = "binomial",alpha = 1,lambda = 0.1)
model_2 <- glmnet(X,y,family = "binomial",alpha = 1,lambda = 0.025)
model_3 <- glmnet(X,y,family = "binomial",alpha = 1,lambda = 0.001)

glue("model 1 coef")
```

```
## model 1 coef
```

```
coef(model_1)
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                            s0
## (Intercept)      -2.3028314152
## age               .
## universityno      .
## universityyes     .
## housingyes        .
## duration          0.0008480065
## contacttelephone  .
## maritalmarried    .
## maritalsingle     .
## previous          .
```

```
glue("model 2 coef")
```

```
## model 2 coef
```

```
coef(model_2)
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                           s0
## (Intercept)      -3.073630492
## age               .
## universityno      .
## universityyes     .
## housingyes        .
## duration          0.002875006
## contacttelephone -0.271507717
## maritalmarried    .
## maritalsingle     .
## previous          0.657661786
```

```
glue("model 3 coef")
```

```
## model 3 coef
```

```
coef(model_3)
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                            s0
## (Intercept)      -3.998269359
## age               0.016995999
## universityno     -0.297899262
## universityyes     .
## housingyes        .
## duration          0.003910384
## contacttelephone -0.990221562
## maritalmarried    .
## maritalsingle     0.427391765
## previous          0.905993851
```

as lambda increases the parameter coefficients decrease because we have a larger penalty on them, also fewer coefficients are > 0

b

I

1

```
cv_lambda <- cv.glmnet(X,y,family = "binomial",alpha = 1)
glue("best lambda value is: {cv_lambda$lambda.min}")
```

```
## best lambda value is: 0.00052945644827656
```

2

```
min_lambd <- cv_lambda$lambda[which.min(cv_lambda$cvm)]
glue("best lambda value is: {min_lambd}")
```

```
## best lambda value is: 0.00052945644827656
```

3

```
cv_lambda
```

```
##
## Call:  cv.glmnet(x = X, y = y, family = "binomial", alpha = 1)
##
## Measure: Binomial Deviance
##
##       Lambda Measure       SE Nonzero
## min 0.000529  0.5243 0.007090       7
## 1se 0.008629  0.5306 0.006639       6
```

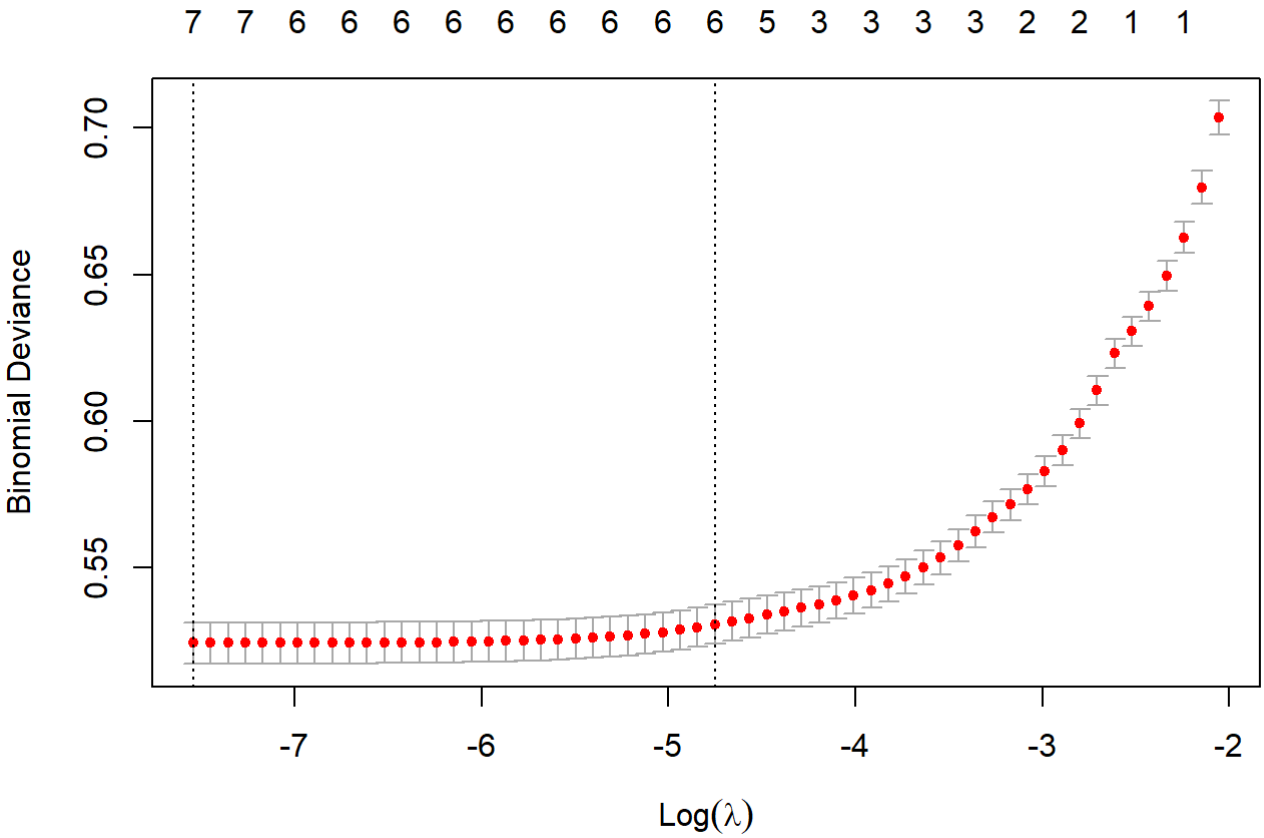
```
glue("from the table we can see that all variables are non zero")
```

```
## from the table we can see that all variables are non zero
```

II

1

```
plot(cv_lambda)
```




```
glue("biggest lambda with 1 std away is approx: {exp(-4.6)}")
```

```
## biggest lambda with 1 std away is approx: 0.0100518357446336
```

2

```
ci <- min(cv_lambda$cvm) + c(-1,1)* cv_lambda$cvsd[60]
# all values are begger then the lower bound
lambdas <- cv_lambda$lambda[cv_lambda$cvm < ci[2]]
biggest_lambda <- max(lambdas)
biggest_lambda
```

```
## [1] 0.008628821
```

3

```
non_zero <- tidy(cv_lambda) %>% filter(near(lambda,0.01039345)) %>% pull (nzero)
glue ("the number f non zero paramaters is: {non_zero}")
```

```
## the number f non zero paramaters is: 5
```

III

```
coef(glmnet(X,y,family = "binomial",alpha = 1,lambda = 0.01039345))
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                            s0
## (Intercept)      -3.189224589
## age               .
## universityno     -0.070439273
## universityyes     .
## housingyes        .
## duration          0.003427289
## contacttelephone -0.670283853
## maritalmarried    .
## maritalsingle     0.035323713
## previous          0.811748315
```

```
coef(glmnet(X,y,family = "binomial",alpha = 1,lambda = exp(-6)))
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                            s0
## (Intercept)      -3.849269993
## age               0.013969495
## universityno     -0.260218274
## universityyes     .
## housingyes        .
## duration          0.003822619
## contacttelephone -0.932407873
## maritalmarried    .
## maritalsingle     0.359963058
## previous          0.889398029
```

```
coef(glmnet(X,y,family = "binomial",alpha = 1,lambda = exp(-7.3)))
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                            s0
## (Intercept)      -4.034505422
## age               0.017667374
## universityno     -0.306240398
## universityyes     .
## housingyes        0.005026383
## duration          0.003930473
## contacttelephone -1.002955199
## maritalmarried    .
## maritalsingle     0.442358746
## previous          0.909782412
```

i would chose one of these 3 models as all of them are in lambdas CI, these are not the same model as EX10.1.d

Q2

שאלה 2

נשתמש בנתונים בשם bank הקיימים כקובץ csv במודל.
נרצה לבנות מודל רגרסיה לוגיסטית החוזה הלוואה.
לשם כך נשתמש במודל המכיל את המשתנים המסבירים: גיל, האם עשה/עשתה תואר באוניברסיטה, המספר אליו נערכה השיחה, אורך השיחה בשניות, מספר הפניות הקודמות ומצב משפחתי (ללא אינטראקציות, עם אפקטים לינאריים בלבד למשתנים הרציפים).

א. חלקו את המודל באקראי ל-70% training set ול-30% test set.

ב. התאימו את המודל על ה training set וחזו סיכוי להלוואה עבור כל אחת מהתצפיות בtest set.

ג. על מנת ליצור סיווג, נבחן את: $\theta = 0.05, 0.1, 0.2$.
צרו וקטור של ערכי תוצאה חזויים $(\bar{Y})$ עבור כל אחד משלושת ערכי ה θ האפשריים.

ד. בסעיף זה נתמקד ב TP (true positive), FP (false positive), TN (true negative), FN (false negative) .

I. הסבירו בקצרה מה המשמעות של כל אחד מהם עבור בעיה זו.

II. דווחו את ערכי ה- TP, FP, TN, FN המתקבלים עבור כל אחד מערכי θ שבדקתם.

III. דווחו את ערכי ה- TP, FP, TN, FN והסבירו את המשמעות שלהם, עבור
(1) $\theta = 0$
(2) $\theta = 0.5$
(3) $\theta = 1$

IV. הסבירו כיצד כל אחד מבין TP, FP, TN, FN משתנה כאשר θ גדל.

ה. בסעיף זה נתמקד ב accuracy, sensitivity¸specificity.

I. הסבירו בקצרה מה המשמעות של כל אחד מהם עבור בעיה זו.

II. דווחו את ערכי ה- accuracy, sensitivity¸specificity המתקבלים עבור כל אחד מערכי θ שבדקתם.

III. דווחו את ערכי ה- accuracy, sensitivity¸specificity והסבירו את המשמעות שלהם, עבור
(1) $\theta = 0$
(2) $\theta = 0.5$
(3) $\theta = 1$

IV. הסבירו כיצד כל אחד מבין accuracy, sensitivity¸specificity משתנה כאשר θ גדל.

ו. ציירו Roc curve עבור מודל זה. הסבירו בקצרה מה ניתן ללמוד מגרף זה.

ז. לסיכום, מהו ערך ה θ שהייתם בוחרים? האם הייתם בוחרים ערך שונה במצבים שונים?

a

```
train_inds <- sample(1:nrow(X),floor(0.7*nrow(X)))
train <- bank %>% slice(train_inds)
test <- bank %>% slice(-train_inds)
```

b

```
logistic_model <- glm(y~.,data = train,family = "binomial")
test$y_pred <- predict(logistic_model,test,type = "response")
test
```

```
## # A tibble: 12,357 x 9
##      age university housing duration contact   marital previous     y y_pred
##    <dbl> <chr>      <chr>      <dbl> <chr>     <chr>      <dbl> <dbl>  <dbl>
##  1    56 no         no           261 telephone married        0     0 0.0365
##  2    40 no         no           151 telephone married        0     0 0.0174
##  3    45 no         no           198 telephone married        0     0 0.0230
##  4    24 no         yes          380 telephone single         0     0 0.0481
##  5    25 no         yes          222 telephone single         0     0 0.0272
##  6    29 no         no           137 telephone single         0     0 0.0212
##  7    35 no         yes          146 telephone married        0     0 0.0155
##  8    50 no         yes          353 telephone married        0     0 0.0459
##  9    55 no         yes          262 telephone married        0     0 0.0361
## 10    35 yes        no            99 telephone married        0     0 0.0175
## # ... with 12,347 more rows
```

c

```
test_with_thetas <- test %>% mutate(theta05 = as.numeric(y_pred>0.05)) %>%
  mutate(theta1 = as.numeric(y_pred>0.1)) %>%
  mutate(theta2 = as.numeric(y_pred>0.2))
test_with_thetas %>% summarise(across(c(theta05,theta1,theta2),mean))
```

```
## # A tibble: 1 x 3
##   theta05 theta1 theta2
##     <dbl>  <dbl>  <dbl>
## 1   0.551  0.288  0.139
```

d

I

false negative we identified an observation as negative (meaning no loan), but the true value was positive

true negative we identified an observation as negative (meaning no loan) and we were correct

false positive we identified an observation as positive (meaning no loan), but the true value was negative

true positive we identified an observation as positive (meaning no loan) and we were correct

II

```
tbl_theta <- function(theta){
  cm <- confusionMatrix(data = factor(as.numeric(test$y_pred>theta),levels = c(0,1)),referenc
e  = factor(test$y))$table
  glue("for theta = {theta}
       fn = {cm[2,1]}
       tn = {cm[1,1]}
       fp = {cm[1,2]}
       tp = {cm[2,2]}
       ")
}
tbl_theta(0.05)
```

```
## for theta = 0.05
## fn = 5417
## tn = 5492
## fp = 62
## tp = 1386
```

```
tbl_theta(0.1)
```

```
## for theta = 0.1
## fn = 2378
## tn = 8531
## fp = 272
## tp = 1176
```

```
tbl_theta(0.2)
```

```
## for theta = 0.2
## fn = 891
## tn = 10018
## fp = 624
## tp = 824
```

III

```
tbl_theta(0)
```

```
## for theta = 0
## fn = 10909
## tn = 0
## fp = 0
## tp = 1448
```

```
tbl_theta(0.5)
```

```
## for theta = 0.5
## fn = 177
## tn = 10732
## fp = 1095
## tp = 353
```

```
tbl_theta(1)
```

```
## for theta = 1
## fn = 0
## tn = 10909
## fp = 1448
## tp = 0
```

for theta = 0 we assume all predictions are true there for we only have true positives and false negatives, same goes for theta = 1 but the other way around

for theta = 0.5 we assume that the chance of taking a lone is the same as not taking one

IV

as theta increases: fn decrease

tn increase

tp decrease

fp decrease

e

I

specificity is P(Y_pred = 0|Y=0)

accuracy is the ratio of our correctly classified predictions out of all the observations

sensitivity is P(Y_pred = 1|Y=1)

II + III

```
acc_vales <- function(theta){
  cm <- confusionMatrix(data = factor(as.numeric(test$y_pred>theta),levels = c(0,1)),referenc
e  = factor(test$y))
  acc = cm$overall["Accuracy"]
  sensi = cm$byClass["Sensitivity"]
  speci = cm$byClass["Specificity"]
  glue("for theta = {theta}
       Accuracy = {acc}
       Sensitivity = {sensi}
       Specificity = {speci}
       ")
}

map(c(0,0.05,0.1,0.2,0.5,1),acc_vales)
```

```
## [[1]]
## for theta = 0
## Accuracy = 0.117180545439832
## Sensitivity = 0
## Specificity = 1
##
## [[2]]
## for theta = 0.05
## Accuracy = 0.5566075908392
## Sensitivity = 0.503437528646072
## Specificity = 0.957182320441989
##
## [[3]]
## for theta = 0.1
## Accuracy = 0.78554665371854
## Sensitivity = 0.782014850123751
## Specificity = 0.812154696132597
##
## [[4]]
## for theta = 0.2
## Accuracy = 0.877397426559845
## Sensitivity = 0.918324319369328
## Specificity = 0.569060773480663
##
## [[5]]
## for theta = 0.5
## Accuracy = 0.897062393784899
## Sensitivity = 0.98377486479054
## Specificity = 0.24378453038674
##
## [[6]]
## for theta = 1
## Accuracy = 0.882819454560168
## Sensitivity = 1
## Specificity = 0
```

IV

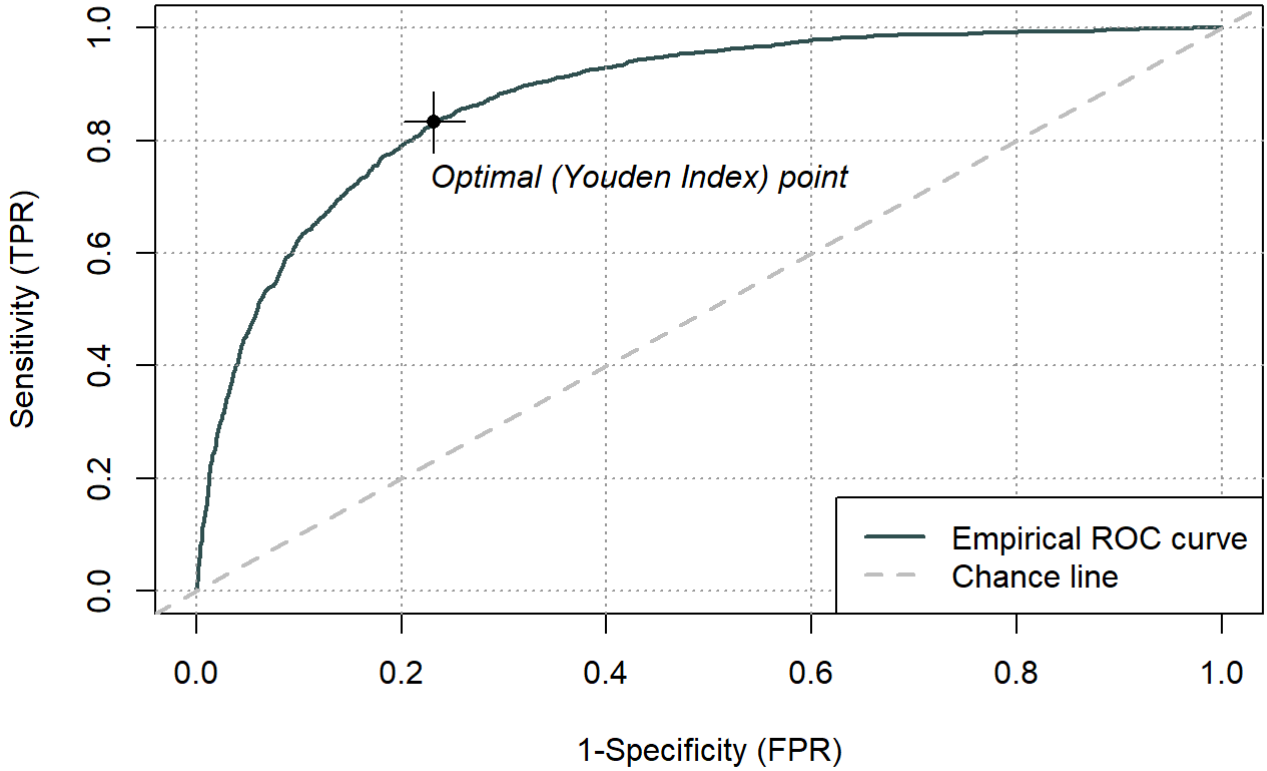
Accuracy increases then decreases

Sensitivity increases

Specificity decreases

f

```
## Warning: package 'ROCit' was built under R version 3.5.2
ROCit_obj <- rocit(score=test$y_pred,class=factor(test$y))
plot(ROCit_obj)
```



we can learn from this graph for each cut point theta what is the Sensitivity (y-axis) and the Specificity(x-axis)

g

it depends what we are more concerned with is it better to predict a loan but a person wont take it or not to predict but he would take it?

because i don't know which is better i would go with youden index meaning theta = 0.25 (approx)

Q3

שאלה 3

נשתמש בנתונים בשם bank_full הקיימים כקובץ csv במודל.

אלו אותם נתונים בהם השתמשנו בשאלות הקודמות אך כעת נוספו עוד משתנים מסבירים בהם ניתן להשתמש. המשתנים הקיימים הם:

- y - האם לקח/ה הלוואה (כן/לא)
- age - גיל (נומרי)
- housing - האם ללקוח/ה יש משכנתא (כן/לא)
- university - האם הלקוח/ה עשה/עשתה תואר באוניברסיטה (כן/לא)
- contact - המספר אליו נערכה השיחה (סלולרי/קווי)
- duration - אורך השיחה בשניות (נומרי)
- previous - מספר הפניות הקודמות שנעשו ללקוח/ה לפני קמפיין זה (נומרי)
- marital - מצב משפחתי (רווק/ה, נשוי/אה, גרוש/ה)
- job - עבודה (קטגוריאלי)
- default - האם יש ללקוח/ה קרדיט דיפולטי (כן/ לא)
- loan - האם יש ללקוח/ה הלוואה קיימת (כן/ לא)
- month - החודש בשנה בו פנו ללקוח/ה בפעם האחרונה (קטגוריאלי)
- day_of_week - היום בשנה בו פנו ללקוח/ה בפעם האחרונה (קטגוריאלי)
- campaign- מספר הפעמים בהן פנו ללקוח/ה כחלק מקמפיין זה (נומרי)
- pdays - מספר הימים שעברו מאז שפנו ללקוח/ה בפעם האחרונה כחלק מקמפיין זה (נומרי)
- poutcome - תוצאה של הפניות הקודמות בקמפיין זה (הצלחה/ כישלון/ לא הייתה פניה קודמת)

נרצה לבנות מודל רגרסיה לוגיסטית החוזה הלוואה.

א. אם נרצה להתאים מודל בעזרת משתנים אלה, ללא אינטראקציות, עם אפקטים לינאריים בלבד למשתנים הרציפים, ללא רגולריזציה.

   I. כמה מודלים שונים נוכל להתאים?

   II. העריכו כמה זמן יקח להתאים את כל המודלים האפשריים הללו (במחשב שלכם).

ב. אם נרצה להתאים מודל בעזרת משתנים אלה, עם אינטראקציות זוגיות, עם אפקטים לינאריים בלבד למשתנים הרציפים, ללא רגולריזציה.

   I. כמה מודלים שונים נוכל להתאים?

   II. העריכו כמה זמן יקח להתאים את כל המודלים האפשריים הללו (במחשב שלכם).

ג. נרצה להתאים מודל בעזרת משתנים אלה, עם אינטראקציות זוגיות, עם אפקטים לינאריים בלבד למשתנים הרציפים, תוך שימוש ברגלוריזציית Lasso.
השתמשו ב Cross Validation על מנת לבחור את המודל.
את מטריצת ה-X הכוללת אפקטים עיקריים ואינטראקציות זוגיות תוכלו למשל ליצור ב R באופן הזה:
model.matrix(y ~ .^2, bank, family = "binomial")[,-1]

   I. בחרו מודל בעזרת הλ הגדול ביותר אשר ממזער את ממוצע הטעות.

(1) דווחו אילו משתנים מסבירים נכנסו למודל זה ומה האומדים לפרמטרים שלהם.
(2) עבור ערך λ זה, כמה מקדמים הם אפסים?
(3) האם ישנה בעיתיות בפירוש של המודל?

II. בחרו מודל בעזרת הλ הגדול ביותר אשר הטעות שלו בטווח של סטיית תקן אחת מהטעות המינימלית.

(1) דווחו אילו משתנים מסבירים נכנסו למודל זה ומה האומדים לפרמטרים שלהם.
(2) עבור ערך λ זה, כמה מקדמים הם אפסים?
(3) האם ישנה בעיתיות בפירוש של המודל?

a

I

$$2^{14} = 16384\ models$$

II

lets assume that training time is symetric around 7 (the mean number of predictors)

```
bank_full <- read_csv("bank_full.csv") %>% mutate(y = ifelse(y == "yes",1,0)) %>% select(-1,-
2)
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   age = col_double(),
##   job = col_character(),
##   marital = col_character(),
##   default = col_character(),
##   housing = col_character(),
##   loan = col_character(),
##   contact = col_character(),
##   month = col_character(),
##   day_of_week = col_character(),
##   duration = col_double(),
##   campaign = col_double(),
##   pdays = col_double(),
##   previous = col_double(),
##   poutcome = col_character(),
##   y = col_character(),
##   university = col_character()
## )
```

```
totsec <- mean(microbenchmark(glm(y~.,data = bank_full[c(sample(colnames(bank_full)[-14],7),
"y")],family = "binomial"), times = 100, unit = "s")$time/10^9)
glue("the estimated amout of time is: {totsec*16384/60} mintues ")
```

```
## the estimated amout of time is: 142.092992785067 mintues
```

b

I

$$2^{\binom{14}{2}+14} \approx 4.05 * 10^{31} models$$

II

ill make the same assumption but now i will use all the interactions for the selected predictors

```
totmin <- mean(microbenchmark(glm(y~.^2,data = bank_full[c(sample(colnames(bank_full)[-14],7
),"y")],family = "binomial"), times = 10, unit = "s")$time/10^9)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
glue("the estimated amout of time is: {totmin*4.05*10^31/60/60/24/365} years")
```

```
## the estimated amout of time is: 4.56992562071918e+24 years
```

c

I

cv.glmnet does not work for some reason, i left the code

```
X <- model.matrix(y~ .^2,data = bank_full)
y <- bank_full$y

las_cv <- cv.glmnet(X,y,family = "binomial",alpha = 1,nfolds = 5,lambda = seq(0.005,0.15,leng
th.out=50))
las_cv

plot(las_cv)
lasso_model <- glmnet(X,y,family = "binomial",alpha = 1,lambda = las_cv$lambda.min)
```

1

```
coef(lasso_model)
```

2

```
ncol(X) - tidy(cv_lambda) %>% filter(near(lambda,las_cv$lambda.min)) %>% pull (nzero)
```

3

we can interpet the model nicley

II

1

```
plot(las_cv)

ci <- min(cv_lambda$cvm) + c(-1,1)* cv_lambda$cvsd[which.min(cv_lambda$cvm)]
# all values are begger then the lower bound
lambdas <- cv_lambda$lambda[cv_lambda$cvm < ci[2]]
biggest_lambda <- max(lambdas)
biggest_lambda
```

2

```
ncols(X) - tidy(cv_lambda) %>% filter(near(lambda,las_cv$lambda.min)) %>% pull (nzero)
```

3

too many predictors makes the interpettebbilty of the model harder