

Gov 2018: Section 3 LASSO and Post Double Selection

Your name:

Wednesday February 14, 2024

Part 1: OLS and LASSO

Question 1. Setting up New York Times Annotated Corpus

In the part of the exercise, we are going to analyze the New York Times Annotated Corpus. From Canvas please download `NYT.RData` and load the file.

This loads a list, `nyt_list`, with the following components:

- `train` : the document term matrix for the training set
- `train_label`: an indicator equal to 1 if the story comes from the national desk for each document in the training set
- `test`: the document term matrix for the test set.

- `test_label`: an indicator equal to 1 if the story comes from the national desk for each document in the test set

We will work with `train` and `train_label` to build our prediction models. We will use the `test` set to test the fit of our model.

Run the following code to put these components in individual objects:

```
rm(list=ls())
#setwd('~YOUR/DIRECTORY/HERE')
load("NYT.RData")
train<- nyt_list[[1]]
train_label<- nyt_list[[2]]
test<- nyt_list[[3]]
test_label<- nyt_list[[4]]
```

Print the dimensions of the train and test set. What is the ratio of n to the number of covariates?

Note that the `train` and `test` matrices do not contain a column for the labels. Combine the document term matrix and labels into two data frames for the train set and for the test set.

```
## your code here
```

Answer 1

```
dim(test)
```

```
## [1] 88 1000
```

```
dim(train)
```

```
## [1] 200 1000
```

```
cat("The ratio of n to cov for test is: ",nrow(test)/ncol(test),"\n")
```

```
## The ratio of n to cov for test is: 0.088
```

```
cat("The ratio of n to cov for train is: ", nrow(train)/ncol(train),"\n")
```

```
## The ratio of n to cov for train is: 0.2
```

```
train.df <- as.data.frame(cbind(train, train_label))
```

```
test.df <- as.data.frame(cbind(test, test_label))
```

Question 2. Linear Probability Model

We are ready to apply a linear probability model to perform classification. Using the `lm` function regress `train_label` against all the words in `train`. To do this, note that you can include all the variable in a regression using the following syntax: `full_reg <- lm(train_label ~ . , data = train.df)`

The `~.` tells R to include all the variables in the data frame.

Analyze the coefficients from `full_reg`, what do you notice? Specifically, what happens to the number of coefficients in the model?

```
## your code here
```

Answer 2

```
full_reg <- lm(train_label ~ . , data = train.df)
```

```
summary(full_reg)
```

```
##
```

```
## Call:
```

```
## lm(formula = train_label ~ . , data = train.df)
```

```
##
```

```
## Residuals:
```

```
## ALL 200 residuals are 0: no residual degrees of freedom!
```

```
##
```

```
## Coefficients: (801 not defined because of singularities)
```

```
##           Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -0.834089         NaN      NaN      NaN
```

```
## mr           0.002376         NaN      NaN      NaN
```

## year	-0.135075	NaN	NaN	NaN
## state	-0.131555	NaN	NaN	NaN
## vote	0.082484	NaN	NaN	NaN
## bush	0.038113	NaN	NaN	NaN
## elect	-0.021215	NaN	NaN	NaN
## time	0.110133	NaN	NaN	NaN
## peopl	-0.323673	NaN	NaN	NaN
## democrat	-0.215114	NaN	NaN	NaN
## republican	0.841856	NaN	NaN	NaN
## kerri	-0.244728	NaN	NaN	NaN
## day	0.168282	NaN	NaN	NaN
## percent	0.003585	NaN	NaN	NaN
## voter	0.104117	NaN	NaN	NaN
## presid	0.290382	NaN	NaN	NaN
## work	0.131921	NaN	NaN	NaN
## compani	0.166860	NaN	NaN	NaN
## make	-0.242899	NaN	NaN	NaN
## million	-0.408072	NaN	NaN	NaN
## campaign	-0.264775	NaN	NaN	NaN
## poll	0.215594	NaN	NaN	NaN
## american	-0.291735	NaN	NaN	NaN
## call	0.544669	NaN	NaN	NaN
## nation	-0.147124	NaN	NaN	NaN
## play	-0.019157	NaN	NaN	NaN
## report	0.499244	NaN	NaN	NaN
## X000	0.247259	NaN	NaN	NaN
## senat	0.095396	NaN	NaN	NaN
## team	0.318106	NaN	NaN	NaN
## X1	0.435223	NaN	NaN	NaN
## ms	-0.584836	NaN	NaN	NaN
## show	0.234104	NaN	NaN	NaN
## race	-0.056761	NaN	NaN	NaN
## week	-0.413232	NaN	NaN	NaN
## back	0.074105	NaN	NaN	NaN
## unit	-0.396167	NaN	NaN	NaN
## citi	-0.500861	NaN	NaN	NaN
## includ	0.797823	NaN	NaN	NaN
## run	-0.301010	NaN	NaN	NaN
## york	0.390734	NaN	NaN	NaN
## made	0.689005	NaN	NaN	NaN
## long	-0.316295	NaN	NaN	NaN
## group	-0.368647	NaN	NaN	NaN
## offici	0.140978	NaN	NaN	NaN
## hous	-0.285889	NaN	NaN	NaN
## season	-0.134547	NaN	NaN	NaN
## game	-0.100964	NaN	NaN	NaN
## X2	-0.497383	NaN	NaN	NaN
## world	0.202757	NaN	NaN	NaN
## govern	-0.251873	NaN	NaN	NaN
## countri	0.680269	NaN	NaN	NaN
## john	-0.988063	NaN	NaN	NaN
## parti	-0.738360	NaN	NaN	NaN
## support	-0.799493	NaN	NaN	NaN
## point	-0.917728	NaN	NaN	NaN

## home	-0.345326	NaN	NaN	NaN
## polit	-0.248197	NaN	NaN	NaN
## close	-0.280419	NaN	NaN	NaN
## news	-0.369641	NaN	NaN	NaN
## manag	-0.275981	NaN	NaN	NaN
## challeng	-0.160019	NaN	NaN	NaN
## end	0.506465	NaN	NaN	NaN
## win	0.150458	NaN	NaN	NaN
## don	0.671808	NaN	NaN	NaN
## part	-0.612541	NaN	NaN	NaN
## ballot	0.427791	NaN	NaN	NaN
## start	0.286078	NaN	NaN	NaN
## court	0.139476	NaN	NaN	NaN
## offic	0.039450	NaN	NaN	NaN
## won	1.502998	NaN	NaN	NaN
## recent	0.130824	NaN	NaN	NaN
## ad	-0.881173	NaN	NaN	NaN
## month	0.531073	NaN	NaN	NaN
## candid	0.731037	NaN	NaN	NaN
## line	-1.080072	NaN	NaN	NaN
## gener	0.565095	NaN	NaN	NaN
## night	0.278401	NaN	NaN	NaN
## expect	-1.072608	NaN	NaN	NaN
## high	0.539800	NaN	NaN	NaN
## major	-0.481338	NaN	NaN	NaN
## number	0.332995	NaN	NaN	NaN
## lead	0.065598	NaN	NaN	NaN
## ago	0.183364	NaN	NaN	NaN
## issu	-0.768440	NaN	NaN	NaN
## left	-0.050068	NaN	NaN	NaN
## execut	0.954717	NaN	NaN	NaN
## director	-0.611593	NaN	NaN	NaN
## repres	-0.630897	NaN	NaN	NaN
## X2000	-0.810765	NaN	NaN	NaN
## turn	0.035846	NaN	NaN	NaN
## earli	0.019776	NaN	NaN	NaN
## chief	0.215560	NaN	NaN	NaN
## dr	-0.018431	NaN	NaN	NaN
## return	-0.004168	NaN	NaN	NaN
## busi	0.092464	NaN	NaN	NaN
## good	0.392070	NaN	NaN	NaN
## plan	0.135742	NaN	NaN	NaN
## base	0.070441	NaN	NaN	NaN
## open	-0.293394	NaN	NaN	NaN
## list	0.203822	NaN	NaN	NaN
## told	-0.324829	NaN	NaN	NaN
## center	0.025516	NaN	NaN	NaN
## ohio	-0.644758	NaN	NaN	NaN
## X10	0.897142	NaN	NaN	NaN
## ve	0.524292	NaN	NaN	NaN
## job	-0.212918	NaN	NaN	NaN
## face	-0.017457	NaN	NaN	NaN
## worker	-1.106879	NaN	NaN	NaN
## thing	1.036948	NaN	NaN	NaN

## big	0.376632	NaN	NaN	NaN
## case	-0.444223	NaN	NaN	NaN
## victori	-0.157730	NaN	NaN	NaN
## X5	-0.706252	NaN	NaN	NaN
## hour	0.227252	NaN	NaN	NaN
## result	-0.457804	NaN	NaN	NaN
## florida	0.215252	NaN	NaN	NaN
## record	-1.530590	NaN	NaN	NaN
## X3	0.324894	NaN	NaN	NaN
## talk	-0.990877	NaN	NaN	NaN
## china	-0.229115	NaN	NaN	NaN
## sign	0.170090	NaN	NaN	NaN
## set	0.132477	NaN	NaN	NaN
## live	0.949188	NaN	NaN	NaN
## put	-1.520416	NaN	NaN	NaN
## offer	-0.031564	NaN	NaN	NaN
## iraq	-0.056756	NaN	NaN	NaN
## public	0.220849	NaN	NaN	NaN
## presidenti	0.130378	NaN	NaN	NaN
## coach	-0.021988	NaN	NaN	NaN
## move	0.363748	NaN	NaN	NaN
## market	0.754294	NaN	NaN	NaN
## tuesday	0.767870	NaN	NaN	NaN
## sunday	1.746363	NaN	NaN	NaN
## famili	-0.158813	NaN	NaN	NaN
## polic	-0.366117	NaN	NaN	NaN
## receiv	-0.905729	NaN	NaN	NaN
## player	-0.577244	NaN	NaN	NaN
## person	0.174298	NaN	NaN	NaN
## monday	0.494506	NaN	NaN	NaN
## servic	-0.096163	NaN	NaN	NaN
## problem	-0.683291	NaN	NaN	NaN
## forc	-0.286810	NaN	NaN	NaN
## final	-0.177616	NaN	NaN	NaN
## system	0.109223	NaN	NaN	NaN
## man	-0.014515	NaN	NaN	NaN
## kill	0.044795	NaN	NaN	NaN
## war	-0.336227	NaN	NaN	NaN
## lot	0.281514	NaN	NaN	NaN
## question	1.761310	NaN	NaN	NaN
## trade	0.134093	NaN	NaN	NaN
## counti	1.544319	NaN	NaN	NaN
## yesterday	1.355226	NaN	NaN	NaN
## school	0.658244	NaN	NaN	NaN
## import	0.287562	NaN	NaN	NaN
## side	1.235257	NaN	NaN	NaN
## X2004	-0.312573	NaN	NaN	NaN
## cost	0.528937	NaN	NaN	NaN
## washington	0.167304	NaN	NaN	NaN
## univers	0.778017	NaN	NaN	NaN
## program	-0.228423	NaN	NaN	NaN
## life	-0.052566	NaN	NaN	NaN
## lost	-1.016005	NaN	NaN	NaN
## seat	0.046281	NaN	NaN	NaN

## member	0.567511	NaN	NaN	NaN
## hand	-0.518461	NaN	NaN	NaN
## deal	-0.172299	NaN	NaN	NaN
## intern	1.134476	NaN	NaN	NaN
## X4	0.430497	NaN	NaN	NaN
## employe	-0.272915	NaN	NaN	NaN
## term	-0.837326	NaN	NaN	NaN
## contract	0.098675	NaN	NaN	NaN
## find	0.330751	NaN	NaN	NaN
## larg	-1.034846	NaN	NaN	NaN
## health	0.096763	NaN	NaN	NaN
## interest	1.202402	NaN	NaN	NaN
## research	-0.289364	NaN	NaN	NaN
## money	0.153078	NaN	NaN	NaN
## street	0.805491	NaN	NaN	NaN
## found	0.436493	NaN	NaN	NaN
## product	0.175038	NaN	NaN	NaN
## secur	0.280076	NaN	NaN	NaN
## white	0.319960	NaN	NaN	NaN
## book	0.011064	NaN	NaN	NaN
## inform	-0.106826	NaN	NaN	NaN
## seri	-0.029016	NaN	NaN	NaN
## care	1.142920	NaN	NaN	NaN
## price	0.082571	NaN	NaN	NaN
## televis	0.451963	NaN	NaN	NaN
## site	-0.411784	NaN	NaN	NaN
## law	0.361449	NaN	NaN	NaN
## red	0.014302	NaN	NaN	NaN
## head	-0.978863	NaN	NaN	NaN
## decis	-0.771066	NaN	NaN	NaN
## al	-0.801393	NaN	NaN	NaN
## america	-0.242332	NaN	NaN	NaN
## rule	0.438728	NaN	NaN	NaN
## remain	-0.371184	NaN	NaN	NaN
## X20	-0.365326	NaN	NaN	NaN
## X30	0.882641	NaN	NaN	NaN
## increas	NA	NA	NA	NA
## share	NA	NA	NA	NA
## agenc	NA	NA	NA	NA
## critic	NA	NA	NA	NA
## music	NA	NA	NA	NA
## X6	NA	NA	NA	NA
## area	NA	NA	NA	NA
## count	NA	NA	NA	NA
## continu	NA	NA	NA	NA
## local	NA	NA	NA	NA
## effort	NA	NA	NA	NA
## industri	NA	NA	NA	NA
## past	NA	NA	NA	NA
## governor	NA	NA	NA	NA
## network	NA	NA	NA	NA
## half	NA	NA	NA	NA
## film	NA	NA	NA	NA
## build	NA	NA	NA	NA

## organ	NA	NA	NA	NA
## union	NA	NA	NA	NA
## bank	NA	NA	NA	NA
## hard	NA	NA	NA	NA
## held	NA	NA	NA	NA
## quarter	NA	NA	NA	NA
## south	NA	NA	NA	NA
## develop	NA	NA	NA	NA
## produc	NA	NA	NA	NA
## pass	NA	NA	NA	NA
## leagu	NA	NA	NA	NA
## matter	NA	NA	NA	NA
## oil	NA	NA	NA	NA
## children	NA	NA	NA	NA
## feel	NA	NA	NA	NA
## attack	NA	NA	NA	NA
## top	NA	NA	NA	NA
## boston	NA	NA	NA	NA
## defeat	NA	NA	NA	NA
## feder	NA	NA	NA	NA
## interview	NA	NA	NA	NA
## volunt	NA	NA	NA	NA
## pay	NA	NA	NA	NA
## kind	NA	NA	NA	NA
## leader	NA	NA	NA	NA
## score	NA	NA	NA	NA
## oper	NA	NA	NA	NA
## thousand	NA	NA	NA	NA
## rais	NA	NA	NA	NA
## men	NA	NA	NA	NA
## met	NA	NA	NA	NA
## tax	NA	NA	NA	NA
## project	NA	NA	NA	NA
## west	NA	NA	NA	NA
## control	NA	NA	NA	NA
## polici	NA	NA	NA	NA
## yard	NA	NA	NA	NA
## jet	NA	NA	NA	NA
## billion	NA	NA	NA	NA
## field	NA	NA	NA	NA
## bill	NA	NA	NA	NA
## led	NA	NA	NA	NA
## district	NA	NA	NA	NA
## technolog	NA	NA	NA	NA
## perform	NA	NA	NA	NA
## X11	NA	NA	NA	NA
## giant	NA	NA	NA	NA
## small	NA	NA	NA	NA
## north	NA	NA	NA	NA
## conserv	NA	NA	NA	NA
## effect	NA	NA	NA	NA
## watch	NA	NA	NA	NA
## gave	NA	NA	NA	NA
## young	NA	NA	NA	NA

## releas	NA	NA	NA	NA
## guy	NA	NA	NA	NA
## X12	NA	NA	NA	NA
## justic	NA	NA	NA	NA
## colleg	NA	NA	NA	NA
## death	NA	NA	NA	NA
## lawyer	NA	NA	NA	NA
## water	NA	NA	NA	NA
## power	NA	NA	NA	NA
## administr	NA	NA	NA	NA
## low	NA	NA	NA	NA
## sale	NA	NA	NA	NA
## charg	NA	NA	NA	NA
## X50	NA	NA	NA	NA
## stori	NA	NA	NA	NA
## iraqi	NA	NA	NA	NA
## clear	NA	NA	NA	NA
## georg	NA	NA	NA	NA
## black	NA	NA	NA	NA
## advertis	NA	NA	NA	NA
## X7	NA	NA	NA	NA
## X8	NA	NA	NA	NA
## women	NA	NA	NA	NA
## carri	NA	NA	NA	NA
## town	NA	NA	NA	NA
## success	NA	NA	NA	NA
## thoma	NA	NA	NA	NA
## suggest	NA	NA	NA	NA
## minut	NA	NA	NA	NA
## board	NA	NA	NA	NA
## stop	NA	NA	NA	NA
## defens	NA	NA	NA	NA
## posit	NA	NA	NA	NA
## decid	NA	NA	NA	NA
## X9	NA	NA	NA	NA
## leav	NA	NA	NA	NA
## commun	NA	NA	NA	NA
## studi	NA	NA	NA	NA
## meet	NA	NA	NA	NA
## die	NA	NA	NA	NA
## visit	NA	NA	NA	NA
## shop	NA	NA	NA	NA
## spent	NA	NA	NA	NA
## direct	NA	NA	NA	NA
## free	NA	NA	NA	NA
## lose	NA	NA	NA	NA
## note	NA	NA	NA	NA
## incumb	NA	NA	NA	NA
## involv	NA	NA	NA	NA
## media	NA	NA	NA	NA
## front	NA	NA	NA	NA
## form	NA	NA	NA	NA
## wait	NA	NA	NA	NA
## claim	NA	NA	NA	NA

## yanke	NA	NA	NA	NA
## today	NA	NA	NA	NA
## author	NA	NA	NA	NA
## didn	NA	NA	NA	NA
## great	NA	NA	NA	NA
## fight	NA	NA	NA	NA
## serv	NA	NA	NA	NA
## senior	NA	NA	NA	NA
## cast	NA	NA	NA	NA
## economi	NA	NA	NA	NA
## began	NA	NA	NA	NA
## gain	NA	NA	NA	NA
## idea	NA	NA	NA	NA
## pick	NA	NA	NA	NA
## morn	NA	NA	NA	NA
## comput	NA	NA	NA	NA
## approv	NA	NA	NA	NA
## X15	NA	NA	NA	NA
## militari	NA	NA	NA	NA
## aid	NA	NA	NA	NA
## turnout	NA	NA	NA	NA
## california	NA	NA	NA	NA
## store	NA	NA	NA	NA
## chairman	NA	NA	NA	NA
## figur	NA	NA	NA	NA
## X2003	NA	NA	NA	NA
## reach	NA	NA	NA	NA
## hold	NA	NA	NA	NA
## east	NA	NA	NA	NA
## measur	NA	NA	NA	NA
## begin	NA	NA	NA	NA
## sox	NA	NA	NA	NA
## edward	NA	NA	NA	NA
## japan	NA	NA	NA	NA
## rate	NA	NA	NA	NA
## evid	NA	NA	NA	NA
## real	NA	NA	NA	NA
## elector	NA	NA	NA	NA
## X14	NA	NA	NA	NA
## cancer	NA	NA	NA	NA
## spend	NA	NA	NA	NA
## view	NA	NA	NA	NA
## committe	NA	NA	NA	NA
## histori	NA	NA	NA	NA
## chanc	NA	NA	NA	NA
## addit	NA	NA	NA	NA
## wife	NA	NA	NA	NA
## travel	NA	NA	NA	NA
## bring	NA	NA	NA	NA
## muslim	NA	NA	NA	NA
## analyst	NA	NA	NA	NA
## X2002	NA	NA	NA	NA
## regist	NA	NA	NA	NA
## judg	NA	NA	NA	NA

## stand	NA	NA	NA	NA
## friday	NA	NA	NA	NA
## fox	NA	NA	NA	NA
## data	NA	NA	NA	NA
## full	NA	NA	NA	NA
## sell	NA	NA	NA	NA
## survey	NA	NA	NA	NA
## decad	NA	NA	NA	NA
## strong	NA	NA	NA	NA
## write	NA	NA	NA	NA
## finish	NA	NA	NA	NA
## popular	NA	NA	NA	NA
## bad	NA	NA	NA	NA
## averag	NA	NA	NA	NA
## declin	NA	NA	NA	NA
## train	NA	NA	NA	NA
## messag	NA	NA	NA	NA
## insur	NA	NA	NA	NA
## protect	NA	NA	NA	NA
## requir	NA	NA	NA	NA
## creat	NA	NA	NA	NA
## machin	NA	NA	NA	NA
## woman	NA	NA	NA	NA
## global	NA	NA	NA	NA
## loss	NA	NA	NA	NA
## estim	NA	NA	NA	NA
## spokesman	NA	NA	NA	NA
## grow	NA	NA	NA	NA
## expert	NA	NA	NA	NA
## thought	NA	NA	NA	NA
## complet	NA	NA	NA	NA
## design	NA	NA	NA	NA
## respons	NA	NA	NA	NA
## love	NA	NA	NA	NA
## custom	NA	NA	NA	NA
## independ	NA	NA	NA	NA
## car	NA	NA	NA	NA
## friend	NA	NA	NA	NA
## star	NA	NA	NA	NA
## mail	NA	NA	NA	NA
## oppon	NA	NA	NA	NA
## account	NA	NA	NA	NA
## texa	NA	NA	NA	NA
## vice	NA	NA	NA	NA
## prison	NA	NA	NA	NA
## fact	NA	NA	NA	NA
## resid	NA	NA	NA	NA
## corpor	NA	NA	NA	NA
## basebal	NA	NA	NA	NA
## seek	NA	NA	NA	NA
## foreign	NA	NA	NA	NA
## patient	NA	NA	NA	NA
## announc	NA	NA	NA	NA
## david	NA	NA	NA	NA

## earlier	NA	NA	NA	NA
## order	NA	NA	NA	NA
## appeal	NA	NA	NA	NA
## futur	NA	NA	NA	NA
## X18	NA	NA	NA	NA
## stock	NA	NA	NA	NA
## god	NA	NA	NA	NA
## flight	NA	NA	NA	NA
## marriag	NA	NA	NA	NA
## cover	NA	NA	NA	NA
## church	NA	NA	NA	NA
## air	NA	NA	NA	NA
## read	NA	NA	NA	NA
## main	NA	NA	NA	NA
## improv	NA	NA	NA	NA
## mart	NA	NA	NA	NA
## dollar	NA	NA	NA	NA
## X25	NA	NA	NA	NA
## press	NA	NA	NA	NA
## fail	NA	NA	NA	NA
## competit	NA	NA	NA	NA
## benefit	NA	NA	NA	NA
## art	NA	NA	NA	NA
## cut	NA	NA	NA	NA
## guard	NA	NA	NA	NA
## approach	NA	NA	NA	NA
## demand	NA	NA	NA	NA
## digit	NA	NA	NA	NA
## dozen	NA	NA	NA	NA
## crowd	NA	NA	NA	NA
## san	NA	NA	NA	NA
## richard	NA	NA	NA	NA
## learn	NA	NA	NA	NA
## stage	NA	NA	NA	NA
## chines	NA	NA	NA	NA
## confer	NA	NA	NA	NA
## retir	NA	NA	NA	NA
## medic	NA	NA	NA	NA
## airlin	NA	NA	NA	NA
## articl	NA	NA	NA	NA
## agre	NA	NA	NA	NA
## telephon	NA	NA	NA	NA
## financi	NA	NA	NA	NA
## mark	NA	NA	NA	NA
## margin	NA	NA	NA	NA
## beat	NA	NA	NA	NA
## phone	NA	NA	NA	NA
## hear	NA	NA	NA	NA
## hit	NA	NA	NA	NA
## schedul	NA	NA	NA	NA
## father	NA	NA	NA	NA
## X100	NA	NA	NA	NA
## kid	NA	NA	NA	NA
## flu	NA	NA	NA	NA

## file	NA	NA	NA	NA
## ran	NA	NA	NA	NA
## wide	NA	NA	NA	NA
## invest	NA	NA	NA	NA
## push	NA	NA	NA	NA
## activ	NA	NA	NA	NA
## cheney	NA	NA	NA	NA
## action	NA	NA	NA	NA
## ball	NA	NA	NA	NA
## prepar	NA	NA	NA	NA
## drop	NA	NA	NA	NA
## miss	NA	NA	NA	NA
## contest	NA	NA	NA	NA
## room	NA	NA	NA	NA
## mile	NA	NA	NA	NA
## retail	NA	NA	NA	NA
## X60	NA	NA	NA	NA
## class	NA	NA	NA	NA
## miami	NA	NA	NA	NA
## relat	NA	NA	NA	NA
## elgindi	NA	NA	NA	NA
## web	NA	NA	NA	NA
## post	NA	NA	NA	NA
## test	NA	NA	NA	NA
## angel	NA	NA	NA	NA
## publish	NA	NA	NA	NA
## michael	NA	NA	NA	NA
## X24	NA	NA	NA	NA
## moment	NA	NA	NA	NA
## conduct	NA	NA	NA	NA
## level	NA	NA	NA	NA
## short	NA	NA	NA	NA
## largest	NA	NA	NA	NA
## wednesday	NA	NA	NA	NA
## stadium	NA	NA	NA	NA
## replac	NA	NA	NA	NA
## econom	NA	NA	NA	NA
## firm	NA	NA	NA	NA
## age	NA	NA	NA	NA
## ga	NA	NA	NA	NA
## X40	NA	NA	NA	NA
## collect	NA	NA	NA	NA
## featur	NA	NA	NA	NA
## randolph	NA	NA	NA	NA
## biggest	NA	NA	NA	NA
## capit	NA	NA	NA	NA
## region	NA	NA	NA	NA
## legal	NA	NA	NA	NA
## appli	NA	NA	NA	NA
## contribut	NA	NA	NA	NA
## investig	NA	NA	NA	NA
## fire	NA	NA	NA	NA
## argu	NA	NA	NA	NA
## sport	NA	NA	NA	NA

## imag	NA	NA	NA	NA
## step	NA	NA	NA	NA
## advis	NA	NA	NA	NA
## fill	NA	NA	NA	NA
## debat	NA	NA	NA	NA
## experi	NA	NA	NA	NA
## japanes	NA	NA	NA	NA
## saturday	NA	NA	NA	NA
## swing	NA	NA	NA	NA
## review	NA	NA	NA	NA
## minor	NA	NA	NA	NA
## goal	NA	NA	NA	NA
## professor	NA	NA	NA	NA
## vike	NA	NA	NA	NA
## drug	NA	NA	NA	NA
## neal	NA	NA	NA	NA
## total	NA	NA	NA	NA
## wal	NA	NA	NA	NA
## theater	NA	NA	NA	NA
## block	NA	NA	NA	NA
## page	NA	NA	NA	NA
## X16	NA	NA	NA	NA
## energi	NA	NA	NA	NA
## fan	NA	NA	NA	NA
## missionari	NA	NA	NA	NA
## enter	NA	NA	NA	NA
## opera	NA	NA	NA	NA
## battl	NA	NA	NA	NA
## emerg	NA	NA	NA	NA
## son	NA	NA	NA	NA
## drive	NA	NA	NA	NA
## act	NA	NA	NA	NA
## opposit	NA	NA	NA	NA
## space	NA	NA	NA	NA
## promis	NA	NA	NA	NA
## mind	NA	NA	NA	NA
## X13	NA	NA	NA	NA
## ll	NA	NA	NA	NA
## minnesota	NA	NA	NA	NA
## oct	NA	NA	NA	NA
## social	NA	NA	NA	NA
## tie	NA	NA	NA	NA
## fourth	NA	NA	NA	NA
## determin	NA	NA	NA	NA
## regul	NA	NA	NA	NA
## sens	NA	NA	NA	NA
## huge	NA	NA	NA	NA
## clinton	NA	NA	NA	NA
## newspap	NA	NA	NA	NA
## comment	NA	NA	NA	NA
## central	NA	NA	NA	NA
## pictur	NA	NA	NA	NA
## winner	NA	NA	NA	NA
## tradit	NA	NA	NA	NA

## propos	NA	NA	NA	NA
## earn	NA	NA	NA	NA
## minist	NA	NA	NA	NA
## X17	NA	NA	NA	NA
## squar	NA	NA	NA	NA
## track	NA	NA	NA	NA
## shot	NA	NA	NA	NA
## audienc	NA	NA	NA	NA
## staff	NA	NA	NA	NA
## buy	NA	NA	NA	NA
## quickli	NA	NA	NA	NA
## period	NA	NA	NA	NA
## hotel	NA	NA	NA	NA
## compar	NA	NA	NA	NA
## fear	NA	NA	NA	NA
## difficult	NA	NA	NA	NA
## kidd	NA	NA	NA	NA
## refer	NA	NA	NA	NA
## philadelphia	NA	NA	NA	NA
## depart	NA	NA	NA	NA
## footbal	NA	NA	NA	NA
## disput	NA	NA	NA	NA
## palestinian	NA	NA	NA	NA
## speak	NA	NA	NA	NA
## stay	NA	NA	NA	NA
## similar	NA	NA	NA	NA
## neighborhood	NA	NA	NA	NA
## statement	NA	NA	NA	NA
## intellig	NA	NA	NA	NA
## join	NA	NA	NA	NA
## ventur	NA	NA	NA	NA
## profit	NA	NA	NA	NA
## X500	NA	NA	NA	NA
## potenti	NA	NA	NA	NA
## divis	NA	NA	NA	NA
## favor	NA	NA	NA	NA
## present	NA	NA	NA	NA
## injur	NA	NA	NA	NA
## prevent	NA	NA	NA	NA
## screen	NA	NA	NA	NA
## hundr	NA	NA	NA	NA
## jewish	NA	NA	NA	NA
## owner	NA	NA	NA	NA
## korean	NA	NA	NA	NA
## arrest	NA	NA	NA	NA
## insid	NA	NA	NA	NA
## signal	NA	NA	NA	NA
## institut	NA	NA	NA	NA
## limit	NA	NA	NA	NA
## accus	NA	NA	NA	NA
## net	NA	NA	NA	NA
## type	NA	NA	NA	NA
## agent	NA	NA	NA	NA
## understand	NA	NA	NA	NA

## ralli	NA	NA	NA	NA
## precinct	NA	NA	NA	NA
## consum	NA	NA	NA	NA
## nov	NA	NA	NA	NA
## event	NA	NA	NA	NA
## titl	NA	NA	NA	NA
## promot	NA	NA	NA	NA
## lake	NA	NA	NA	NA
## christian	NA	NA	NA	NA
## commiss	NA	NA	NA	NA
## hospit	NA	NA	NA	NA
## summer	NA	NA	NA	NA
## chicago	NA	NA	NA	NA
## fund	NA	NA	NA	NA
## break.	NA	NA	NA	NA
## septemb	NA	NA	NA	NA
## coverag	NA	NA	NA	NA
## realiti	NA	NA	NA	NA
## terror	NA	NA	NA	NA
## reduc	NA	NA	NA	NA
## food	NA	NA	NA	NA
## doesn	NA	NA	NA	NA
## peterson	NA	NA	NA	NA
## explain	NA	NA	NA	NA
## human	NA	NA	NA	NA
## predict	NA	NA	NA	NA
## easili	NA	NA	NA	NA
## suprem	NA	NA	NA	NA
## gay	NA	NA	NA	NA
## rank	NA	NA	NA	NA
## western	NA	NA	NA	NA
## touchdown	NA	NA	NA	NA
## brother	NA	NA	NA	NA
## lower	NA	NA	NA	NA
## attent	NA	NA	NA	NA
## arriv	NA	NA	NA	NA
## respond	NA	NA	NA	NA
## send	NA	NA	NA	NA
## trial	NA	NA	NA	NA
## offens	NA	NA	NA	NA
## aggress	NA	NA	NA	NA
## cleveland	NA	NA	NA	NA
## contend	NA	NA	NA	NA
## broadcast	NA	NA	NA	NA
## X19	NA	NA	NA	NA
## parent	NA	NA	NA	NA
## speech	NA	NA	NA	NA
## park	NA	NA	NA	NA
## iowa	NA	NA	NA	NA
## congression	NA	NA	NA	NA
## marathon	NA	NA	NA	NA
## pennsylvania	NA	NA	NA	NA
## threat	NA	NA	NA	NA
## surpris	NA	NA	NA	NA

## arafat	NA	NA	NA	NA
## wrong	NA	NA	NA	NA
## confid	NA	NA	NA	NA
## artist	NA	NA	NA	NA
## door	NA	NA	NA	NA
## land	NA	NA	NA	NA
## word	NA	NA	NA	NA
## india	NA	NA	NA	NA
## prove	NA	NA	NA	NA
## club	NA	NA	NA	NA
## focu	NA	NA	NA	NA
## veteran	NA	NA	NA	NA
## ground	NA	NA	NA	NA
## estat	NA	NA	NA	NA
## match	NA	NA	NA	NA
## william	NA	NA	NA	NA
## singl	NA	NA	NA	NA
## box	NA	NA	NA	NA
## johnson	NA	NA	NA	NA
## prime	NA	NA	NA	NA
## paid	NA	NA	NA	NA
## cultur	NA	NA	NA	NA
## common	NA	NA	NA	NA
## milit	NA	NA	NA	NA
## rove	NA	NA	NA	NA
## absente	NA	NA	NA	NA
## sept	NA	NA	NA	NA
## individu	NA	NA	NA	NA
## wrote	NA	NA	NA	NA
## growth	NA	NA	NA	NA
## student	NA	NA	NA	NA
## english	NA	NA	NA	NA
## pro	NA	NA	NA	NA
## jim	NA	NA	NA	NA
## consult	NA	NA	NA	NA
## knick	NA	NA	NA	NA
## cb	NA	NA	NA	NA
## paul	NA	NA	NA	NA
## warn	NA	NA	NA	NA
## negoti	NA	NA	NA	NA
## subject	NA	NA	NA	NA
## oppos	NA	NA	NA	NA
## suffer	NA	NA	NA	NA
## latest	NA	NA	NA	NA
## attract	NA	NA	NA	NA
## check	NA	NA	NA	NA
## villag	NA	NA	NA	NA
## river	NA	NA	NA	NA
## troubl	NA	NA	NA	NA
## condit	NA	NA	NA	NA
## rest	NA	NA	NA	NA
## process	NA	NA	NA	NA
## educ	NA	NA	NA	NA
## natur	NA	NA	NA	NA

## doubt	NA	NA	NA	NA
## novemb	NA	NA	NA	NA
## heart	NA	NA	NA	NA
## size	NA	NA	NA	NA
## hire	NA	NA	NA	NA
## manhattan	NA	NA	NA	NA
## photograph	NA	NA	NA	NA
## fell	NA	NA	NA	NA
## initi	NA	NA	NA	NA
## gore	NA	NA	NA	NA
## airport	NA	NA	NA	NA
## vaccin	NA	NA	NA	NA
## simpli	NA	NA	NA	NA
## wage	NA	NA	NA	NA
## insurg	NA	NA	NA	NA
## tom	NA	NA	NA	NA
## focus	NA	NA	NA	NA
## cloth	NA	NA	NA	NA
## wall	NA	NA	NA	NA
## centuri	NA	NA	NA	NA
## runner	NA	NA	NA	NA
## knew	NA	NA	NA	NA
## carolina	NA	NA	NA	NA
## constitut	NA	NA	NA	NA
## station	NA	NA	NA	NA
## vehicl	NA	NA	NA	NA
## incom	NA	NA	NA	NA
## repeat.	NA	NA	NA	NA
## green	NA	NA	NA	NA
## standard	NA	NA	NA	NA
## revenu	NA	NA	NA	NA
## middl	NA	NA	NA	NA
## engin	NA	NA	NA	NA
## ahead	NA	NA	NA	NA
## accept	NA	NA	NA	NA
## prosecutor	NA	NA	NA	NA
## draw	NA	NA	NA	NA
## fuel	NA	NA	NA	NA
## editor	NA	NA	NA	NA
## felt	NA	NA	NA	NA
## daili	NA	NA	NA	NA
## diseas	NA	NA	NA	NA
## sound	NA	NA	NA	NA
## island	NA	NA	NA	NA
## eye	NA	NA	NA	NA
## explor	NA	NA	NA	NA
## mother	NA	NA	NA	NA
## championship	NA	NA	NA	NA
## thursday	NA	NA	NA	NA
## joe	NA	NA	NA	NA
## employ	NA	NA	NA	NA
## spot	NA	NA	NA	NA
## opinion	NA	NA	NA	NA
## movi	NA	NA	NA	NA

## passeng	NA	NA	NA	NA
## avoid	NA	NA	NA	NA
## demonstr	NA	NA	NA	NA
## ill	NA	NA	NA	NA
## jone	NA	NA	NA	NA
## special	NA	NA	NA	NA
## francisco	NA	NA	NA	NA
## dead	NA	NA	NA	NA
## mill	NA	NA	NA	NA
## road	NA	NA	NA	NA
## bodi	NA	NA	NA	NA
## lo	NA	NA	NA	NA
## search	NA	NA	NA	NA
## practic	NA	NA	NA	NA
## coupl	NA	NA	NA	NA
## israel	NA	NA	NA	NA
## style	NA	NA	NA	NA
## brought	NA	NA	NA	NA
## spoke	NA	NA	NA	NA
## internet	NA	NA	NA	NA
## convent	NA	NA	NA	NA
## struggl	NA	NA	NA	NA
## spread	NA	NA	NA	NA
## heard	NA	NA	NA	NA
## hsbc	NA	NA	NA	NA
## baghdad	NA	NA	NA	NA
## treat	NA	NA	NA	NA
## assist	NA	NA	NA	NA
## X21	NA	NA	NA	NA
## X22	NA	NA	NA	NA
## complain	NA	NA	NA	NA
## shape	NA	NA	NA	NA
## X2001	NA	NA	NA	NA
## brand	NA	NA	NA	NA
## risk	NA	NA	NA	NA
## rise	NA	NA	NA	NA
## blue	NA	NA	NA	NA
## choic	NA	NA	NA	NA
## declar	NA	NA	NA	NA
## origin	NA	NA	NA	NA
## amend	NA	NA	NA	NA
## moder	NA	NA	NA	NA
## britain	NA	NA	NA	NA
## larger	NA	NA	NA	NA
## walk	NA	NA	NA	NA
## mike	NA	NA	NA	NA
## role	NA	NA	NA	NA
## address	NA	NA	NA	NA
## argument	NA	NA	NA	NA
## introduc	NA	NA	NA	NA
## paper	NA	NA	NA	NA
## england	NA	NA	NA	NA
## sex	NA	NA	NA	NA
## de	NA	NA	NA	NA

## born	NA	NA	NA	NA
## paint	NA	NA	NA	NA
## jame	NA	NA	NA	NA
## welfar	NA	NA	NA	NA
## round	NA	NA	NA	NA
## rock	NA	NA	NA	NA
## correct	NA	NA	NA	NA
## channel	NA	NA	NA	NA
## connect	NA	NA	NA	NA
## surfac	NA	NA	NA	NA
## rush	NA	NA	NA	NA
## radio	NA	NA	NA	NA
## violenc	NA	NA	NA	NA
## feet	NA	NA	NA	NA
## elig	NA	NA	NA	NA
## discuss	NA	NA	NA	NA
## peoplesoft	NA	NA	NA	NA
## canyon	NA	NA	NA	NA
## cite	NA	NA	NA	NA
## neighbor	NA	NA	NA	NA
## factor	NA	NA	NA	NA
## touch	NA	NA	NA	NA
## suit	NA	NA	NA	NA
## houston	NA	NA	NA	NA
## oracl	NA	NA	NA	NA
## percentag	NA	NA	NA	NA
## gun	NA	NA	NA	NA
## hall	NA	NA	NA	NA
## treatment	NA	NA	NA	NA
## injuri	NA	NA	NA	NA
## key	NA	NA	NA	NA
## starbuck	NA	NA	NA	NA
## bob	NA	NA	NA	NA
## girl	NA	NA	NA	NA
## fall	NA	NA	NA	NA
## liber	NA	NA	NA	NA
## express	NA	NA	NA	NA
## add	NA	NA	NA	NA
## insist	NA	NA	NA	NA
## virginia	NA	NA	NA	NA
## scientist	NA	NA	NA	NA
## advanc	NA	NA	NA	NA
## strike	NA	NA	NA	NA
## schumer	NA	NA	NA	NA
## plane	NA	NA	NA	NA
## secretari	NA	NA	NA	NA
## magazin	NA	NA	NA	NA
## husband	NA	NA	NA	NA
## king	NA	NA	NA	NA
## increasingli	NA	NA	NA	NA
## martin	NA	NA	NA	NA
## attempt	NA	NA	NA	NA
## situat	NA	NA	NA	NA
## troop	NA	NA	NA	NA

## athlet	NA	NA	NA	NA
## monitor	NA	NA	NA	NA
## doctor	NA	NA	NA	NA
## built	NA	NA	NA	NA
## jason	NA	NA	NA	NA
## wade	NA	NA	NA	NA
## defend	NA	NA	NA	NA
## journalist	NA	NA	NA	NA
## circul	NA	NA	NA	NA
## client	NA	NA	NA	NA
## civil	NA	NA	NA	NA
## ticket	NA	NA	NA	NA
## weight	NA	NA	NA	NA
## X80	NA	NA	NA	NA
## heavili	NA	NA	NA	NA
## appar	NA	NA	NA	NA
## intens	NA	NA	NA	NA
## chain	NA	NA	NA	NA
## pull	NA	NA	NA	NA
## congress	NA	NA	NA	NA
## agreement	NA	NA	NA	NA
## garden	NA	NA	NA	NA
## video	NA	NA	NA	NA
## meant	NA	NA	NA	NA
## expand	NA	NA	NA	NA
## captur	NA	NA	NA	NA
## domin	NA	NA	NA	NA
## arab	NA	NA	NA	NA
## provision	NA	NA	NA	NA
## caught	NA	NA	NA	NA
## robert	NA	NA	NA	NA
## sheik	NA	NA	NA	NA
## armi	NA	NA	NA	NA
## answer	NA	NA	NA	NA
## marin	NA	NA	NA	NA
## mexico	NA	NA	NA	NA
## commit	NA	NA	NA	NA
## classic	NA	NA	NA	NA
## shift	NA	NA	NA	NA
## faith	NA	NA	NA	NA
## previou	NA	NA	NA	NA
## annual	NA	NA	NA	NA
## forward	NA	NA	NA	NA
## physic	NA	NA	NA	NA
## mix	NA	NA	NA	NA
## X45	NA	NA	NA	NA
## basketbal	NA	NA	NA	NA
## dolphin	NA	NA	NA	NA
## combin	NA	NA	NA	NA
## refus	NA	NA	NA	NA
## wisconsin	NA	NA	NA	NA
## sold	NA	NA	NA	NA
## van	NA	NA	NA	NA
## remind	NA	NA	NA	NA

```

## roll          NA          NA          NA          NA
## draft         NA          NA          NA          NA
## surviv        NA          NA          NA          NA
## expens        NA          NA          NA          NA
## acknowledgd   NA          NA          NA          NA
## scene         NA          NA          NA          NA
## mayor         NA          NA          NA          NA
## target        NA          NA          NA          NA
## contact       NA          NA          NA          NA
## lewi          NA          NA          NA          NA
## opportun      NA          NA          NA          NA
## request       NA          NA          NA          NA
## stake         NA          NA          NA          NA
## fraud         NA          NA          NA          NA
## poor          NA          NA          NA          NA
## foot          NA          NA          NA          NA
## pretti        NA          NA          NA          NA
## readi         NA          NA          NA          NA
## milwaukee     NA          NA          NA          NA
## handl         NA          NA          NA          NA
## career        NA          NA          NA          NA
## arnel         NA          NA          NA          NA
## voic          NA          NA          NA          NA
## identifi      NA          NA          NA          NA
## X0            NA          NA          NA          NA
## X23           NA          NA          NA          NA
## merger        NA          NA          NA          NA
##
## Residual standard error: NaN on 0 degrees of freedom
## Multiple R-squared:      1, Adjusted R-squared:      NaN
## F-statistic:      NaN on 199 and 0 DF, p-value: NA

```

```

# Count the number of coefficients dropped from the model
length(which(is.na(full_reg$coeff)==T))

```

```
## [1] 801
```

Question 3. Evaluating the OLS performance

We are now going to make predictions using the training data and the test data and compare their properties.

Using the `predict` function, make predictions for all observations in the training set. Then, classify the documents as national or not using a threshold of 0.5. Assess your classification to the actual data. Create a 2x2 table of the predicted train labels and true train label and note your findings.

```
## your code here
```

Next, use the model to make a prediction for the `test` data and classify using a 0.5 threshold.

Assess the accuracy of your classification by comparing it to the actual test data. What do you notice? What would happen if you randomly guessed the test labels using a prior on the probability of 1 as the proportion of 1s in the train labels? Remember to `set.seed(2018)`. Compare your findings between the two methods.

Answer 3

```
train_pred <- predict(full_reg, as.data.frame(train))
class_doc <- ifelse(train_pred > 0.5, 1, 0)
table(class_doc, train_label)
```

```
##           train_label
## class_doc  0     1
##           0 150   0
##           1   0  50
```

There's perfect in-sample training prediction.

```
# predict test observations
test_pred <- predict(full_reg, as.data.frame(test))
```

```
## Warning in predict.lm(full_reg, as.data.frame(test)): prediction from
## rank-deficient fit; attr(*, "non-estim") has doubtful cases
```

```
# classify using threshold of 0.5
class_pred <- ifelse(test_pred > 0.5, 1, 0)
```

```
# create confusion matrix
table(class_pred, test_label)
```

```
##           test_label
## class_pred  0     1
##           0  31 17
##           1  31   9
```

```
# Get accuracy score of our predictions
acc1 <- (sum(class_pred & test_label) + sum(!class_pred & !test_label)) / length(test_label) # or sum(class_pred == test_label) / length(test_label)
acc1
```

```
## [1] 0.4545455
```

```
set.seed(2018)
rand_guess <- rbinom(length(test_label),
                    prob = sum(train_label)/length(train_label), # the proportion of 1s in the train l
                    size = 1)
acc2 <- sum(diag(table(rand_guess, test_label)))/length(test_label)
acc2
```

```
## [1] 0.5909091
```

We have very low accuracy of 0.45. In fact we could just guess and do better: 0.59.

Let's turn to the LASSO instead, a regularization technique we just learned in theory and which we'll put in practice now.

Question 4. Fit LASSO regression

We are going to use the `glmnet` library to fit the LASSO regression. Load the package.

The syntax for the `glmnet` model is as follows: `lasso <- glmnet(x = train, y = train_label)`

This defaults to linear regression. To do logistic regression you can fit the same model, but add `lasso_logist <- glmnet(x = train, y = train_label, family = 'binomial')`

Fit a LASSO linear regression.

```
#install.packages('glmnet')
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-8
```

```
set.seed(2018)
```

```
## your code here
```

The LASSO function automatically fits the model for several values of λ , and produces β values for all covariates for each value of λ all of which is found in the object `lasso$beta`.

Sum up the absolute values of `lasso$beta` for each column. Plot that against `lasso$lambda`. What generally happens as λ increases?

```
## your code here
```

There are different methods to selecting lambda, which we set aside for another day. Today, we're going to set a particular value of lambda arbitrarily and then assess its performance. We will set lambda to 0.05.

Formulate predictions for the training set using the following syntax: `lasso_pred <- predict(lasso, newx=train, s = 0.05)`

- `lasso` is the lasso regression
- `newx` are the values you want to predict
- `s` is the value of lambda.

Classify the observations using a threshold of 0.5. Then assess the accuracy of those predictions by comparing them to the training set labels and create a confusion matrix. Do the same but use a threshold of prior information on the training set – the proportion of 1s. Which threshold is better?

```
## your code here
```

Answer 4

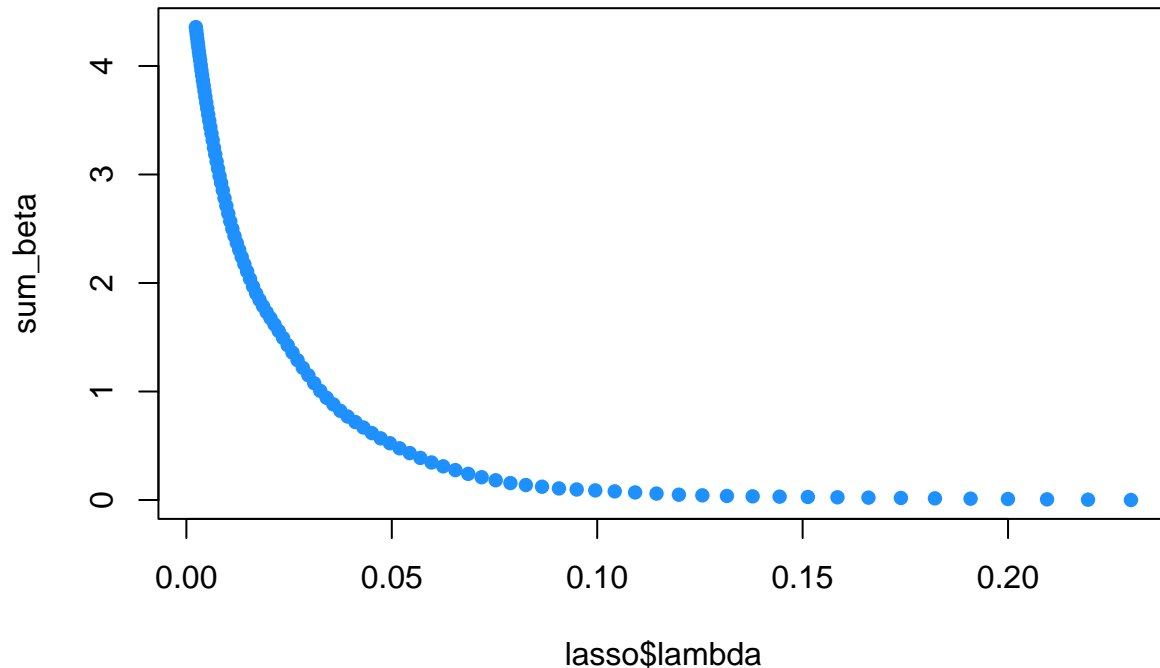
```
#install.packages('glmnet')
library(glmnet)
set.seed(2018)
lasso <- glmnet(x = train, y = train_label)
```

```

# sum absolute values of the betas
sum_beta <- colSums(abs(lasso$beta))

# plot against values of lambda
plot(sum_beta ~ lasso$lambda, pch=16, col="dodgerblue1")

```



As λ increases, the number of β coefficients (and hence the sum) gets pushed down to 0, increasing regularization.

Question 5. Evaluating the performance of LASSO

There are different methods to selecting lambda, which we set aside for another day. Today, we're going to set a particular value of lambda arbitrarily and then assess its performance. We will set lambda to 0.05.

Formulate predictions for the training set using the following syntax: `lasso_pred <- predict(lasso, newx=train, s = 0.05)`

- `lasso` is the lasso regression
- `newx` are the values you want to predict
- `s` is the value of lambda.

Classify the observations using a threshold of 0.5. Then assess the accuracy of those predictions by comparing them to the training set labels and create a confusion matrix. Do the same but use a threshold of prior information on the training set – the proportion of 1s. Which threshold is better?


```
## your code here
```

Finally, let's formulate predictions for the test set, classify the documents as national or not with a threshold using the prior proportion of 1 labels in the training set as well as 0.5, and assess the accuracy of those predictions by comparing them to the test set labels. What do you notice about the quality of the predictions from LASSO relative to the predictions from OLS?

Answer 5

```
# formulate predictions
lasso_pred <- predict(lasso, newx=train, s = 0.05 )

# classify obs using threshold of 0.5
class_lasso1 <- ifelse(lasso_pred>0.5, 1, 0)

# create confusion matrix
table(class_lasso1, train_label)
```

```
##           train_label
## class_lasso1  0  1
##              0 149 18
##              1  1 32
```

```
# get accuracy score
(sum(class_lasso1 & train_label) + sum(!class_lasso1 & !train_label)) / length(train_label)
```

```
## [1] 0.905
```

```
# classify obs using threshold with prior
class_lasso2 <- ifelse(lasso_pred>sum(train_label)/length(train_label), 1, 0)

# create confusion matrix
table(class_lasso2, train_label)
```

```
##           train_label
## class_lasso2  0  1
##              0 137  4
##              1  13 46
```

```
# get accuracy score
(sum(class_lasso2 & train_label) + sum(!class_lasso2 & !train_label)) / length(train_label)
```

```
## [1] 0.915
```

Notice that we don't have the perfect in-sample fit that we had before. This is a good thing – it means that we're not overfitting our data. Using prior information from our training data helped us here in boosting the in-sample correct prediction rate. However, clearly threshold matters.

```

# formulate predictions
lasso_test <- predict(lasso, newx=test, s = 0.05)

# classify obs using threshold of prior info
class_lasso_test <- ifelse(lasso_test>sum(train_label)/length(train_label), 1, 0)

# create confusion matrix
table(class_lasso_test, test_label)

##           test_label
## class_lasso_test 0  1
##                0 53  6
##                1  9 20

# get accuracy score
acc3 <- (sum(class_lasso_test & test_label) + sum(!class_lasso_test & !test_label)) / length(test_label)
acc3

## [1] 0.8295455

# classify obs using threshold of 0.5
class_lasso_test2 <- ifelse(lasso_test>sum(train_label)/length(train_label), 1, 0)

# create confusion matrix
table(class_lasso_test2, test_label)

##           test_label
## class_lasso_test2 0  1
##                0 53  6
##                1  9 20

# get accuracy score
acc4<-(sum(class_lasso_test2 & test_label) + sum(!class_lasso_test2 & !test_label)) / length(test_label)
acc4

## [1] 0.8295455

```

We have much better out of sample accuracy of 0.83 (but notice lower than our in sample accuracy).

Part 2: Post Double Selection

In this lab exercise, we will conduct simulation study of post double selection from Belloni, Chernozhukov, and Hansen (2014). Using a synthetic data, we will investigate the following (partially) linear model with high-dimensional covariates.

$$\begin{aligned}
 Y_i &= D_i \alpha + X_i^c \beta + X_i^y \delta + \epsilon_{1i} \\
 D_i &= X_i^c \gamma + X_i^e \sigma + \epsilon_{2i}
 \end{aligned}$$

where

- Y_i is the outcome variable (n by 1 vector);
- D_i is the policy (treatment) variable (n by 1 vector);
- X_i^c is a set of confounders we would like to control (n by p_c matrix);
- X_i^y is a set of covariates that only has an impact on Y_i (n by p_y matrix);
- X_i^e is a set of exogenous variables that only affects D_i (n by p_e matrix);
- $\mathbb{E}[\epsilon_{1i} | D_i, X_i^c, X_i^y] = 0$ and $\mathbb{E}[\epsilon_{2i} | X_i^c, X_i^e] = 0$.

Under the assumption of (approximate) sparsity and some regularity conditions, we will estimate α , the impact of D_i on Y_i . We have a high-dimensional covariates $X_i = X_i^c \cup X_i^y \cup X_i^e \cup X_i^u$ from which we may select X_i^c , X_i^y , and X_i^e . Note that X_i^u does not affect D_i nor Y_i ; we will use exact sparse regime instead of approximate sparse regime in the simulation study for simplicity. We will compare two different estimation strategies, one with usual post model selection and the other with post double selection. Please check the lecture slides and the paper for more details.

A note on Causal Inference: Note that we would need additional assumptions to interpret the regression coefficient α as an unbiased estimator for an Average Treatment Effect (ATE). In this question, we will ignore the issues with identification of the causal effect and focus on investigating the post double selection method.

Question 1: Creating synthetic data

We will start by creating a function that generates synthetic data given true parameters. The data generating process (DGP) we will use is as follows:

1. Generate $Z_{i1}^c, \dots, Z_{i,p_c}^c, Z_{i1}^y, \dots, Z_{i,p_y}^y, Z_{i1}^e, \dots, Z_{i,p_e}^e$, and $Z_{i1}^u, \dots, Z_{i,p}^u$ as follows:
 - $Z_{ik}^c \sim \mathcal{N}(0, 1)$ for $k = 1, \dots, p_c$
 - $Z_{ik}^y \sim \mathcal{N}(0, 1)$ for $k = 1, \dots, p_y$
 - $Z_{ik}^e \sim \mathcal{N}(0, 1)$ for $k = 1, \dots, p_e$
 - $Z_{ik}^u \sim \mathcal{N}(0, 1)$ for $k = 1, \dots, p_u$
2. Let X_i^c, X_i^y, X_i^e , and X_i^u be a function of $(Z_{i1}^c, \dots, Z_{i,p_c}^c)$, $(Z_{i1}^y, \dots, Z_{i,p_y}^y)$, $(Z_{i1}^e, \dots, Z_{i,p_e}^e)$, and $(Z_{i1}^u, \dots, Z_{i,p_u}^u)$ respectively as follows:
 - $X_i^c = (Z_{i1}^c, Z_{i2}^c, \dots, Z_{i,p_c}^c, Z_{i1}^{c^2}, Z_{i2}^{c^2}, \dots, Z_{i,p_c}^{c^2}, Z_{i1}^c \cdot Z_{i2}^c, Z_{i2}^c \cdot Z_{i3}^c, \dots, Z_{i,p_c-1}^c \cdot Z_{i,p_c}^c)$
 - $X_i^y = (Z_{i1}^y, Z_{i2}^y, \dots, Z_{i,p_y}^y, Z_{i1}^{y^2}, Z_{i2}^{y^2}, \dots, Z_{i,p_y}^{y^2}, Z_{i1}^y \cdot Z_{i2}^y, Z_{i2}^y \cdot Z_{i3}^y, \dots, Z_{i,p_y-1}^y \cdot Z_{i,p_y}^y)$
 - $X_i^e = (Z_{i1}^e, Z_{i2}^e, \dots, Z_{i,p_e}^e, Z_{i1}^{e^2}, Z_{i2}^{e^2}, \dots, Z_{i,p_e}^{e^2}, Z_{i1}^e \cdot Z_{i2}^e, Z_{i2}^e \cdot Z_{i3}^e, \dots, Z_{i,p_e-1}^e \cdot Z_{i,p_e}^e)$
 - $X_i^u = (Z_{i1}^u, Z_{i2}^u, \dots, Z_{i,p_u}^u, Z_{i1}^{u^2}, Z_{i2}^{u^2}, \dots, Z_{i,p_u}^{u^2}, Z_{i1}^u \cdot Z_{i2}^u, Z_{i2}^u \cdot Z_{i3}^u, \dots, Z_{i,p_u-1}^u \cdot Z_{i,p_u}^u)$
3. Let $X_i = (X_i^c \quad X_i^y \quad X_i^e \quad X_i^u)$
4. Generate regression coefficients as follows:
 - $\alpha = 0.5$
 - $\beta_j \sim \mathcal{N}(0.1, 0.01^2)$ for $j = 1, \dots, (3p_c - 1)$
 - $\delta_j \sim \mathcal{N}(0, 0.01^2)$ for $j = 1, \dots, (3p_y - 1)$
 - $\gamma_j \sim \mathcal{N}(0.8, 0.01^2)$ for $j = 1, \dots, (3p_e - 1)$
 - $\sigma_j \sim \mathcal{N}(0, 0.01^2)$ for $j = 1, \dots, (3p_e - 1)$

5. Generate random noise as follows:

- $\epsilon_{1i} \sim \mathcal{N}(0, 1)$ for $i = 1, \dots, n$
- $\epsilon_{2i} \sim \mathcal{N}(0, 1)$ for $i = 1, \dots, n$

6. Let $D_i = X_i^c \gamma + X_i^e \sigma + \epsilon_{2i}$ and $Y_i = D_i \alpha + X_i^y \beta + X_i^u \delta + \epsilon_{1i}$.

Using this DGP and the sample codes below, write a function `generate_synt_data` which generates the data and parameters.

```
generate_synt_data <- function(n = 100, # number of observations
                               pc = 5, # dimensionality (size of dimensions) of Z_i^c
                               py = 2, # dimensionality of Z_i^y
                               pe = 2, # dimensionality of Z_i^e
                               pu = 30, # dimensionality of Z_i^u
                               alpha = 0.5,
                               beta = NULL,
                               delta = NULL,
                               gamma = NULL,
                               sigma = NULL,
                               seed.number = 2018
                               ) {
  set.seed(seed.number)

  #-- TODO 1: Generate Z^c, Z^y, Z^e, and Z^u (Hint: use rnorm())
  Zc = matrix(NA, nrow = n, ncol = pc)
  Zy = matrix(NA, nrow = n, ncol = py)
  Ze = matrix(NA, nrow = n, ncol = pe)
  Zu = matrix(NA, nrow = n, ncol = pu)
  # Your code here

  #-- TODO 2: Create X^c, X^y, X^e, and X^u
  Xc = matrix(NA, nrow = n, ncol = pc*3-1)
  Xy = matrix(NA, nrow = n, ncol = py*3-1)
  Xe = matrix(NA, nrow = n, ncol = pe*3-1)
  Xu = matrix(NA, nrow = n, ncol = pu*3-1)
  # Your codes here
  colnames(Xc) = paste0("Xc", 1:(pc*3-1))
  colnames(Xy) = paste0("Xy", 1:(py*3-1))
  colnames(Xe) = paste0("Xe", 1:(pe*3-1))
  colnames(Xu) = paste0("Xu", 1:(pu*3-1))

  #-- 3: Create X
  X = cbind(Xc, Xy, Xe, Xu)

  #-- TODO 4: Generate regression coefficients
  if (is.null(alpha)) {
    alpha = rnorm(1, mean = 0, sd = 2) # example
  }
  if (is.null(beta)) {
    # Your code here
  }
  if (is.null(delta)) {
    # Your code here
  }
}
```

```

}
if (is.null(gamma)) {
  # Your code here
}
if (is.null(sigma)) {
  # Your code here
}

#-- TODO 5: Generate random noise
epsilon1 = rep(NA, n)
epsilon2 = rep(NA, n)
# Your code here

#-- TODO 6: Create D and Y
D = rep(NA, n)
Y = rep(NA, n)
# Your code here

colnames(D) = "D"
colnames(Y) = "Y"

# cat("## num of observations:",n,"\n## num of predictors:",ncol(X),"\n")

res = list(X = X, D = D, Y = Y,
           alpha = alpha, beta = beta, delta = delta, gamma = gamma, sigma = sigma)
return(res)
}

```

Answer 1

Comment Note that there are two ways to generate high-dimensional data as specified in the 2nd footnote of the original paper:

- The baseline set of conditioning variables itself may be large so $X_i = Z_i$
- Z_i may be low-dimensional, but one may wish to entertain many non-linear transformations of Z_i in forming $X_i = f(Z_i)$ as in traditional series-based estimation of the partially linear model.

Here, we're following the second way to generate our high-dimensional data.

```

generate_synth_data <- function(n = 100, # number of observations
                                pc = 5, # dimensionality (size of dimensions) of  $Z_i^c$ 
                                py = 2, # dimensionality of  $Z_i^y$ 
                                pe = 2, # dimensionality of  $Z_i^e$ 
                                pu = 30, # dimensionality of  $Z_i^u$ 
                                alpha = 0.5,
                                beta = NULL,
                                delta = NULL,
                                gamma = NULL,
                                sigma = NULL,

```

```

seed.number = 2018
) {
  set.seed(seed.number)

  ## 1: Generate Z^c, Z^y, Z^e, and Z^u (Hint: use rnorm())
  Zc = matrix(rnorm(n*pc, 0, 1), nrow = n, ncol = pc)
  Zy = matrix(rnorm(n*py, 0, 1), nrow = n, ncol = py)
  Ze = matrix(rnorm(n*pe, 0, 1), nrow = n, ncol = pe)
  Zu = matrix(rnorm(n*pu, 0, 1), nrow = n, ncol = pu)

  ## 2: Create X^c, X^y, X^e, and X^u
  Xc = cbind(Zc, Zc^2, Zc[,1:(pc-1)]*Zc[,2:pc])
  Xy = cbind(Zy, Zy^2, Zy[,1:(py-1)]*Zy[,2:py])
  Xe = cbind(Ze, Ze^2, Ze[,1:(pe-1)]*Ze[,2:pe])
  Xu = cbind(Zu, Zu^2, Zu[,1:(pu-1)]*Zu[,2:pu])

  colnames(Xc) = paste0("Xc", 1:(pc*3-1))
  colnames(Xy) = paste0("Xy", 1:(py*3-1))
  colnames(Xe) = paste0("Xe", 1:(pe*3-1))
  colnames(Xu) = paste0("Xu", 1:(pu*3-1))

  ## 3: Create X
  X = cbind(Xc, Xy, Xe, Xu)

  ## 4: Generate regression coefficients
  if (is.null(alpha)) {
    alpha = rnorm(1, mean = 0, sd = 2) # example
  }
  if (is.null(beta)) {
    beta = matrix(rnorm(3*pc-1, mean = 0.1, sd = 0.01), ncol = 1)
  }
  if (is.null(delta)) {
    delta = matrix(rnorm(3*py-1, mean = 0, sd = 0.01), ncol = 1)
  }
  if (is.null(gamma)) {
    gamma = matrix(rnorm(3*pc-1, mean = 0.8, sd = 0.01), ncol = 1)
  }
  if (is.null(sigma)) {
    sigma = matrix(rnorm(3*pe-1, mean = 0, sd = 0.01), ncol = 1)
  }

  ## 5: Generate random noise
  epsilon1 = rnorm(n, mean = 0, sd = 1)
  epsilon2 = rnorm(n, mean = 0, sd = 1)

  ## 6: Create D and Y
  D = Xc %*% gamma + Xe %*% sigma + epsilon2
  Y = D * alpha + Xc %*% beta + Xy %*% delta + epsilon1
  colnames(D) = "D"
  colnames(Y) = "Y"

  # cat("## num of observations:",n,"\n## num of predictors:",ncol(X),"\n")

```

```

res = list(X = X, D = D, Y = Y,
          alpha = alpha, beta = beta, delta = delta, gamma = gamma, sigma = sigma)
return(res)
}

```

Solution

Question 2: Lasso methods

Generate synthetic data using the function `generate_synth_data` with following arguments:

```

n = 100; pc = 5; py = 2; pe = 2; pu = 30
alpha = 0.5
set.seed(2018)
beta = matrix(rnorm(pc*3-1, mean = 0.1, sd = 0.01), ncol = 1)
delta = matrix(rnorm(py*3-1, mean = 0, sd = 0.01), ncol = 1)
gamma = matrix(rnorm(pc*3-1, mean = 0.8, sd = 0.01), ncol = 1)
sigma = matrix(rnorm(pe*3-1, mean = 0, sd = 0.01), ncol = 1)

```

Estimate α using `glmnet` with two different methods: post single lasso and post double lasso. For post double lasso, use two different packages for the estimation: `glmnet` and `hdm` (the latter is written by the authors of the paper). Briefly discuss the results.

Hint: For `glmnet`, use `cv.glmnet` in place of `glmnet` — this is a cross-validated `glmnet` — and `coef(your-glmnet-object, s = "lambda.1se")` functions to select λ value. For `hdm`, the syntax is as follows: `hdm::rlassoEffect(x=your-X,d=your-D,y=your-Y,method="double selection")` Note that the estimates may differ between `glmnet` and `hdm` due to different default methods of selecting for λ .

```

library(glmnet)
library(hdm)

## Your code here

```

Answer 2

Comment Observe that we're assuming a small effect of confounders on the outcome (`beta`) whereas a larger effect on the treatment (`gamma`) in this setup. This is intended to show you a clear contrast between post single selection and post double selection — since the former only focuses on capturing the effect of confounders on the outcome, it may mistakenly drop part of the true confounders that has a small effect on the outcome but a larger effect on the treatment. You can try other setups as well to test in which case post double selection might actually fail (e.g., in case we have a larger number of exogenous covariates that shouldn't be controlled in the outcome model.)

As you check `coef.label` and (`coef.y.label`, `coef.d.label`) in the following codes, you can see that single selection actually drop out these confounders whereas double selection still captures them (and sometimes the latter overestimates some of coefficients so that it ends up including unrelated variables).

```

## Generate synthetic data
dat <- generate_synth_data(alpha = alpha,

```

```

beta = beta,
gamma = gamma,
sigma = sigma,
delta = delta,
n = n,
pc = pc,
py = py,
pe = pe,
pu = pu,
seed.number = 2018)

library(glmnet)
library(hdm)

# Fit post lasso
## Using glmnet
# 1. Pick lambda using cv
lasso <- cv.glmnet(x = cbind(dat$X, dat$D), y = dat$Y)
# 2. Fit lasso and check non-zero coefficients
coef.lasso <- coef(lasso, s = "lambda.1se")
coef.label <- rownames(coef.lasso)[as.vector(!(coef.lasso == 0))]
coef.label <- coef.label[grepl("X.+?", coef.label)]
coef.label

```

Solution

```
## character(0)
```

```

# 3. Estimate alpha
dat.lasso = data.frame(Y = dat$Y, D = dat$D)
dat.lasso = cbind(dat.lasso, dat$X[,coef.label])
post.lasso <- lm("Y ~ .", data = dat.lasso)
post.lasso$coefficients["D"]

```

```
##          D
## 0.6002334
```

```

# Fit post double lasso
## Using glmnet
# 1. Selection of predictors for Y
y.lasso <- cv.glmnet(x = dat$X, y = dat$Y)
coef.y.lasso <- coef(y.lasso, s = "lambda.1se")
coef.y.label <- rownames(coef.y.lasso)[as.vector(!(coef.y.lasso == 0))]
coef.y.label <- coef.y.label[grepl("X.+?", coef.y.label)]
# 2. Selection of predictors for D
d.lasso <- cv.glmnet(x = dat$X, y = dat$D)
coef.d.lasso <- coef(d.lasso, s = "lambda.1se")
coef.d.label <- rownames(coef.d.lasso)[as.vector(!(coef.d.lasso == 0))]
coef.d.label <- coef.d.label[grepl("X.+?", coef.d.label)]
# 3. Refit the model
coef.double.label <- union(coef.y.label, coef.d.label)
coef.double.label

```

```
## [1] "Xc1" "Xc2" "Xc3" "Xc4" "Xc5" "Xc6" "Xc7" "Xc8" "Xc9" "Xc10"
```



```
## [11] "Xc11" "Xc12" "Xc13" "Xc14" "Xy1"  "Xu6"  "Xu10" "Xu17" "Xu21" "Xu36"
## [21] "Xu38" "Xu43" "Xu47" "Xu52" "Xu58" "Xu61" "Xu63" "Xu66" "Xu69" "Xu80"
## [31] "Xu82" "Xe1"  "Xe4"  "Xu12" "Xu14" "Xu18" "Xu20" "Xu34" "Xu45" "Xu50"
## [41] "Xu51" "Xu81" "Xu87"
```

```
coef.double <- dat$X[,coef.double.label]
dat.double <- data.frame(Y = dat$Y, D = dat$D)
dat.double <- cbind(dat.double, coef.double)
post.double.lasso <- lm("Y ~ .", data = dat.double)
post.double.lasso$coefficients["D"]
```

```
##          D
## 0.469523
```

```
## Using hdm
rlassoEffect(x=dat$X,d=dat$D,y=dat$Y,method="double selection")
```

```
##
## Call:
## rlassoEffect(x = dat$X, y = dat$Y, d = dat$D, method = "double selection")
##
## Coefficients:
##          D
## 0.6024
```

Question 3: Monte-Carlo simulation (3 pts)

Note that the result from Question 2 is based on a single simulation. Now, we will conduct a Monte-Carlo simulation where we generate synthetic data multiple times (e.g., 1000 iterations) and compare the distribution of α for each method. Plot the histogram of three distributions of α (single LASSO, double LASSO with `glmnet`, and double LASSO with `hdm`). Briefly discuss the results and the difference between the methods.

Hint: Make sure to set different seed number for each iteration while using the same parameters given in the previous question.

```
## your code here
```

Answer 3

Comment The double lasso (`glmnet`) approach has highest variation in spread of empirical α but is not biased (centered around the true $\alpha = 0.5$ value), followed by the double lasso (`hdm`), with the single lasso performing the worst in bias.

Again, note that this result may depend on the specific DGP you are using for simulation study.

```
S = 1000
```

```
alpha.double.lasso.hdm <- alpha.post.lasso.glmnet <- alpha.double.lasso.glmnet <- rep(NA, S)
```

```

for (i in 1:S) {
  dat <- generate_synth_data(alpha = alpha,
                             beta = beta,
                             gamma = gamma,
                             sigma = sigma,
                             delta = delta,
                             n = n,
                             pc = pc,
                             py = py,
                             pe = pe,
                             pu = pu,
                             seed.number = i)

  # Fit post lasso
  # 1. Pick lambda using cv
  lasso <- cv.glmnet(x = cbind(dat$X, dat$D), y = dat$Y)
  # 2. Fit lasso and check non-zero coefficients
  coef.lasso <- coef(lasso, s = "lambda.1se")
  coef.label <- rownames(coef.lasso)[as.vector(!(coef.lasso == 0))]
  coef.label <- coef.label[grepl("X.+?", coef.label)]
  # 3. Estimate alpha
  dat.lasso = data.frame(Y = dat$Y, D = dat$D)
  dat.lasso = cbind(dat.lasso, dat$X[,coef.label])
  post.lasso <- lm("Y ~ .", data = dat.lasso)
  alpha.post.lasso.glmnet[i] <- post.lasso$coefficients["D"]

  # Fit post double lasso
  # 1. Selection of predictors for Y
  y.lasso <- cv.glmnet(x = dat$X, y = dat$Y)
  coef.y.lasso <- coef(y.lasso, s = "lambda.1se")
  coef.y.label <- rownames(coef.y.lasso)[as.vector(!(coef.y.lasso == 0))]
  coef.y.label <- coef.y.label[grepl("X.+?", coef.y.label)]
  # 2. Selection of predictors for D
  d.lasso <- cv.glmnet(x = dat$X, y = dat$D)
  coef.d.lasso <- coef(d.lasso, s = "lambda.1se")
  coef.d.label <- rownames(coef.d.lasso)[as.vector(!(coef.d.lasso == 0))]
  coef.d.label <- coef.d.label[grepl("X.+?", coef.d.label)]
  # 3. Refit the model
  coef.double.label <- union(coef.y.label, coef.d.label)
  coef.double <- dat$X[,coef.double.label]
  dat.double <- data.frame(Y = dat$Y, D = dat$D)
  dat.double <- cbind(dat.double, coef.double)
  post.double.lasso <- lm("Y ~ .", data = dat.double)
  alpha.double.lasso.glmnet[i] <- post.double.lasso$coefficients["D"]

  hdm.lasso <- rlassoEffect(x=dat$X,d=dat$D,y=dat$Y,method="double selection")
  alpha.double.lasso.hdm[i] <- hdm.lasso$coefficient

  if((i%100)==0) cat(i,"th iteration done.\n")
}

```

Solution

```
## 100 th iteration done.
```

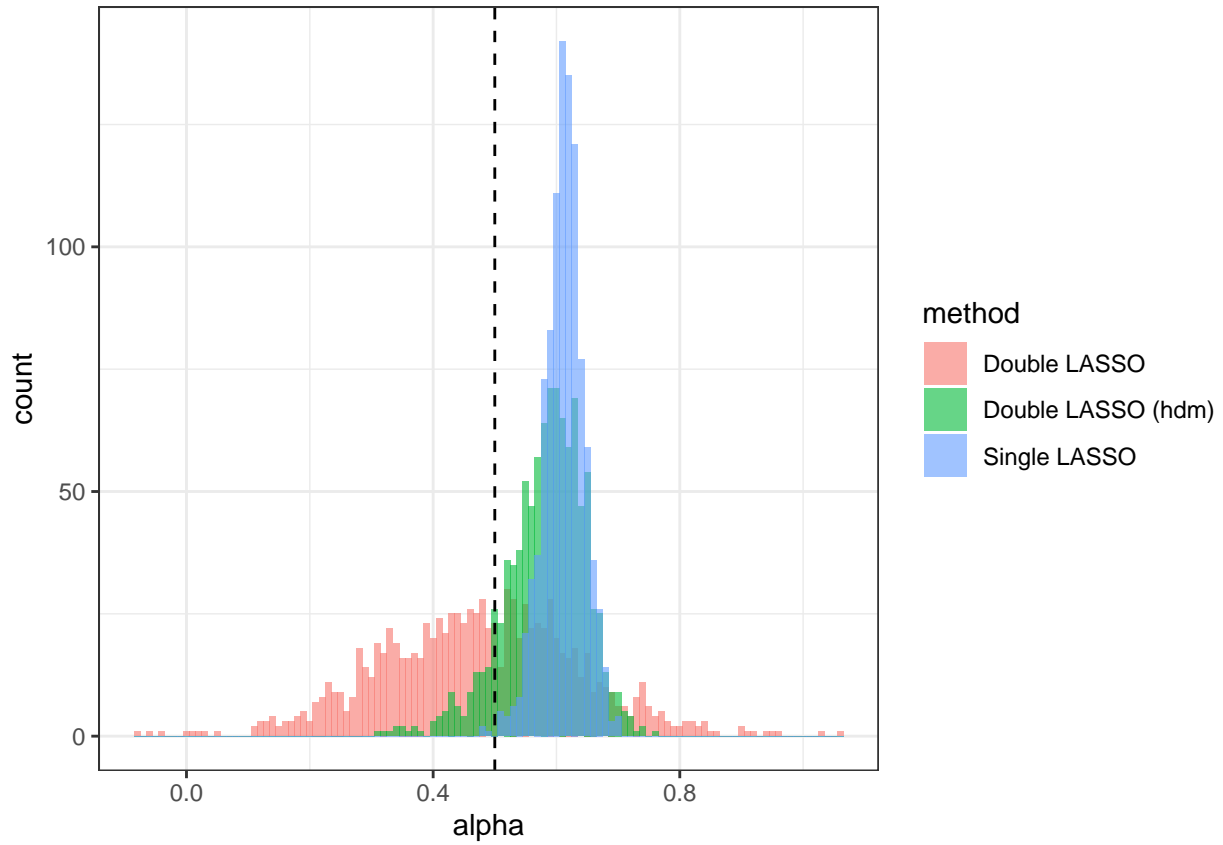
```
## 200 th iteration done.
## 300 th iteration done.
## 400 th iteration done.
## 500 th iteration done.
## 600 th iteration done.
## 700 th iteration done.
## 800 th iteration done.
## 900 th iteration done.
## 1000 th iteration done.
```

```
res = data.frame(est = c(alpha.post.lasso.glmnet,
                        alpha.double.lasso.glmnet,
                        alpha.double.lasso.hdm),
                method = c(rep("Single LASSO", S),
                           rep("Double LASSO", S),
                           rep("Double LASSO (hdm)", S)))

library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats   1.0.0      v stringr    1.5.1
## v ggplot2   3.4.4      v tibble     3.2.1
## v lubridate 1.9.3      v tidyr      1.3.1
## v purrr     1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x tidyr::pack()   masks Matrix::pack()
## x tidyr::unpack() masks Matrix::unpack()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
ggplot(res, aes(x=est, fill=method)) +
  geom_histogram(alpha=0.6, position = 'identity', binwidth = 0.01) +
  labs(x = "alpha") + theme_bw() +
  geom_vline(xintercept = alpha, linetype = "dashed")
```



```
mean(alpha.post.lasso.glmnet)
```

```
## [1] 0.6117834
```

```
mean(alpha.double.lasso.glmnet)
```

```
## [1] 0.4753872
```

```
mean(alpha.double.lasso.hdm)
```

```
## [1] 0.5813753
```