1.) I have a dataset containing family information of married couples, which have around 10 variables & 600+ observations. Independent variables are ~ gender, age, years married, children, religion etc. I have one response variable which is number of extra marital affairs. Now, I want to know what all factor influence the chances of extra marital affair. Since extra marital affair is a binary variable (either a person will have or not), so we can fit logistic regression model here to predict the probability of extra marital affair. install.packages('AER')
data(Affairs,package="AER")

R Code –

```
library('AER')
library(plyr)

# Read the data
Affairs <- read.csv(file.choose())
View(Affairs)
class(Affairs)

affairs1 <- Affairs
summary(affairs1)

table(affairs1$affairs)

affairs1$ynaffairs[affairs1$affairs > 0] <- 1
affairs1$ynaffairs[affairs1$affairs == 0] <- 0
affairs1$gender <- as.factor(revalue(Affairs$gender,c("male"=1, "female"=0)))
affairs1$children <- as.factor(revalue(Affairs$children,c("yes"=1, "no"=0)))
# sum(is.na(claimants))
# claimants <- na.omit(claimants) # Omitting NA values from the Data
# na.omit => will omit the rows which has atleast 1 NA value
View(affairs1)
```

```
colnames(affairs1)

class(affairs1)

attach(affairs1)

# Preparing a linear regression
mod_lm <- lm(naffairs ~ factor(unhap) + unhap+ yrsmarr1+ factor(kids) + vryhap+
        vryrel+vryunhap+avgmarr, data = affairs1)
summary(mod_lm)

pred1 <- predict(mod_lm,affairs1)
pred1

# plot(affairs,pred1)
# We can no way use the linear regression technique to classify the data
plot(pred1)

# GLM function use sigmoid curve to produce desirable results
# The output of sigmoid function lies in between 0-1
model <- glm(naffairs ~ factor(unhap) + unhap+ yrsmarr2+ factor(kids) + vryhap+
        vryrel+vryunhap+avgmarr, data = affairs1)

# To calculate the odds ratio manually we going r going to take exp of coef(model)
exp(coef(model))

# Confusion matrix table
```

prob <- predict(model,affairs1,type="response")

summary(model)

Creating empty vectors to store predicted classes based on threshold value

pred_values <- NULL

yes_no <- NULL

pred_values <- ifelse(prob>=0.5,1,0)

yes_no <- ifelse(prob>=0.5,"yes","no")

Creating new column to store the above values

affairs1[,"prob"] <- prob

affairs1[,"pred_values"] <- pred_values

affairs1[,"yes_no"] <- yes_no

View(affairs1[,c(1,9:11)])

table(affairs1$ynaffairs,affairs1$pred_values)

Output –

```
> library('AER')
Error in library("AER") : there is no package called 'AER'
> library(plyr)
Warning message:
package 'plyr' was built under R version 3.4.4
>
> # Read the data
> Affairs <- read.csv(file.choose())
> View(Affairs)
> class(Affairs)
[1] "data.frame"
>
> affairs1 <- Affairs
> summary(affairs1)
       X           naffairs            kids            vryunhap             unhap
```

```
 Min.   :  1   Min.   : 0.000   Min.   :0.0000   Min.   :0.00000   Min.   :0.
0000
 1st Qu.:151   1st Qu.: 0.000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.
0000
 Median :301   Median : 0.000   Median :1.0000   Median :0.00000   Median :0.
0000
 Mean   :301   Mean   : 1.456   Mean   :0.7155   Mean   :0.02662   Mean   :0.
1098
 3rd Qu.:451   3rd Qu.: 0.000   3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:0.
0000
 Max.   :601   Max.   :12.000   Max.   :1.0000   Max.   :1.00000   Max.   :1.
0000
    avgmarr          hapavg           vryhap         antirel            notr
el
 Min.   :0.0000   Min.   :0.0000   Min.   :0.000   Min.   :0.00000   Min.   :
0.0000
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.00000   1st Qu.:
0.0000
 Median :0.0000   Median :0.0000   Median :0.000   Median :0.00000   Median :
0.0000
 Mean   :0.1547   Mean   :0.3228   Mean   :0.386   Mean   :0.07987   Mean   :
0.2729
 3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:0.00000   3rd Qu.:
1.0000
 Max.   :1.0000   Max.   :1.0000   Max.   :1.000   Max.   :1.00000   Max.   :
1.0000
    slghtrel          smerel           vryrel          yrsmarr1           yrsm
arr2
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   
:0.0000
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.
:0.0000
 Median :0.0000   Median :0.0000   Median :0.0000   Median :0.00000   Median 
:0.0000
 Mean   :0.2146   Mean   :0.3161   Mean   :0.1165   Mean   :0.08652   Mean   
:0.1464
 3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.
:0.0000
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   
:1.0000
    yrsmarr3         yrsmarr4         yrsmarr5         yrsmarr6     
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
 Median :0.0000   Median :0.0000   Median :0.0000   Median :0.0000  
 Mean   :0.1747   Mean   :0.1364   Mean   :0.1165   Mean   :0.3394  
 3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:1.0000  
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
> 
> table(affairs1$affairs)
< table of extent 0 >
> 
> affairs1$ynaffairs[affairs1$affairs > 0] <- 1
Error in `$<-.data.frame`(`*tmp*`, ynaffairs, value = numeric(0)) : 
  replacement has 0 rows, data has 601
> affairs1$ynaffairs[affairs1$affairs == 0] <- 0
Error in `$<-.data.frame`(`*tmp*`, ynaffairs, value = numeric(0)) : 
  replacement has 0 rows, data has 601
```

```
> affairs1$gender <- as.factor(revalue(Affairs$gender,c("male"=1, "female"=0)
))
The following `from` values were not present in `x`: male, female
Error in `$<-.data.frame`(`*tmp*`, gender, value = integer(0)) : 
  replacement has 0 rows, data has 601
> affairs1$children <- as.factor(revalue(Affairs$children,c("yes"=1, "no"=0))
)
The following `from` values were not present in `x`: yes, no
Error in `$<-.data.frame`(`*tmp*`, children, value = integer(0)) : 
  replacement has 0 rows, data has 601
> # sum(is.na(claimants))
> # claimants <- na.omit(claimants) # Omitting NA values from the Data
> # na.omit => will omit the rows which has atleast 1 NA value
> View(affairs1)
> 
> 
> colnames(affairs1)
 [1] "X"        "naffairs" "kids"     "vryunhap" "unhap"    "avgmarr"  "hapav
g"   "vryhap"  
 [9] "antirel"  "notrel"   "slghtrel" "smerel"   "vryrel"   "yrsmarr1" "yrsma
rr2" "yrsmarr3"
[17] "yrsmarr4" "yrsmarr5" "yrsmarr6"
> 
> class(affairs1)
[1] "data.frame"
> 
> attach(affairs1)
> 
> # Preparing a linear regression 
> mod_lm <- lm(naffairs ~ factor(unhap) + unhap+ yrsmarr1+ factor(kids) + vry
hap+
+                vryrel+vryunhap+avgmarr, data = affairs1)
> summary(mod_lm)

Call:
lm(formula = naffairs ~ factor(unhap) + unhap + yrsmarr1 + factor(kids) + 
    vryhap + vryrel + vryunhap + avgmarr, data = affairs1)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.0207 -1.4688 -0.9721 -0.2304 11.4640 

Coefficients: (1 not defined because of singularities)
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)      1.1633     0.3485   3.338 0.000896 ***
factor(unhap)1   2.5518     0.4491   5.682 2.09e-08 ***
unhap                NA         NA      NA       NA    
yrsmarr1        -0.4361     0.4925  -0.885 0.376249    
factor(kids)1    0.3056     0.3147   0.971 0.332032    
vryhap          -0.4968     0.3121  -1.592 0.111964    
vryrel          -0.5804     0.4016  -1.445 0.148954    
vryunhap         2.5360     0.8184   3.099 0.002034 ** 
avgmarr          0.1510     0.3967   0.381 0.703666    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.143 on 593 degrees of freedom
```

```
Multiple R-squared:  0.1028,	Adjusted R-squared:  0.0922 
F-statistic: 9.706 on 7 and 593 DF,  p-value: 1.817e-11

> 
> pred1 <- predict(mod_lm,affairs1)
Warning message:
In predict.lm(mod_lm, affairs1) :
  prediction from a rank-deficient fit may be misleading
> pred1
          1           2           3           4           5           6           
7 
 1.16326796  1.16326796  1.16326796  1.46882344  0.97206593  0.39165804  0.87
819103 
          8           9          10          11          12          13          
14 
 0.66651045  3.44024295  0.87819103  1.46882344  4.02065084  0.66651045  4.02
065084 
         15          16          17          18          19          20          
21 
 4.02065084  1.46882344  0.88841555  0.23044626  0.97206593  1.16326796  1.16
326796 
         22          23          24          25          26          27          
28 
 0.97206593  1.61981071  0.72720377  0.66651045  0.97206593  4.02065084  0.66
651045 
         29          30          31          32          33          34          
35 
 1.16326796  0.66651045  1.46882344  1.46882344  0.97206593  1.46882344  4.02
065084 
         36          37          38          39          40          41          
42 
 0.66651045  0.66651045  0.88841555  0.97206593  4.02065084  1.46882344  1.46
882344 
         43          44          45          46          47          48          
49 
 1.61981071  0.66651045  0.72720377  0.88841555  1.46882344  1.31425522  3.42
443475 
         50          51          52          53          54          55          
56 
 4.02065084  4.02065084  0.66651045  1.46882344  1.61981071  0.66651045  0.66
651045 
         57          58          59          60          61          62          
63 
 0.97206593  0.23044626  0.97206593  0.39165804  0.39165804  1.03940282  0.14
679587 
         64          65          66          67          68          69          
70 
 4.00484264  1.03940282  0.66651045  1.46882344  0.88841555  0.97206593  1.46
882344 
         71          72          73          74          75          76          
77 
 0.97206593  1.46882344  1.46882344  4.02065084 -0.34996163  1.46882344  0.39
165804 
         78          79          80          81          82          83          
84 
 1.46882344  0.66651045  1.46882344  1.61981071  4.00484264  3.71509535  1.16
326796 
```

```
         85          86          87          88          89          90          
91 
 0.88841555  0.97206593  0.72720377  1.46882344  3.44024295  1.46882344  0.23
044626 
         92          93          94          95          96          97          
98 
 0.66651045  1.61981071  1.46882344  1.16326796  4.02065084  0.97206593 -0.34
996163 
         99         100         101         102         103         104         
105 
 0.97206593  1.46882344  0.97206593  0.08610256  1.16326796  1.46882344  1.46
882344 
        106         107         108         109         110         111         
112 
 0.66651045  0.39165804  0.23044626  1.61981071  0.97206593  0.97206593  0.97
206593 
        113         114         115         116         117         118         
119 
 1.31425522  0.97206593  0.97206593  0.66651045  4.02065084  0.97206593  0.97
206593 
        120         121         122         123         124         125         
126 
 0.72720377  1.31425522  0.97206593  1.46882344  1.61981071  0.72720377  1.61
981071 
        127         128         129         130         131         132         
133 
 0.97206593  3.71509535  0.97206593  1.46882344  1.46882344  1.61981071  1.46
882344 
        134         135         136         137         138         139         
140 
 0.97206593  1.03940282  0.72720377  0.39165804  4.00484264  1.46882344  0.97
206593 
        141         142         143         144         145         146         
147 
 0.97206593  0.39165804  1.46882344  1.16326796  0.23044626  4.02065084  0.39
165804 
        148         149         150         151         152         153         
154 
 0.97206593  0.97206593  4.00484264  1.16326796  1.46882344  1.46882344  1.46
882344 
        155         156         157         158         159         160         
161 
 1.46882344  1.61981071  1.16326796  0.97206593  0.97206593  0.97206593  0.08
610256 
        162         163         164         165         166         167         
168 
 0.88841555  1.03275925  1.16326796  0.66651045  4.02065084  0.58286006  0.23
044626 
        169         170         171         172         173         174         
175 
 1.46882344  0.97206593  0.88841555  0.88841555  1.61981071  4.02065084  0.66
651045 
        176         177         178         179         180         181         
182 
 1.46882344  0.39165804  0.23044626  0.66651045  0.66651045  1.46882344  0.66
651045 
```

```
       183        184        185        186        187        188 
189 
 1.61981071  0.66651045  0.97206593  0.66651045  1.31425522  1.61981071  1.03
940282 
       190        191        192        193        194        195 
196 
 1.31425522  1.46882344  1.46882344  0.23044626  1.46882344  0.97206593  1.46
882344 
       197        198        199        200        201        202 
203 
 1.16326796  1.46882344  0.88841555  1.03940282  1.46882344  1.31425522  1.46
882344 
       204        205        206        207        208        209 
210 
 1.46882344  1.03940282  4.00484264  1.46882344  1.46882344  1.61981071  0.87
819103 
       211        212        213        214        215        216 
217 
 1.61981071  1.61981071  1.18374652  1.61981071  3.42443475  4.02065084  1.61
981071 
       218        219        220        221        222        223 
224 
 3.71509535  1.46882344  0.97206593  0.97206593  1.46882344  0.97206593  0.39
165804 
       225        226        227        228        229        230 
231 
 0.88841555  1.46882344  1.46882344  4.02065084  4.02065084  1.16326796  4.02
065084 
       232        233        234        235        236        237 
238 
 1.46882344  4.02065084  0.97206593  0.23044626  1.46882344  0.97206593  1.46
882344 
       239        240        241        242        243        244 
245 
 0.66651045  3.58458665  1.46882344  1.16326796  1.46882344  1.61981071  1.46
882344 
       246        247        248        249        250        251 
252 
 3.44024295  0.97206593  1.46882344  1.61981071  0.66651045  0.53600174  0.97
206593 
       253        254        255        256        257        258 
259 
 0.23044626  0.97206593  0.97206593  0.97206593  0.23044626  1.46882344  4.02
065084 
       260        261        262        263        264        265 
266 
 1.61981071  1.03275925  1.46882344  1.61981071  1.46882344  1.61981071  0.66
651045 
       267        268        269        270        271        272 
273 
 4.02065084  0.66651045  0.39165804  0.97206593  0.66651045  0.87819103  4.02
065084 
       274        275        276        277        278        279 
280 
 0.87819103  4.00484264  1.46882344  4.02065084  0.66651045  1.61981071  1.16
326796 
```

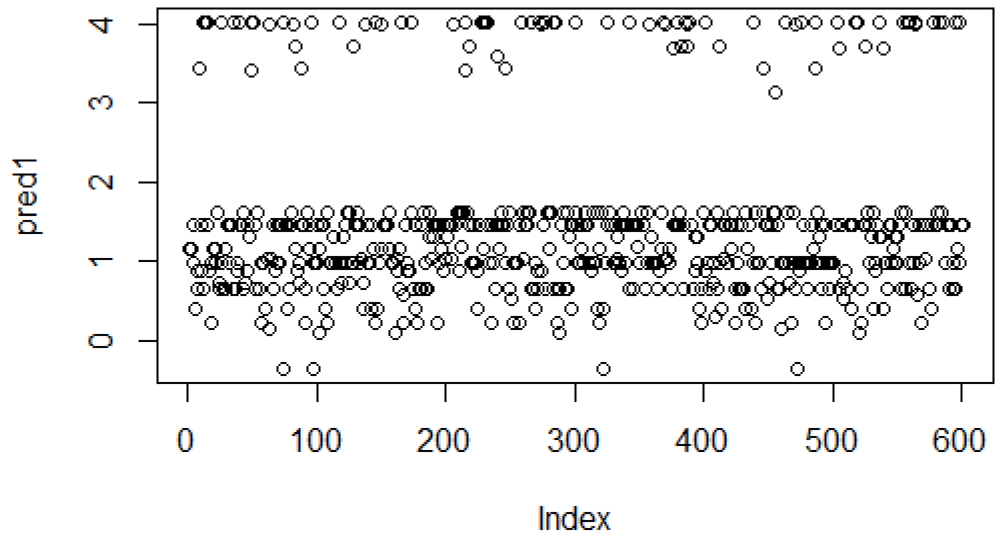
```
        281         282         283         284         285         286 
287 
 1.61981071  1.61981071  0.66651045  4.02065084  4.02065084  0.23044626  0.66
651045 
        288         289         290         291         292         293 
294 
 0.08610256  0.66651045  1.46882344  0.97206593  1.61981071  0.66651045  1.61
981071 
        295         296         297         298         299         300 
301 
 0.66651045  1.31425522  1.46882344  0.39165804  1.46882344  4.02065084  1.61
981071 
        302         303         304         305         306         307 
308 
 1.46882344  0.97206593  0.97206593  1.16326796  1.46882344  0.97206593  1.46
882344 
        309         310         311         312         313         314 
315 
 1.61981071  1.46882344  0.97206593  0.97206593  0.88841555  0.97206593  1.61
981071 
        316         317         318         319         320         321 
322 
 0.97206593  1.61981071  0.39165804  1.16326796  0.23044626  1.61981071 -0.34
996163 
        323         324         325         326         327         328 
329 
 0.39165804  0.97206593  4.02065084  1.46882344  1.61981071  1.46882344  1.46
882344 
        330         331         332         333         334         335 
336 
 0.97206593  0.97206593  1.46882344  1.03275925  1.31425522  0.97206593  1.46
882344 
        337         338         339         340         341         342 
343 
 0.88841555  1.61981071  1.46882344  0.97206593  1.46882344  4.02065084  0.66
651045 
        344         345         346         347         348         349 
350 
 1.46882344  0.66651045  1.46882344  1.46882344  1.46882344  1.18374652  0.97
206593 
        351         352         353         354         355         356 
357 
 0.97206593  1.61981071  1.46882344  1.46882344  0.66651045  1.61981071  0.97
206593 
        358         359         360         361         362         363 
364 
 4.00484264  0.97206593  1.31425522  0.97206593  0.97206593  0.97206593  0.97
206593 
        365         366         367         368         369         370 
371 
 0.66651045  0.87819103  1.61981071  1.16326796  4.02065084  1.03940282  4.00
484264 
        372         373         374         375         376         377 
378 
 1.03940282  0.97206593  1.16326796  0.66651045  3.69928715  1.46882344  1.46
882344 
```

```
       379        380        381        382        383        384 
385 
 1.46882344  4.02065084  1.61981071  1.46882344  3.71509535  1.46882344  0.66
651045 
       386        387        388        389        390        391 
392 
 4.02065084  3.71509535  4.00484264  4.02065084  1.46882344  1.46882344  0.66
651045 
       393        394        395        396        397        398 
399 
 0.88841555  1.31425522  0.97206593  1.31425522  0.39165804  0.23044626  0.66
651045 
       400        401        402        403        404        405 
406 
 1.61981071  0.88841555  4.02065084  0.97206593  1.46882344  0.66651045  1.46
882344 
       407        408        409        410        411        412 
413 
 0.72720377  0.97206593  0.29778314  1.61981071  0.66651045  3.71509535  0.97
206593 
       414        415        416        417        418        419 
420 
 0.39165804  1.46882344  1.03940282  1.46882344  0.97206593  0.97206593  1.46
882344 
       421        422        423        424        425        426 
427 
 1.16326796  1.46882344  1.61981071  0.23044626  0.66651045  1.46882344  0.97
206593 
       428        429        430        431        432        433 
434 
 0.66651045  0.66651045  1.46882344  0.66651045  0.39165804  1.61981071  1.16
326796 
       435        436        437        438        439        440 
441 
 1.46882344  1.46882344  0.39165804  0.97206593  4.02065084  1.61981071  0.23
044626 
       442        443        444        445        446        447 
448 
 0.97206593  0.97206593  0.97206593  0.66651045  3.44024295  1.61981071  1.46
882344 
       449        450        451        452        453        454 
455 
 0.53600174  1.46882344  0.72720377  0.97206593  1.46882344  1.61981071  1.61
981071 
       456        457        458        459        460        461 
462 
 3.13468746  1.46882344  0.66651045  0.66651045  0.14679587  0.97206593  0.97
206593 
       463        464        465        466        467        468 
469 
 4.02065084  0.97206593  1.61981071  0.97206593  0.72720377  0.23044626  0.97
206593 
       470        471        472        473        474        475 
476 
 1.61981071  4.00484264  0.66651045 -0.34996163  0.88841555  0.97206593  4.02
065084 
```

```
       477        478        479        480        481        482 
483 
 1.46882344  0.97206593  0.97206593  0.66651045  0.97206593  0.97206593  0.97
206593 
       484        485        486        487        488        489 
490 
 1.46882344  1.16326796  4.02065084  3.44024295  1.61981071  0.97206593  1.31
425522 
       491        492        493        494        495        496 
497 
 0.66651045  1.46882344  0.97206593  1.46882344  0.23044626  0.97206593  0.66
651045 
       498        499        500        501        502        503 
504 
 0.97206593  0.97206593  0.97206593  0.97206593  1.46882344  1.46882344  4.02
065084 
       505        506        507        508        509        510 
511 
 3.69928715  0.66651045  1.46882344  0.53600174  0.72720377  0.88841555  0.66
651045 
       512        513        514        515        516        517 
518 
 0.66651045  1.46882344  0.39165804  1.46882344  1.46882344  4.02065084  4.02
065084 
       519        520        521        522        523        524 
525 
 4.02065084  4.02065084  0.08610256  0.66651045  0.23044626  1.46882344  1.46
882344 
       526        527        528        529        530        531 
532 
 3.71509535  1.61981071  0.97206593  1.46882344  1.31425522  0.39165804  1.61
981071 
       533        534        535        536        537        538 
539 
 0.97206593  0.88841555  1.46882344  1.31425522  4.02065084  1.31425522  3.69
928715 
       540        541        542        543        544        545 
546 
 0.39165804  0.97206593  0.66651045  0.66651045  1.46882344  1.46882344  0.97
206593 
       547        548        549        550        551        552 
553 
 1.16326796  0.97206593  1.31425522  0.97206593  1.31425522  1.46882344  0.66
651045 
       554        555        556        557        558        559 
560 
 1.61981071  4.02065084  0.66651045  1.61981071  4.02065084  0.97206593  4.02
065084 
       561        562        563        564        565        566 
567 
 0.66651045  0.97206593  1.46882344  4.00484264  4.02065084  0.58286006  1.46
882344 
       568        569        570        571        572        573 
574 
 0.97206593  0.23044626  1.46882344  1.46882344  1.03940282  1.46882344  1.46
882344 
```

```
        575          576          577          578          579          580          
581 
 0.66651045  1.46882344  0.39165804  4.02065084  1.46882344  1.46882344  1.46
882344 
        582          583          584          585          586          587          
588 
 1.61981071  4.02065084  1.61981071  1.61981071  4.02065084  0.97206593  0.97
206593 
        589          590          591          592          593          594          
595 
 1.46882344  1.46882344  0.66651045  1.46882344  0.66651045  0.97206593  4.02
065084 
        596          597          598          599          600          601 
 0.66651045  1.16326796  0.97206593  4.02065084  1.46882344  1.46882344 
> 
> # plot(affairs,pred1)
> # We can no way use the linear regression technique to classify the data
> plot(pred1)
> 
> # GLM function use sigmoid curve to produce desirable results 
> # The output of sigmoid function lies in between 0-1
> model <- glm(naffairs ~ factor(unhap) + unhap+ yrsmarr2+ factor(kids) + vry
hap+
+                 vryrel+vryunhap+avgmarr, data = affairs1)
> 
> # To calculate the odds ratio manually we going r going to take exp of coef
(model)
> exp(coef(model))
  (Intercept) factor(unhap)1          unhap       yrsmarr2  factor(kids)1       
vryhap 
    3.8235258     13.8210486             NA      0.4239067      1.1424363      
0.6571631 
       vryrel       vryunhap        avgmarr 
    0.5394117     12.2165496      1.1681618 
> 
> # Confusion matrix table
> prob <- predict(model,affairs1,type="response")
Warning message:
In predict.lm(object, newdata, se.fit, scale = 1, type = ifelse(type ==  :
  prediction from a rank-deficient fit may be misleading
> summary(model)

Call:
glm(formula = naffairs ~ factor(unhap) + unhap + yrsmarr2 + factor(kids) + 
    vryhap + vryrel + vryunhap + avgmarr, data = affairs1)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-4.1005  -1.4743  -1.0545  -0.0631  11.3839  

Coefficients: (1 not defined because of singularities)
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)      1.3412     0.3513   3.818 0.000149 ***
factor(unhap)1   2.6262     0.4480   5.862 7.61e-09 ***
unhap                NA         NA      NA       NA    
yrsmarr2        -0.8582     0.4035  -2.127 0.033852 *  
factor(kids)1    0.1332     0.3196   0.417 0.677050    
```

```
vryhap          -0.4198     0.3129  -1.342 0.180255
vryrel          -0.6173     0.4001  -1.543 0.123402
vryunhap         2.5028     0.8156   3.069 0.002248 **
avgmarr          0.1554     0.3955   0.393 0.694466
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 9.816643)

    Null deviance: 6529.1  on 600  degrees of freedom
Residual deviance: 5821.3  on 593  degrees of freedom
AIC: 3088.2

Number of Fisher Scoring iterations: 2

>
> # Creating empty vectors to store predicted classes based on threshold valu
e
> pred_values <- NULL
> yes_no <- NULL
>
> pred_values <- ifelse(prob>=0.5,1,0)
> yes_no <- ifelse(prob>=0.5,"yes","no")
>
> # Creating new column to store the above values
> affairs1[,"prob"] <- prob
> affairs1[,"pred_values"] <- pred_values
> affairs1[,"yes_no"] <- yes_no
>
> View(affairs1[,c(1,9:11)])
>
> table(affairs1$ynaffairs,affairs1$pred_values)
```



Python Code –

```
import numpy as np
import pandas as pd
import statsmodels.api as sm
import matplotlib.pyplot as plt
from patsy import dmatrices
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split as split
from sklearn import metrics
from sklearn.model_selection import cross_val_score

# load dataset
dta = sm.datasets.fair.load_pandas().data
```

```
# adding "affair" column: 1 represents having affairs, 0 represents not

dta['affair'] = (dta.affairs > 0).astype(int)

dta = dta.rename(columns={"rate_marriage": "rateMarriage", "yrs_married":
"yearsMarried","occupation_husb":"husbandOccupation"})


print(dta.sample(5))




print(dta.groupby('affair').mean())


print(dta.groupby('rateMarriage').mean())


# histogram of education

dta.educ.hist()

plt.title('Histogram of Education')

plt.xlabel('Education Level')

plt.ylabel('Frequency')




# histogram of marriage rating

dta.rateMarriage.hist()

plt.title('Histogram of Marriage Rating')

plt.xlabel('Marriage Rating')

plt.ylabel('Frequency')




pd.crosstab(dta.rateMarriage, dta.affair.astype(bool)).plot(kind='bar')

plt.title('Marriage Rating Distribution by Affair Status')
```

```
plt.xlabel('Marriage Rating')
plt.ylabel('Frequency')


affair_yrs_married = pd.crosstab(dta.yearsMarried, dta.affair.astype(bool))
affair_yrs_married.div(affair_yrs_married.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Affair Percentage by Years Married')
plt.xlabel('Years Married')
plt.ylabel('Percentage')


# create dataframes with an intercept column and dummy variables for
# occupation and occupation_husb
y, X = dmatrices('affair ~ rateMarriage + age + yearsMarried + children + \
            religious + educ + C(occupation) + C(husbandOccupation)',
            dta, return_type="dataframe")
X.columns
print(X.head(5))


# fix column names of X
X = X.rename(columns = {'C(occupation)[T.2.0]':'occ_2',
                'C(occupation)[T.3.0]':'occ_3',
                'C(occupation)[T.4.0]':'occ_4',
                'C(occupation)[T.5.0]':'occ_5',
                'C(occupation)[T.6.0]':'occ_6',
                'C(husbandOccupation)[T.2.0]':'occ_husb_2',
                'C(husbandOccupation)[T.3.0]':'occ_husb_3',
                'C(husbandOccupation)[T.4.0]':'occ_husb_4',
```

```
                    'C(husbandOccupation)[T.5.0]':'occ_husb_5',
                    'C(husbandOccupation)[T.6.0]':'occ_husb_6'})


print(X.head())


# flatten y into a 1-D array
y = np.ravel(y)


model = LogisticRegression()
model = model.fit(X, y)

#accuracy obtained from training dataset
model.score(X, y)


# what percentage had affairs?
print(y.mean())


# examine the coefficients
X.columns, np.transpose(model.coef_)


# evaluate the model by splitting the data-set into train and test sets
X_train, X_test, y_train, y_test = split(X, y, test_size=0.3)
```

```
model2 = LogisticRegression()
model2.fit(X_train, y_train)


predicted = model2.predict(X_test)
print(y_test)
predicted


# generate class probabilities
probs = model2.predict_proba(X_test)
probs


# generate evaluation metrics
print(metrics.accuracy_score(y_test, predicted))
print(metrics.roc_auc_score(y_test, probs[:, 1]))

import seaborn as sns
conf_matrix = metrics.confusion_matrix(y_test, predicted)
sns.heatmap(conf_matrix, annot=True,cmap='Blues')

print(metrics.classification_report(y_test, predicted))
```

Output –

```
   rateMarriage   age  yearsMarried  ...  husbandOccupation  affairs  affair
```

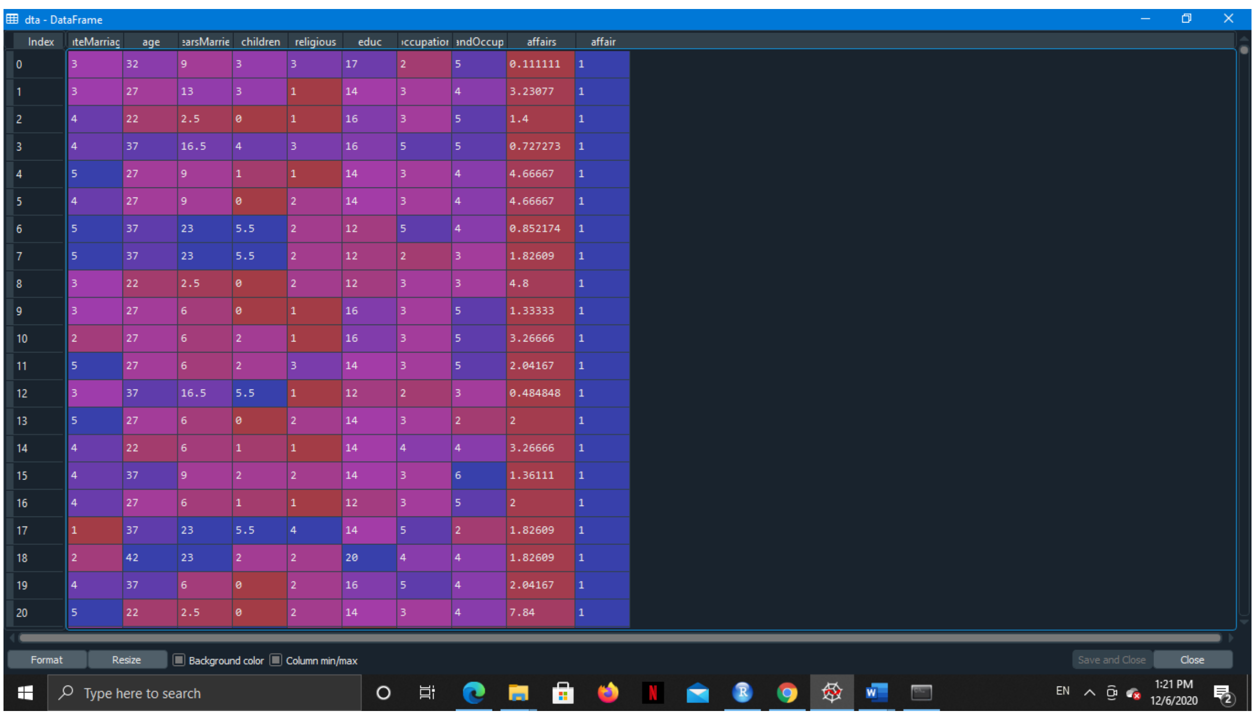
```
4198        3.0  32.0        13.0  ...          5.0    0.0     0
5478        3.0  37.0        16.5  ...          6.0    0.0     0
6163        4.0  27.0         9.0  ...          5.0    0.0     0
3505        5.0  27.0        13.0  ...          5.0    0.0     0
6175        4.0  27.0         2.5  ...          2.0    0.0     0

[5 rows x 10 columns]
       rateMarriage        age  ...  husbandOccupation   affairs
affair                          ...
0          4.329701  28.390679  ...           3.833758  0.000000
1          3.647345  30.537019  ...           3.884559  2.187243

[2 rows x 9 columns]
                   age  yearsMarried  ...   affairs    affair
rateMarriage                          ...
1.0          33.823232     13.914141  ...  1.201671  0.747475
2.0          30.471264     10.727011  ...  1.615745  0.635057
3.0          30.008056     10.239174  ...  1.371281  0.550856
4.0          28.856601      8.816905  ...  0.674837  0.322926
5.0          28.574702      8.311662  ...  0.348174  0.181446

[5 rows x 9 columns]
   Intercept  C(occupation)[T.2.0]  ...  religious  educ
0        1.0                   1.0  ...        3.0  17.0
1        1.0                   0.0  ...        1.0  14.0
2        1.0                   0.0  ...        1.0  16.0
3        1.0                   0.0  ...        3.0  16.0
4        1.0                   0.0  ...        1.0  14.0
```

```
[5 rows x 17 columns]
   Intercept  occ_2  occ_3  occ_4  ...  yearsMarried  children  religious  educ
0        1.0    1.0    0.0    0.0  ...           9.0       3.0        3.0  17.0
1        1.0    0.0    1.0    0.0  ...          13.0       3.0        1.0  14.0
2        1.0    0.0    1.0    0.0  ...           2.5       0.0        1.0  16.0
3        1.0    0.0    0.0    0.0  ...          16.5       4.0        3.0  16.0
4        1.0    0.0    1.0    0.0  ...           9.0       1.0        1.0  14.0

[5 rows x 17 columns]
0.3224945020420987
C:\anaconda\lib\site-packages\sklearn\linear_model\_logistic.py:764: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
C:\anaconda\lib\site-packages\sklearn\linear_model\_logistic.py:764: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
[0. 1. 1. ... 0. 1. 0.]
```

0.7303664921465969

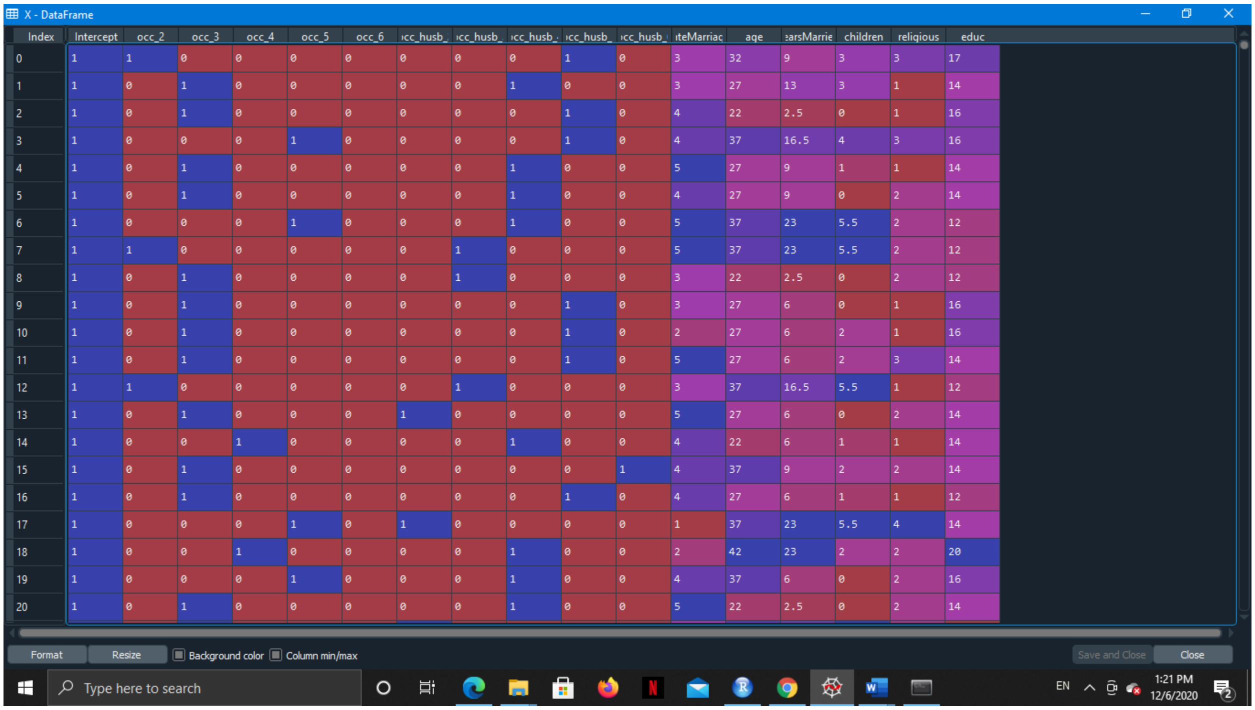
0.7413537624302692

	precision	recall	f1-score	support
0.0	0.75	0.90	0.82	1302
1.0	0.63	0.37	0.46	608
accuracy			0.73	1910
macro avg	0.69	0.63	0.64	1910
weighted avg	0.71	0.73	0.71	1910

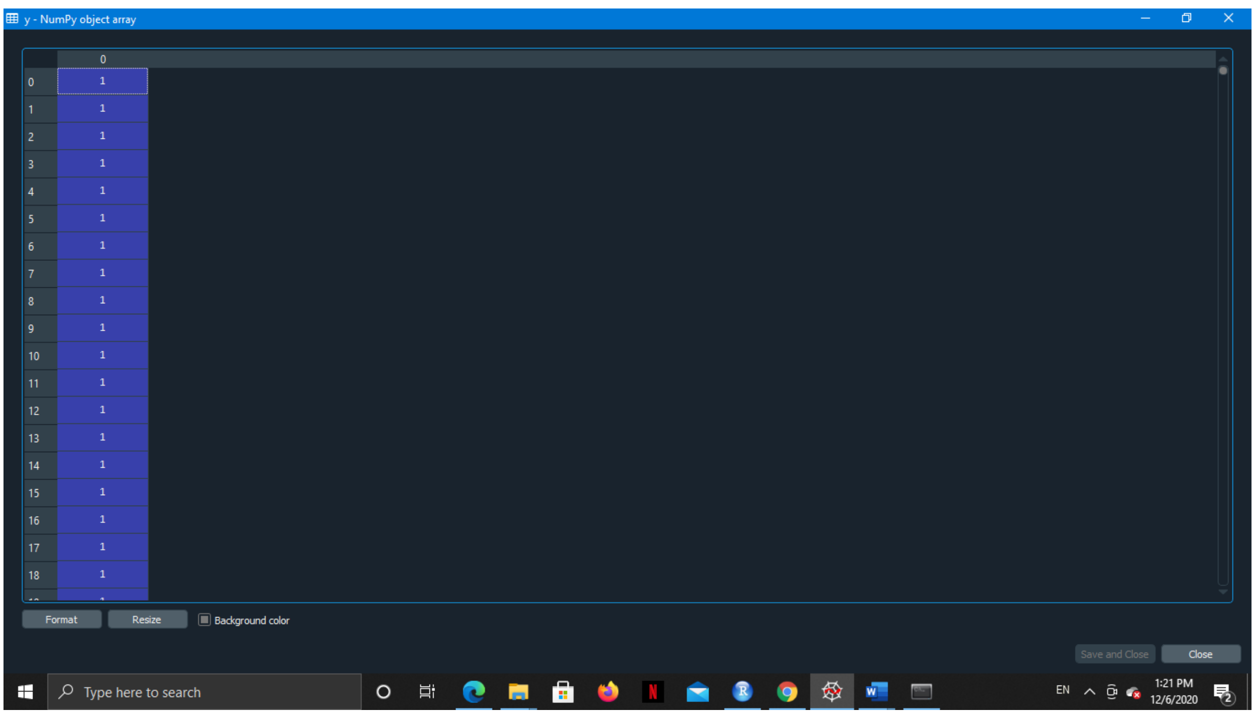

dta - DataFrame

Index	iteMarriag	age	earsMarrie	children	religious	educ	ccupatior	indOccup	affairs	affair
0	3	32	9	3	3	17	2	5	0.111111	1
1	3	27	13	3	1	14	3	4	3.23077	1
2	4	22	2.5	0	1	16	3	5	1.4	1
3	4	37	16.5	4	3	16	5	5	0.727273	1
4	5	27	9	1	1	14	3	4	4.66667	1
5	4	27	9	0	2	14	3	4	4.66667	1
6	5	37	23	5.5	2	12	5	4	0.852174	1
7	5	37	23	5.5	2	12	2	3	1.82609	1
8	3	22	2.5	0	2	12	3	3	4.8	1
9	3	27	6	0	1	16	3	5	1.33333	1
10	2	27	6	2	1	16	3	5	3.26666	1
11	5	27	6	2	3	14	3	5	2.04167	1
12	3	37	16.5	5.5	1	12	2	3	0.484848	1
13	5	27	6	0	2	14	3	2	2	1
14	4	22	6	1	1	14	4	4	3.26666	1
15	4	37	9	2	2	14	3	6	1.36111	1
16	4	27	6	1	1	12	3	5	2	1
17	1	37	23	5.5	4	14	5	2	1.82609	1
18	2	42	23	2	2	20	4	4	1.82609	1
19	4	37	6	0	2	16	5	4	2.04167	1
20	5	22	2.5	0	2	14	3	4	7.84	1

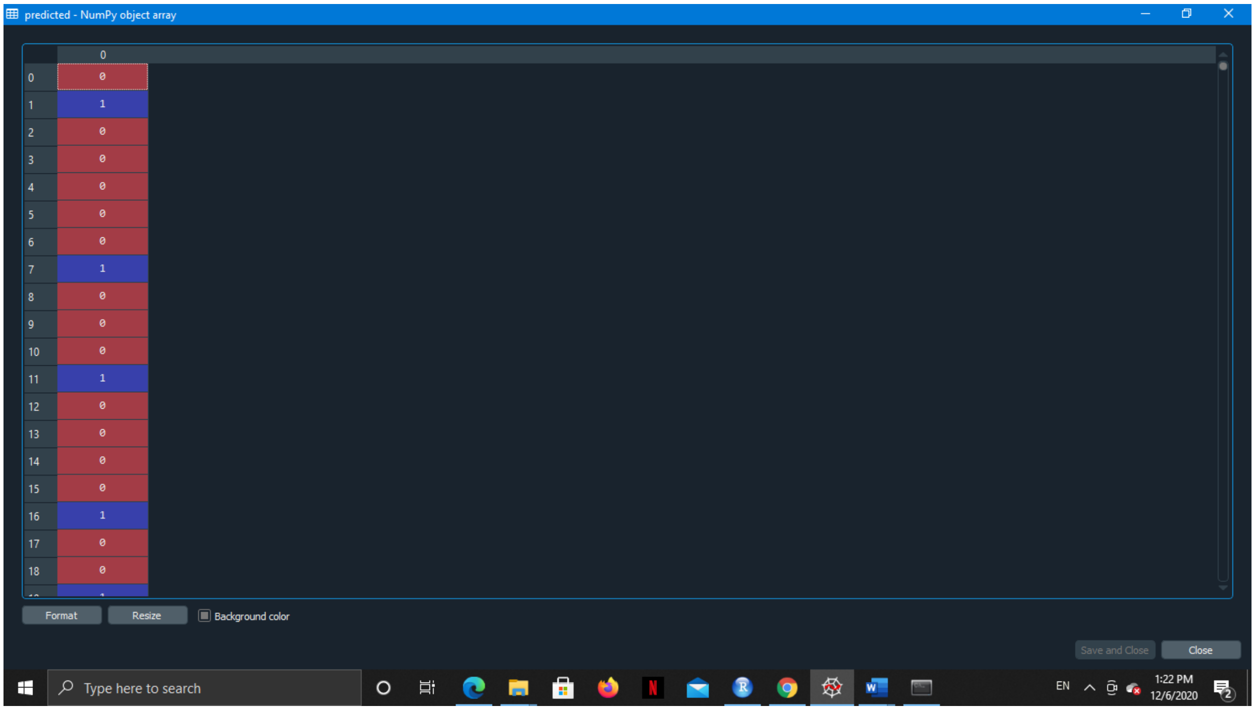
Format Resize Background color Column min/max Save and Close Close


X - DataFrame

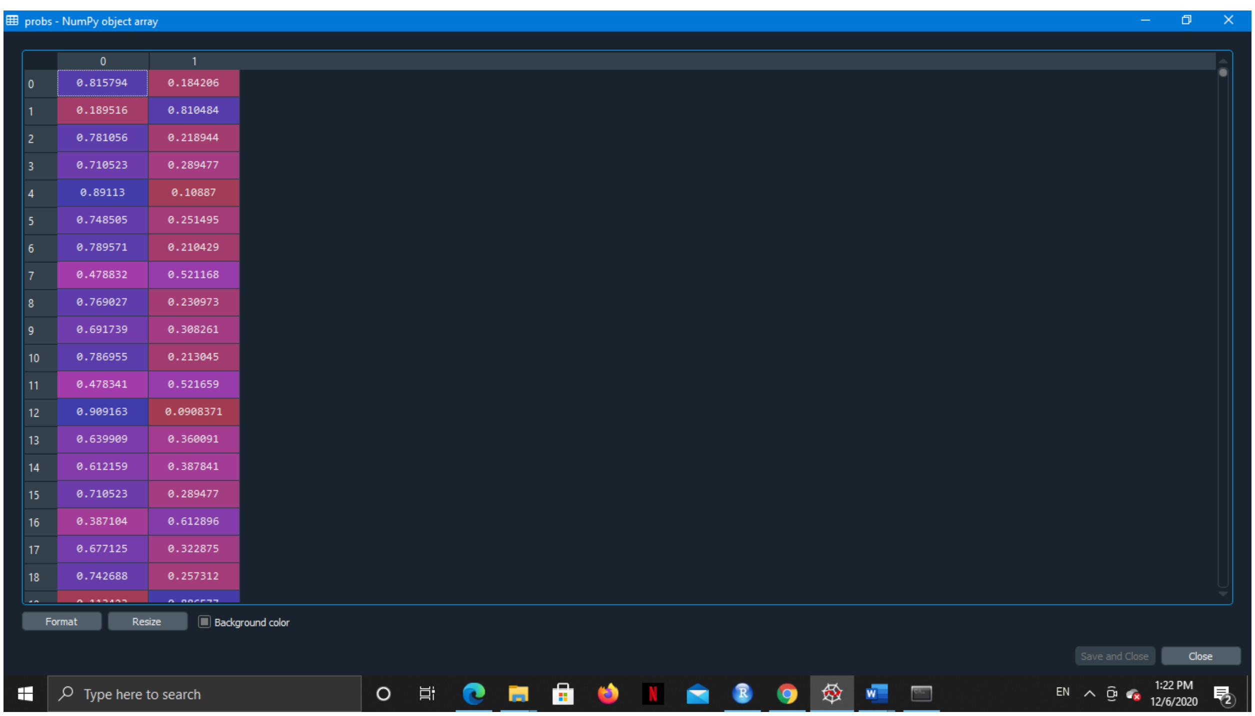
Index	Intercept	occ_2	occ_3	occ_4	occ_5	occ_6	cc_husb_	cc_husb_	cc_husb_	cc_husb_	cc_husb_	iteMarriac	age	earsMarrie	children	religious	educ
0	1	1	0	0	0	0	0	0	0	1	0	3	32	9	3	3	17
1	1	0	1	0	0	0	0	0	1	0	0	3	27	13	3	1	14
2	1	0	1	0	0	0	0	0	0	1	0	4	22	2.5	0	1	16
3	1	0	0	0	1	0	0	0	0	1	0	4	37	16.5	4	3	16
4	1	0	1	0	0	0	0	0	1	0	0	5	27	9	1	1	14
5	1	0	1	0	0	0	0	0	1	0	0	4	27	9	0	2	14
6	1	0	0	0	1	0	0	0	1	0	0	5	37	23	5.5	2	12
7	1	1	0	0	0	0	0	1	0	0	0	5	37	23	5.5	2	12
8	1	0	1	0	0	0	0	1	0	0	0	3	22	2.5	0	2	12
9	1	0	1	0	0	0	0	0	0	1	0	3	27	6	0	1	16
10	1	0	1	0	0	0	0	0	0	1	0	2	27	6	2	1	16
11	1	0	1	0	0	0	0	0	0	1	0	5	27	6	2	3	14
12	1	1	0	0	0	0	0	1	0	0	0	3	37	16.5	5.5	1	12
13	1	0	1	0	0	0	1	0	0	0	0	5	27	6	0	2	14
14	1	0	0	1	0	0	0	0	1	0	0	4	22	6	1	1	14
15	1	0	1	0	0	0	0	0	0	0	1	4	37	9	2	2	14
16	1	0	1	0	0	0	0	0	0	1	0	4	27	6	1	1	12
17	1	0	0	0	1	0	1	0	0	0	0	1	37	23	5.5	4	14
18	1	0	0	1	0	0	0	0	1	0	0	2	42	23	2	2	20
19	1	0	0	0	1	0	0	0	1	0	0	4	37	6	0	2	16
20	1	0	1	0	0	0	0	0	1	0	0	5	22	2.5	0	2	14


y - NumPy object array

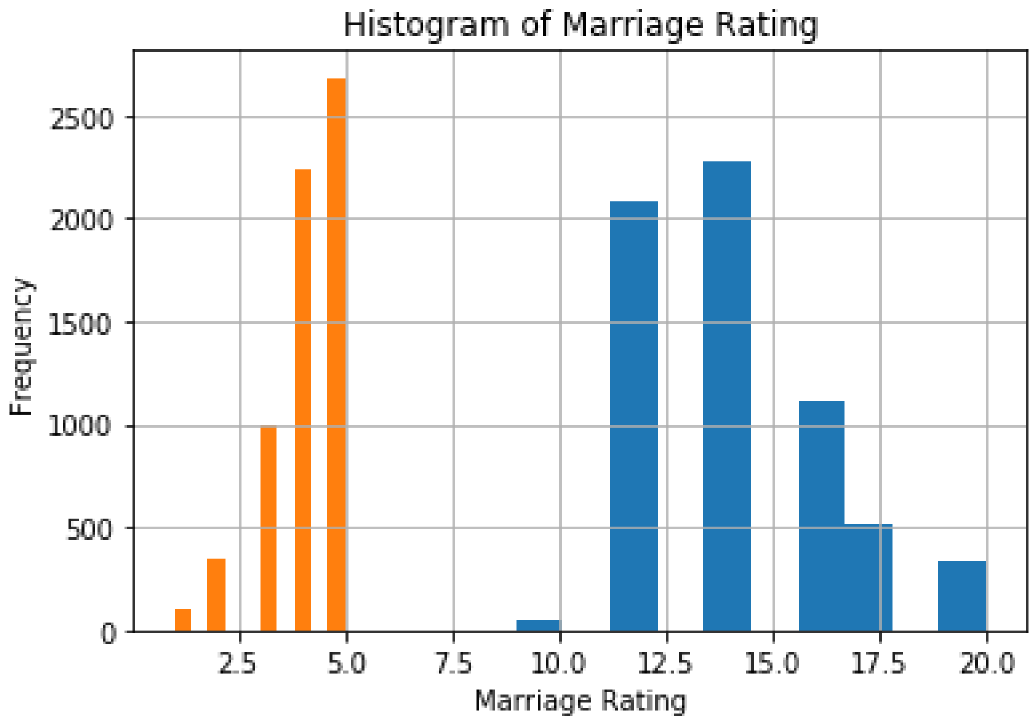
	0
0	1
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	1
17	1
18	1



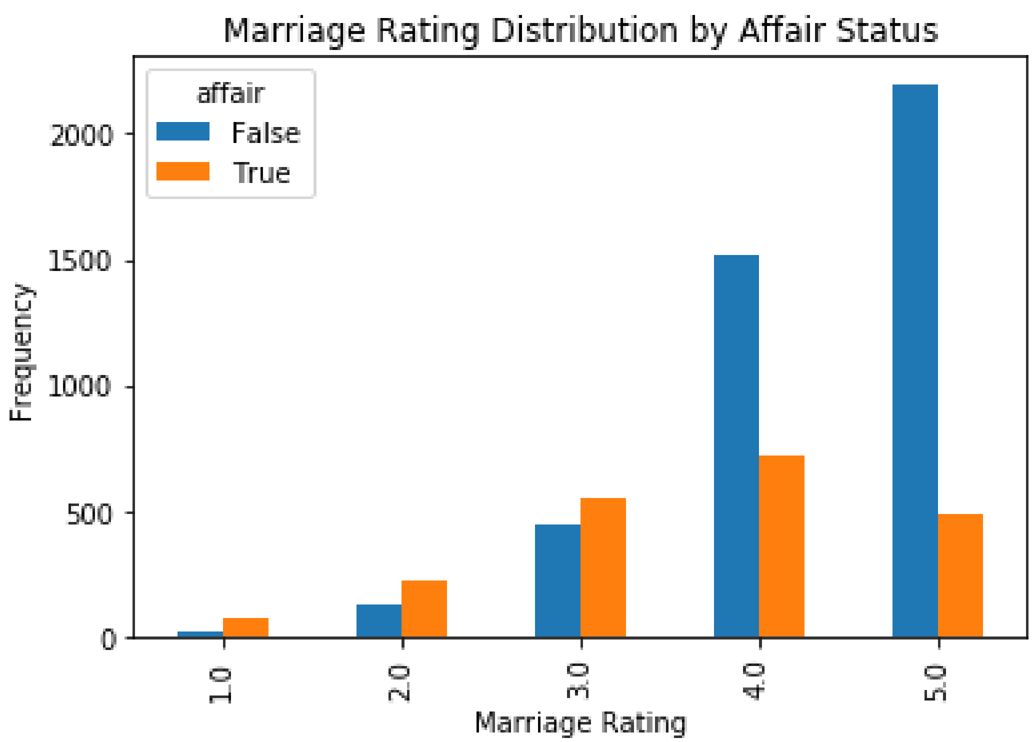
	0
0	0
1	1
2	0
3	0
4	0
5	0
6	0
7	1
8	0
9	0
10	0
11	1
12	0
13	0
14	0
15	0
16	1
17	0
18	0



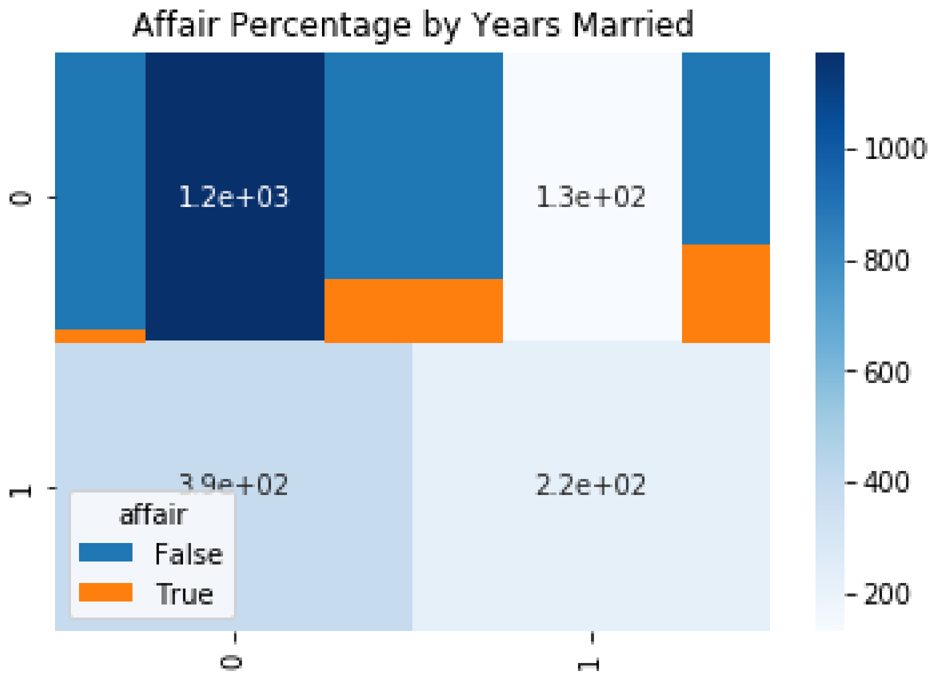
	0	1
0	0.815794	0.184206
1	0.189516	0.810484
2	0.781056	0.218944
3	0.710523	0.289477
4	0.89113	0.10887
5	0.748505	0.251495
6	0.789571	0.210429
7	0.478832	0.521168
8	0.769027	0.230973
9	0.691739	0.308261
10	0.786955	0.213045
11	0.478341	0.521659
12	0.909163	0.0908371
13	0.639909	0.360091
14	0.612159	0.387841
15	0.710523	0.289477
16	0.387104	0.612896
17	0.677125	0.322875
18	0.742688	0.257312

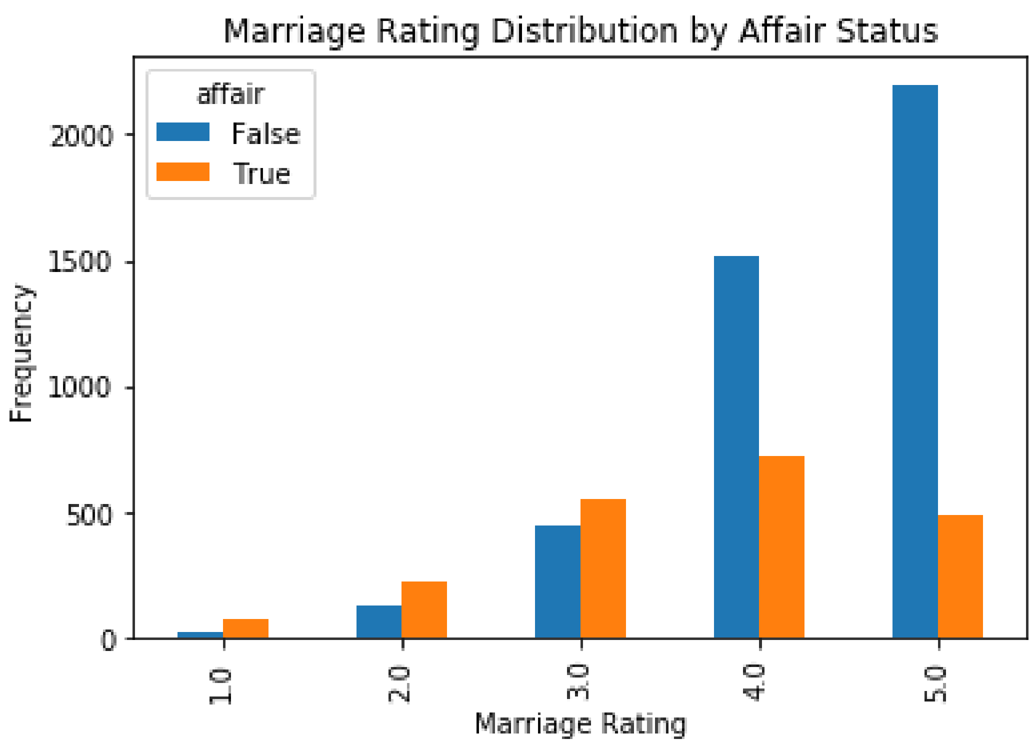


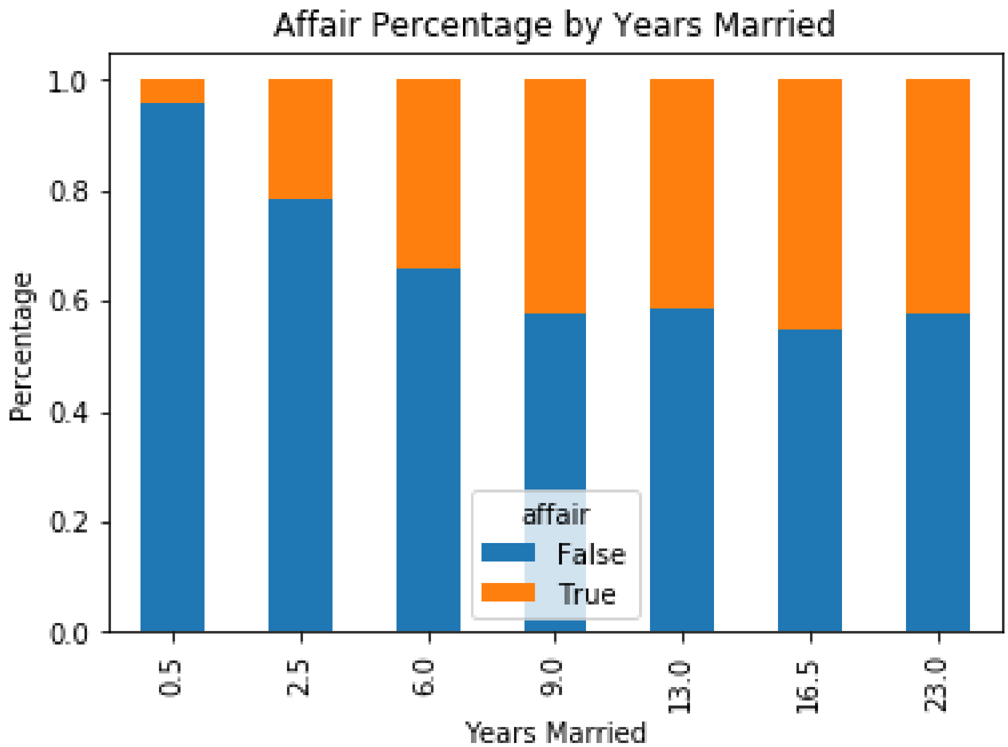
Histogram of Marriage Rating
Frequency
Marriage Rating
0
500
1000
1500
2000
2500
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0



Marriage Rating Distribution by Affair Status
affair
False
True
Frequency
Marriage Rating
0
500
1000
1500
2000
1.0
2.0
3.0
4.0
5.0


Affair Percentage by Years Married
1.2e+03
1.3e+02
3.9e+02
2.2e+02
affair
False
True
0
1
0
1
1000
800
600
400
200


Marriage Rating Distribution by Affair Status
affair
False
True
Frequency
2000
1500
1000
500
0
1.0
2.0
3.0
4.0
5.0
Marriage Rating



2.) Output variable -> y y -> Whether the client has subscribed a term deposit or not Binomial ("yes" or "no")

R code –

```
library(data.table)

library(MASS)


bank_data <- fread("bank_data.csv")

#View(bank_data)

summary(bank_data)


str(bank_data)


attach(bank_data)
```

```
y_model <- glm(y ~ age +balance + duration + campaign + pdays + previous + factor(default) +
factor(housing) + factor(loan)
            + factor(poutfailure) + factor(poutother) + factor(poutsuccess) + factor(poutunknown)
            + factor(con_cellular) + factor(con_telephone) + factor(con_unknown) + factor(divorced)
            + factor(married) + factor(single) + factor(joadmin.) + factor(joblue.collar) +
factor(joentrepreneur)
            + factor(johousemaid) + factor(jomanagement) + factor(joretired) + factor(joself.employed) +
factor(joservices)
            + factor(jostudent) + factor(jotechnician) + factor(jounemployed) + factor(jounknown), data =
bank_data)
summary(y_model)


library(MASS)
library(car)


stepAIC(y_model)


prob_y <- as.data.frame(predict(y_model, type = c("response"), bank_data))


final_y <- cbind(bank_data, prob_y)


confusion_y <- table(prob_y>0.5, bank_data$y)


table(prob_y>0.5)


confusion_y


accuracy_y <- sum(diag(confusion_y)/sum(confusion_y))
```

accuracy_y

Output –

```
> library(data.table)
data.table 1.12.2 using 2 threads (see ?getDTthreads).  Latest news: r-datata
ble.com
Warning message:
package ‘data.table’ was built under R version 3.4.4
> library(MASS)
Warning message:
package ‘MASS’ was built under R version 3.4.4
>
> bank_data <- fread("bank_data.csv")
> #View(bank_data)
> summary(bank_data)
      age           default            balance          housing            loa
n
 Min.   :18.00   Min.   :0.00000   Min.   : -8019   Min.   :0.0000   Min.   :
0.0000
 1st Qu.:33.00   1st Qu.:0.00000   1st Qu.:    72   1st Qu.:0.0000   1st Qu.:
0.0000
 Median :39.00   Median :0.00000   Median :   448   Median :1.0000   Median :
0.0000
 Mean   :40.94   Mean   :0.01803   Mean   :  1362   Mean   :0.5558   Mean   :
0.1602
 3rd Qu.:48.00   3rd Qu.:0.00000   3rd Qu.:  1428   3rd Qu.:1.0000   3rd Qu.:
0.0000
 Max.   :95.00   Max.   :1.00000   Max.   :102127   Max.   :1.0000   Max.   :
1.0000
    duration         campaign          pdays          previous          poutfa
ilure
 Min.   :   0.0   Min.   : 1.000   Min.   : -1.0   Min.   :  0.0000   Min.   
:0.0000
 1st Qu.: 103.0   1st Qu.: 1.000   1st Qu.: -1.0   1st Qu.:  0.0000   1st Qu.
:0.0000
 Median : 180.0   Median : 2.000   Median : -1.0   Median :  0.0000   Median 
:0.0000
 Mean   : 258.2   Mean   : 2.764   Mean   : 40.2   Mean   :  0.5803   Mean   
:0.1084
 3rd Qu.: 319.0   3rd Qu.: 3.000   3rd Qu.: -1.0   3rd Qu.:  0.0000   3rd Qu.
:0.0000
 Max.   :4918.0   Max.   :63.000   Max.   :871.0   Max.   :275.0000   Max.   
:1.0000
   poutother       poutsuccess        poutunknown      con_cellular     con_tel
ephone
 Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.0000   Min.   
:0.00000
 1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.
:0.00000
 Median :0.0000   Median :0.00000   Median :1.0000   Median :1.0000   Median 
:0.00000
```

```
 Mean   :0.0407   Mean   :0.03342   Mean   :0.8175   Mean   :0.6477   Mean   
:0.06428  
 3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.
:0.00000  
 Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :1.0000   Max.   
:1.00000  
  con_unknown       divorced         married           single          joadmi
n.       
 Min.   :0.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0
.0000  
 1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0
.0000  
 Median :0.000   Median :0.0000   Median :1.0000   Median :0.0000   Median :0
.0000  
 Mean   :0.288   Mean   :0.1152   Mean   :0.6019   Mean   :0.2829   Mean   :0
.1144  
 3rd Qu.:1.000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0
.0000  
 Max.   :1.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1
.0000  
 joblue.collar    joentrepreneur     johousemaid       jomanagement      jore
tired     
 Min.   :0.0000   Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   Min.   
:0.00000  
 1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu
.:0.00000  
 Median :0.0000   Median :0.00000   Median :0.00000   Median :0.0000   Median 
:0.00000  
 Mean   :0.2153   Mean   :0.03289   Mean   :0.02743   Mean   :0.2092   Mean   
:0.05008  
 3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu
.:0.00000  
 Max.   :1.0000   Max.   :1.00000   Max.   :1.00000   Max.   :1.0000   Max.   
:1.00000  
 joself.employed     joservices        jostudent         jotechnician     joune
mployed   
 Min.   :0.00000   Min.   :0.00000   Min.   :0.00000   Min.   :0.000   Min.   
:0.00000  
 1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.000   1st Qu
.:0.00000  
 Median :0.00000   Median :0.00000   Median :0.00000   Median :0.000   Median 
:0.00000  
 Mean   :0.03493   Mean   :0.09188   Mean   :0.02075   Mean   :0.168   Mean   
:0.02882  
 3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.000   3rd Qu
.:0.00000  
 Max.   :1.00000   Max.   :1.00000   Max.   :1.00000   Max.   :1.000   Max.   
:1.00000  
   jounknown              y        
 Min.   :0.00000   Min.   :0.000  
 1st Qu.:0.00000   1st Qu.:0.000  
 Median :0.00000   Median :0.000  
 Mean   :0.00637   Mean   :0.117  
 3rd Qu.:0.00000   3rd Qu.:0.000  
 Max.   :1.00000   Max.   :1.000  
> 
> str(bank_data)
```

```
Classes ‘data.table’ and 'data.frame':45211 obs. of  32 variables:
 $ age            : int  58 44 33 47 33 35 28 42 58 43 ...
 $ default        : int  0 0 0 0 0 0 0 1 0 0 ...
 $ balance        : int  2143 29 2 1506 1 231 447 2 121 593 ...
 $ housing        : int  1 1 1 1 0 1 1 1 1 1 ...
 $ loan           : int  0 0 1 0 0 0 1 0 0 0 ...
 $ duration       : int  261 151 76 92 198 139 217 380 50 55 ...
 $ campaign       : int  1 1 1 1 1 1 1 1 1 1 ...
 $ pdays          : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ previous       : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutfailure    : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutother      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutsuccess    : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutunknown    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ con_cellular   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ con_telephone  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ con_unknown    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ divorced       : int  0 0 0 0 0 0 0 1 0 0 ...
 $ married        : int  1 0 1 1 0 1 0 0 1 0 ...
 $ single         : int  0 1 0 0 1 0 1 0 0 1 ...
 $ joadmin.       : int  0 0 0 0 0 0 0 0 0 0 ...
 $ joblue.collar  : int  0 0 0 1 0 0 0 0 0 0 ...
 $ joentrepreneur : int  0 0 1 0 0 0 0 1 0 0 ...
 $ johousemaid    : int  0 0 0 0 0 0 0 0 0 0 ...
 $ jomanagement   : int  1 0 0 0 0 1 1 0 0 0 ...
 $ joretired      : int  0 0 0 0 0 0 0 0 1 0 ...
 $ joself.employed: int  0 0 0 0 0 0 0 0 0 0 ...
 $ joservices     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ jostudent      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ jotechnician   : int  0 1 0 0 0 0 0 0 0 1 ...
 $ jounemployed   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ jounknown      : int  0 0 0 0 1 0 0 0 0 0 ...
 $ y              : int  0 0 0 0 0 0 0 0 0 0 ...
 - attr(*, ".internal.selfref")=<externalptr>
>
> attach(bank_data)
The following object is masked from package:MASS:

    housing

>
> y_model <- glm(y ~ age +balance + duration + campaign + pdays + previous +
factor(default) + factor(housing) + factor(loan)
+                 + factor(poutfailure) + factor(poutother) + factor(poutsucce
ss) + factor(poutunknown)
+                 + factor(con_cellular) + factor(con_telephone) + factor(con_
unknown) + factor(divorced)
+                 + factor(married) + factor(single) + factor(joadmin.) + fact
or(joblue.collar) + factor(joentrepreneur)
+                 + factor(johousemaid) + factor(jomanagement) + factor(joreti
red) + factor(joself.employed) + factor(joservices)
+                 + factor(jostudent) + factor(jotechnician) + factor(jounempl
oyed) + factor(jounknown), data = bank_data)
> summary(y_model)

Call:
glm(formula = y ~ age + balance + duration + campaign + pdays +
```

```
    previous + factor(default) + factor(housing) + factor(loan) +
    factor(poutfailure) + factor(poutother) + factor(poutsuccess) +
    factor(poutunknown) + factor(con_cellular) + factor(con_telephone) +
    factor(con_unknown) + factor(divorced) + factor(married) +
    factor(single) + factor(joadmin.) + factor(joblue.collar) +
    factor(joentrepreneur) + factor(johousemaid) + factor(jomanagement) +
    factor(joretired) + factor(joself.employed) + factor(joservices) +
    factor(jostudent) + factor(jotechnician) + factor(jounemployed) +
    factor(jounknown), data = bank_data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.29345  -0.11582  -0.04883   0.01842   1.06743

Coefficients: (4 not defined because of singularities)
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)              -2.755e-02  1.776e-02  -1.552 0.120766
age                       1.741e-04  1.570e-04   1.109 0.267383
balance                   1.959e-06  4.303e-07   4.552 5.33e-06 ***
duration                  4.733e-04  5.038e-06  93.953  < 2e-16 ***
campaign                 -2.083e-03  4.219e-04  -4.936 8.02e-07 ***
pdays                    -2.589e-05  2.726e-05  -0.950 0.342170
previous                  1.213e-03  6.651e-04   1.824 0.068120 .
factor(default)1         -1.037e-02  9.753e-03  -1.063 0.287572
factor(housing)1         -5.666e-02  2.844e-03 -19.925  < 2e-16 ***
factor(loan)1            -3.314e-02  3.565e-03  -9.296  < 2e-16 ***
factor(poutfailure)1      2.984e-02  8.145e-03   3.664 0.000248 ***
factor(poutother)1        5.788e-02  9.544e-03   6.064 1.34e-09 ***
factor(poutsuccess)1      4.753e-01  8.896e-03  53.426  < 2e-16 ***
factor(poutunknown)1             NA         NA      NA       NA
factor(con_cellular)1     5.555e-02  3.137e-03  17.705  < 2e-16 ***
factor(con_telephone)1    4.941e-02  5.830e-03   8.474  < 2e-16 ***
factor(con_unknown)1             NA         NA      NA       NA
factor(divorced)1        -1.531e-02  4.850e-03  -3.156 0.001601 **
factor(married)1         -2.668e-02  3.305e-03  -8.074 7.00e-16 ***
factor(single)1                  NA         NA      NA       NA
factor(joadmin.)1         2.297e-02  1.672e-02   1.374 0.169376
factor(joblue.collar)1   -4.980e-03  1.652e-02  -0.301 0.763053
factor(joentrepreneur)1  -3.181e-03  1.774e-02  -0.179 0.857741
factor(johousemaid)1     -1.559e-02  1.796e-02  -0.868 0.385371
factor(jomanagement)1     2.133e-02  1.649e-02   1.294 0.195809
factor(joretired)1        6.599e-02  1.730e-02   3.814 0.000137 ***
factor(joself.employed)1  2.451e-03  1.764e-02   0.139 0.889468
factor(joservices)1       1.654e-03  1.683e-02   0.098 0.921715
factor(jostudent)1        1.132e-01  1.876e-02   6.034 1.61e-09 ***
factor(jotechnician)1     5.941e-03  1.656e-02   0.359 0.719819
factor(jounemployed)1     1.554e-02  1.791e-02   0.868 0.385450
factor(jounknown)1               NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.07511589)

    Null deviance: 4670.3  on 45210  degrees of freedom
Residual deviance: 3394.0  on 45183  degrees of freedom
AIC: 11294
```

```
Number of Fisher Scoring iterations: 2

>
> library(MASS)
> library(car)
Error in library(car) : there is no package called ‘car’
>
> stepAIC(y_model)
Start:  AIC=11294.49
y ~ age + balance + duration + campaign + pdays + previous +
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) +
    factor(poutother) + factor(poutsuccess) + factor(poutunknown) +
    factor(con_cellular) + factor(con_telephone) + factor(con_unknown) +
    factor(divorced) + factor(married) + factor(single) + factor(joadmin.) +
    factor(joblue.collar) + factor(joentrepreneur) + factor(johousemaid) +
    factor(jomanagement) + factor(joretired) + factor(joself.employed) +
    factor(joservices) + factor(jostudent) + factor(jotechnician) +
    factor(jounemployed) + factor(jounknown)


Step:  AIC=11294.49
y ~ age + balance + duration + campaign + pdays + previous +
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) +
    factor(poutother) + factor(poutsuccess) + factor(poutunknown) +
    factor(con_cellular) + factor(con_telephone) + factor(con_unknown) +
    factor(divorced) + factor(married) + factor(single) + factor(joadmin.) +
    factor(joblue.collar) + factor(joentrepreneur) + factor(johousemaid) +
    factor(jomanagement) + factor(joretired) + factor(joself.employed) +
    factor(joservices) + factor(jostudent) + factor(jotechnician) +
    factor(jounemployed)


Step:  AIC=11294.49
y ~ age + balance + duration + campaign + pdays + previous +
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) +
    factor(poutother) + factor(poutsuccess) + factor(poutunknown) +
    factor(con_cellular) + factor(con_telephone) + factor(con_unknown) +
    factor(divorced) + factor(married) + factor(joadmin.) + factor(joblue.col
lar) +
    factor(joentrepreneur) + factor(johousemaid) + factor(jomanagement) +
    factor(joretired) + factor(joself.employed) + factor(joservices) +
    factor(jostudent) + factor(jotechnician) + factor(jounemployed)


Step:  AIC=11294.49
y ~ age + balance + duration + campaign + pdays + previous +
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) +
    factor(poutother) + factor(poutsuccess) + factor(poutunknown) +
    factor(con_cellular) + factor(con_telephone) + factor(divorced) +
    factor(married) + factor(joadmin.) + factor(joblue.collar) +
    factor(joentrepreneur) + factor(johousemaid) + factor(jomanagement) +
    factor(joretired) + factor(joself.employed) + factor(joservices) +
    factor(jostudent) + factor(jotechnician) + factor(jounemployed)


Step:  AIC=11294.49
y ~ age + balance + duration + campaign + pdays + previous +
```

```
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) +
    factor(poutother) + factor(poutsuccess) + factor(con_cellular) +
    factor(con_telephone) + factor(divorced) + factor(married) +
    factor(joadmin.) + factor(joblue.collar) + factor(joentrepreneur) +
    factor(johousemaid) + factor(jomanagement) + factor(joretired) +
    factor(joself.employed) + factor(joservices) + factor(jostudent) +
    factor(jotechnician) + factor(jounemployed)

                          Df Deviance   AIC
- factor(joservices)       1   3394.0 11292
- factor(joself.employed)  1   3394.0 11292
- factor(joentrepreneur)   1   3394.0 11292
- factor(joblue.collar)    1   3394.0 11293
- factor(jotechnician)     1   3394.0 11293
- factor(jounemployed)     1   3394.0 11293
- factor(johousemaid)      1   3394.0 11293
- pdays                    1   3394.0 11293
- factor(default)          1   3394.0 11294
- age                      1   3394.1 11294
- factor(jomanagement)     1   3394.1 11294
- factor(joadmin.)         1   3394.1 11294
<none>                         3394.0 11294
- previous                 1   3394.2 11296
- factor(divorced)         1   3394.7 11302
- factor(poutfailure)      1   3395.0 11306
- factor(joretired)        1   3395.1 11307
- balance                  1   3395.5 11313
- campaign                 1   3395.8 11317
- factor(jostudent)        1   3396.7 11329
- factor(poutother)        1   3396.7 11329
- factor(married)          1   3398.9 11358
- factor(con_telephone)    1   3399.4 11364
- factor(loan)             1   3400.5 11379
- factor(con_cellular)     1   3417.5 11605
- factor(housing)          1   3423.8 11688
- factor(poutsuccess)      1   3608.4 14062
- duration                 1   4057.0 19361

Step:  AIC=11292.5
y ~ age + balance + duration + campaign + pdays + previous +
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) +
    factor(poutother) + factor(poutsuccess) + factor(con_cellular) +
    factor(con_telephone) + factor(divorced) + factor(married) +
    factor(joadmin.) + factor(joblue.collar) + factor(joentrepreneur) +
    factor(johousemaid) + factor(jomanagement) + factor(joretired) +
    factor(joself.employed) + factor(jostudent) + factor(jotechnician) +
    factor(jounemployed)

                          Df Deviance   AIC
- factor(joself.employed)  1   3394.0 11290
- factor(joentrepreneur)   1   3394.0 11291
- factor(jotechnician)     1   3394.0 11291
- pdays                    1   3394.0 11291
- factor(default)          1   3394.0 11292
- age                      1   3394.1 11292
- factor(joblue.collar)    1   3394.1 11292
<none>                         3394.0 11292
```

```
- factor(jounemployed)     1   3394.2 11293
- previous                 1   3394.2 11294
- factor(johousemaid)      1   3394.2 11294
- factor(divorced)         1   3394.7 11300
- factor(poutfailure)      1   3395.0 11304
- factor(joadmin.)         1   3395.1 11305
- factor(jomanagement)     1   3395.1 11306
- balance                  1   3395.5 11311
- campaign                 1   3395.8 11315
- factor(poutother)        1   3396.7 11327
- factor(married)          1   3398.9 11356
- factor(joretired)        1   3399.1 11359
- factor(con_telephone)    1   3399.4 11362
- factor(loan)             1   3400.5 11377
- factor(jostudent)        1   3402.9 11410
- factor(con_cellular)     1   3417.5 11603
- factor(housing)          1   3424.0 11688
- factor(poutsuccess)      1   3608.4 14060
- duration                 1   4057.0 19359

Step:  AIC=11290.51
y ~ age + balance + duration + campaign + pdays + previous + 
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) + 
    factor(poutother) + factor(poutsuccess) + factor(con_cellular) + 
    factor(con_telephone) + factor(divorced) + factor(married) + 
    factor(joadmin.) + factor(joblue.collar) + factor(joentrepreneur) + 
    factor(johousemaid) + factor(jomanagement) + factor(joretired) + 
    factor(jostudent) + factor(jotechnician) + factor(jounemployed)

                         Df Deviance   AIC
- factor(joentrepreneur)  1   3394.0 11289
- factor(jotechnician)    1   3394.0 11289
- pdays                   1   3394.0 11289
- factor(default)         1   3394.0 11290
- age                     1   3394.1 11290
<none>                        3394.0 11290
- factor(joblue.collar)   1   3394.1 11291
- factor(jounemployed)    1   3394.2 11291
- previous                1   3394.2 11292
- factor(johousemaid)     1   3394.3 11293
- factor(divorced)        1   3394.7 11298
- factor(poutfailure)     1   3395.0 11302
- factor(joadmin.)        1   3395.2 11305
- factor(jomanagement)    1   3395.3 11307
- balance                 1   3395.5 11309
- campaign                1   3395.8 11313
- factor(poutother)       1   3396.7 11325
- factor(married)         1   3398.9 11354
- factor(con_telephone)   1   3399.4 11360
- factor(joretired)       1   3399.5 11362
- factor(loan)            1   3400.5 11375
- factor(jostudent)       1   3403.3 11413
- factor(con_cellular)    1   3417.6 11602
- factor(housing)         1   3424.0 11687
- factor(poutsuccess)     1   3608.4 14059
- duration                1   4057.1 19357
```

```
Step:  AIC=11288.9
y ~ age + balance + duration + campaign + pdays + previous + 
    factor(default) + factor(housing) + factor(loan) + factor(poutfailure) + 
    factor(poutother) + factor(poutsuccess) + factor(con_cellular) + 
    factor(con_telephone) + factor(divorced) + factor(married) + 
    factor(joadmin.) + factor(joblue.collar) + factor(johousemaid) + 
    factor(jomanagement) + factor(joretired) + factor(jostudent) + 
    factor(jotechnician) + factor(jounemployed)

                        Df Deviance   AIC
- pdays                  1   3394.1 11288
- factor(default)        1   3394.1 11288
- age                    1   3394.1 11288
- factor(jotechnician)   1   3394.1 11288
- factor(joblue.collar)  1   3394.1 11289
<none>                       3394.0 11289
- factor(jounemployed)   1   3394.2 11290
- previous               1   3394.2 11290
- factor(johousemaid)    1   3394.3 11291
- factor(divorced)       1   3394.7 11297
- factor(poutfailure)    1   3395.0 11300
- factor(joadmin.)       1   3395.5 11307
- balance                1   3395.5 11308
- factor(jomanagement)   1   3395.7 11310
- campaign               1   3395.8 11311
- factor(poutother)      1   3396.8 11324
- factor(married)        1   3398.9 11352
- factor(con_telephone)  1   3399.4 11359
- factor(joretired)      1   3400.0 11367
- factor(loan)           1   3400.5 11374
- factor(jostudent)      1   3403.7 11416
- factor(con_cellular)   1   3417.6 11600
- factor(housing)        1   3424.0 11686
- factor(poutsuccess)    1   3608.5 14057
- duration               1   4057.1 19355

Step:  AIC=11287.8
y ~ age + balance + duration + campaign + previous + factor(default) + 
    factor(housing) + factor(loan) + factor(poutfailure) + factor(poutother) 
+ 
    factor(poutsuccess) + factor(con_cellular) + factor(con_telephone) + 
    factor(divorced) + factor(married) + factor(joadmin.) + factor(joblue.col
lar) + 
    factor(johousemaid) + factor(jomanagement) + factor(joretired) + 
    factor(jostudent) + factor(jotechnician) + factor(jounemployed)

                        Df Deviance   AIC
- factor(default)        1   3394.1 11287
- age                    1   3394.2 11287
- factor(jotechnician)   1   3394.2 11287
- factor(joblue.collar)  1   3394.2 11288
<none>                       3394.1 11288
- factor(jounemployed)   1   3394.3 11289
- previous               1   3394.3 11289
- factor(johousemaid)    1   3394.3 11290
- factor(divorced)       1   3394.8 11296
- factor(joadmin.)       1   3395.6 11306
```

```
- balance                  1   3395.6 11307
- factor(jomanagement)     1   3395.8 11309
- factor(poutfailure)      1   3395.9 11310
- campaign                 1   3395.9 11310
- factor(poutother)        1   3398.0 11338
- factor(married)          1   3399.0 11351
- factor(con_telephone)    1   3399.4 11357
- factor(joretired)        1   3400.1 11366
- factor(loan)             1   3400.6 11372
- factor(jostudent)        1   3403.8 11415
- factor(con_cellular)     1   3417.6 11598
- factor(housing)          1   3424.6 11691
- factor(poutsuccess)      1   3681.4 14960
- duration                 1   4057.2 19354

Step:  AIC=11286.97
y ~ age + balance + duration + campaign + previous + factor(housing) +
    factor(loan) + factor(poutfailure) + factor(poutother) +
    factor(poutsuccess) + factor(con_cellular) + factor(con_telephone) +
    factor(divorced) + factor(married) + factor(joadmin.) + factor(joblue.col
lar) +
    factor(johousemaid) + factor(jomanagement) + factor(joretired) +
    factor(jostudent) + factor(jotechnician) + factor(jounemployed)

                        Df Deviance   AIC
- age                    1   3394.2 11286
- factor(jotechnician)   1   3394.3 11286
- factor(joblue.collar)  1   3394.3 11287
<none>                       3394.1 11287
- factor(jounemployed)   1   3394.4 11288
- previous               1   3394.4 11288
- factor(johousemaid)    1   3394.4 11289
- factor(divorced)       1   3394.9 11295
- factor(joadmin.)       1   3395.7 11305
- balance                1   3395.8 11307
- factor(jomanagement)   1   3395.9 11308
- factor(poutfailure)    1   3396.0 11310
- campaign               1   3396.0 11310
- factor(poutother)      1   3398.1 11338
- factor(married)        1   3399.0 11350
- factor(con_telephone)  1   3399.6 11357
- factor(joretired)      1   3400.2 11366
- factor(loan)           1   3400.8 11374
- factor(jostudent)      1   3403.9 11415
- factor(con_cellular)   1   3417.7 11598
- factor(housing)        1   3424.7 11690
- factor(poutsuccess)    1   3681.8 14963
- duration               1   4057.4 19355

Step:  AIC=11286.21
y ~ balance + duration + campaign + previous + factor(housing) +
    factor(loan) + factor(poutfailure) + factor(poutother) +
    factor(poutsuccess) + factor(con_cellular) + factor(con_telephone) +
    factor(divorced) + factor(married) + factor(joadmin.) + factor(joblue.col
lar) +
    factor(johousemaid) + factor(jomanagement) + factor(joretired) +
    factor(jostudent) + factor(jotechnician) + factor(jounemployed)
```

```
                         Df Deviance   AIC
- factor(jotechnician)   1   3394.3 11286
- factor(joblue.collar)  1   3394.4 11286
<none>                       3394.2 11286
- factor(jounemployed)   1   3394.5 11287
- factor(johousemaid)    1   3394.5 11288
- previous               1   3394.5 11288
- factor(divorced)       1   3394.9 11293
- factor(joadmin.)       1   3395.7 11304
- balance                1   3395.9 11307
- factor(jomanagement)   1   3396.0 11308
- campaign               1   3396.1 11309
- factor(poutfailure)    1   3396.1 11309
- factor(poutother)      1   3398.2 11337
- factor(married)        1   3399.4 11353
- factor(con_telephone)  1   3399.9 11359
- factor(loan)           1   3400.9 11373
- factor(joretired)      1   3402.0 11388
- factor(jostudent)      1   3403.9 11413
- factor(con_cellular)   1   3417.7 11596
- factor(housing)        1   3426.1 11706
- factor(poutsuccess)    1   3682.3 14967
- duration               1   4057.4 19353

Step:  AIC=11285.57
y ~ balance + duration + campaign + previous + factor(housing) + 
    factor(loan) + factor(poutfailure) + factor(poutother) + 
    factor(poutsuccess) + factor(con_cellular) + factor(con_telephone) + 
    factor(divorced) + factor(married) + factor(joadmin.) + factor(joblue.col
lar) + 
    factor(johousemaid) + factor(jomanagement) + factor(joretired) + 
    factor(jostudent) + factor(jounemployed)

                         Df Deviance   AIC
<none>                       3394.3 11286
- factor(jounemployed)   1   3394.5 11286
- previous               1   3394.6 11287
- factor(johousemaid)    1   3394.7 11288
- factor(joblue.collar)  1   3394.8 11289
- factor(divorced)       1   3395.0 11293
- factor(joadmin.)       1   3395.8 11303
- balance                1   3396.0 11306
- campaign               1   3396.2 11308
- factor(poutfailure)    1   3396.2 11308
- factor(jomanagement)   1   3396.2 11309
- factor(poutother)      1   3398.3 11336
- factor(married)        1   3399.6 11353
- factor(con_telephone)  1   3400.0 11358
- factor(loan)           1   3401.1 11373
- factor(joretired)      1   3402.4 11391
- factor(jostudent)      1   3404.1 11414
- factor(con_cellular)   1   3418.0 11598
- factor(housing)        1   3426.3 11707
- factor(poutsuccess)    1   3682.4 14967
- duration               1   4057.4 19351
```

```
Call:  glm(formula = y ~ balance + duration + campaign + previous + 
    factor(housing) + factor(loan) + factor(poutfailure) + factor(poutother) 
+ 
    factor(poutsuccess) + factor(con_cellular) + factor(con_telephone) + 
    factor(divorced) + factor(married) + factor(joadmin.) + factor(joblue.col
lar) + 
    factor(johousemaid) + factor(jomanagement) + factor(joretired) + 
    factor(jostudent) + factor(jounemployed), data = bank_data)

Coefficients:
           (Intercept)                  balance                 duration  
            -1.793e-02                2.030e-06                4.733e-04  
              campaign                 previous         factor(housing)1  
            -2.085e-03                1.231e-03               -5.741e-02  
         factor(loan)1     factor(poutfailure)1       factor(poutother)1  
            -3.358e-02                2.377e-02                5.201e-02  
  factor(poutsuccess)1    factor(con_cellular)1   factor(con_telephone)1  
             4.713e-01                5.554e-02                5.011e-02  
     factor(divorced)1        factor(married)1        factor(joadmin.)1  
            -1.379e-02               -2.550e-02                1.958e-02  
factor(joblue.collar)1     factor(johousemaid)1    factor(jomanagement)1  
            -8.475e-03               -1.816e-02                1.807e-02  
    factor(joretired)1      factor(jostudent)1    factor(jounemployed)1  
             6.591e-02                1.083e-01                1.215e-02  

Degrees of Freedom: 45210 Total (i.e. Null);  45190 Residual
Null Deviance:	    4670 
Residual Deviance: 3394 	AIC: 11290
> 
> prob_y <- as.data.frame(predict(y_model, type = c("response"), bank_data))
Warning message:
In predict.lm(object, newdata, se.fit, scale = 1, type = ifelse(type ==  :
  prediction from a rank-deficient fit may be misleading
> 
> final_y <- cbind(bank_data, prob_y)
> 
> confusion_y <- table(prob_y>0.5, bank_data$y)
> 
> table(prob_y>0.5)

FALSE  TRUE 
42986  2225 
> 
> confusion_y
       
            0     1
  FALSE 39150  3836
  TRUE    772  1453
> 
> accuracy_y <- sum(diag(confusion_y)/sum(confusion_y))
> 
> accuracy_y
[1] 0.8980779
```

Python Code –

```
import pandas as pd
import numpy as np
from sklearn import preprocessing
import matplotlib.pyplot as plt
plt.rc("font", size=14)
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
import seaborn as sns
sns.set(style="white")
sns.set(style="whitegrid", color_codes=True)

data = pd.read_csv("bank_data.csv",sep=';')
print(data.head())

data = data.dropna()
print(data.shape)
print(list(data.columns))

print(data.isnull().sum())

data.drop(data.columns[[0, 3, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19]], axis=1, inplace=True)
data2 = pd.get_dummies(data, columns =['job', 'marital', 'default', 'housing', 'loan', 'poutcome'])

data2.drop(data2.columns[[12, 16, 18, 21, 24]], axis=1, inplace=True)
data2.columns
```

```
sns.heatmap(data2.corr())
plt.show()

X = data2.iloc[:,1:]
y = data2.iloc[:,0]
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

X_train.shape

classifier = LogisticRegression(random_state=0)
classifier.fit(X_train, y_train)

y_pred = classifier.predict(X_test)
from sklearn.metrics import confusion_matrix
confusion_matrix = confusion_matrix(y_test, y_pred)
print(confusion_matrix)

print('Accuracy of logistic regression classifier on test set: {:.2f}'.format(classifier.score(X_test, y_test)))

from sklearn.metrics import classification_report
print(classification_report(y_test, y_pred))

from sklearn.decomposition import PCA
X = data2.iloc[:,1:]
y = data2.iloc[:,0]
pca = PCA(n_components=2).fit_transform(X)
X_train, X_test, y_train, y_test = train_test_split(pca, y, random_state=0)

plt.figure(dpi=120)
```

```
plt.scatter(pca[y.values==0,0], pca[y.values==0,1], alpha=0.5, label='YES', s=2, color='navy')

plt.scatter(pca[y.values==1,0], pca[y.values==1,1], alpha=0.5, label='NO', s=2, color='darkorange')

plt.legend()

plt.title('Bank Marketing Data Set\nFirst Two Principal Components')

plt.xlabel('PC1')

plt.ylabel('PC2')

plt.gca().set_aspect('equal')

plt.show()


def plot_bank(X, y, fitted_model):

plt.figure(figsize=(9.8,5), dpi=100)

  for i, plot_type in enumerate(['Decision Boundary', 'Decision Probabilities']):

    plt.subplot(1,2,i+1)

mesh_step_size = 0.01  # step size in the mesh

    x_min, x_max = X[:, 0].min() - .1, X[:, 0].max() + .1

    y_min, y_max = X[:, 1].min() - .1, X[:, 1].max() + .1

    xx, yy = np.meshgrid(np.arange(x_min, x_max, mesh_step_size), np.arange(y_min, y_max,
mesh_step_size))

    if i == 0:

      Z = fitted_model.predict(np.c_[xx.ravel(), yy.ravel()])

    else:

      try:

        Z = fitted_model.predict_proba(np.c_[xx.ravel(), yy.ravel()])[:,1]

      except:

        plt.text(0.4, 0.5, 'Probabilities Unavailable', horizontalalignment='center',

           verticalalignment='center', transform = plt.gca().transAxes, fontsize=12)

        plt.axis('off')

        break

    Z = Z.reshape(xx.shape)
```

```
        plt.scatter(X[y.values==0,0], X[y.values==0,1], alpha=0.8, label='YES', s=5, color='navy')
        plt.scatter(X[y.values==1,0], X[y.values==1,1], alpha=0.8, label='NO', s=5, color='darkorange')
        plt.imshow(Z, interpolation='nearest', cmap='RdYlBu_r', alpha=0.15,
                   extent=(x_min, x_max, y_min, y_max), origin='lower')
        plt.title(plot_type + '\n' +
                  str(fitted_model).split('(')[0]+ ' Test Accuracy: ' + str(np.round(fitted_model.score(X, y), 5)))
        plt.gca().set_aspect('equal');
    plt.tight_layout()
    plt.legend()
    plt.subplots_adjust(top=0.9, bottom=0.08, wspace=0.02)
model = LogisticRegression()
model.fit(X_train,y_train)
plot_bank(X_test, y_test, model)
plt.show()
```

3.) Suppose we are interested in the factors that influence whether a political candidate wins an election. The outcome (response) variable is binary (0/1); win or lose. The predictor variables of interest are the amount of money spent on the campaign, the amount of time spent campaigning negatively and whether or not the candidate is an incumbent.

R code –

```
library(data.table)

election_data <- fread("election_data.csv")

#View(election_data)
setkey(election_data,`Election-id`)
summary(election_data)
```

```
colnames(election_data)

plot(election_data)

attach(election_data)
election_response <- glm(Result ~ Year+`Amount Spent`+`Popularity Rank`, data = election_data)
summary(election_response)

# Residual Deviance is less than Null Deviance that's mean input variable are significance.

library(MASS)
stepAIC(election_response) # Checking best fit model

exp(coef(election_response))

# Creating COnfusion matrix to check the accuracy

prob <- as.data.frame(predict(election_response, type = c("response"), election_data))

final <- cbind(election_data,prob)

confusion <- table(prob>0.5, election_data$Result)
table(prob>0.5)

confusion



Accuracy <- sum(diag(confusion)/sum(confusion))
```

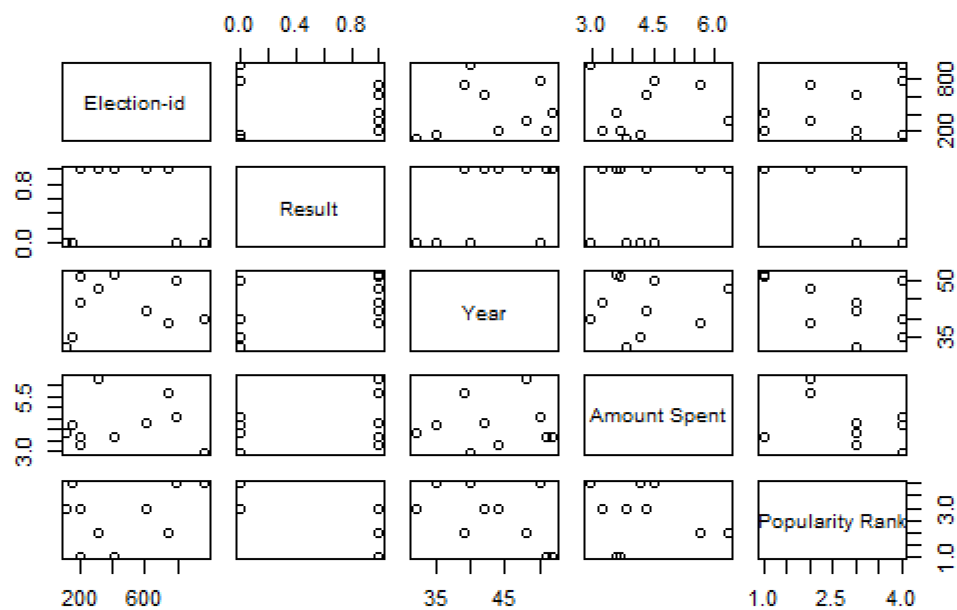
Accuracy

Output –

```
> library(data.table)
>
> election_data <- fread("election_data.csv")
>
> #View(election_data)
> setkey(election_data,`Election-id`)
> summary(election_data)
  Election-id        Result          Year         Amount Spent    Popularity Ran
k
 Min.   :122.0   Min.   :0.0   Min.   :32.00   Min.   :2.930   Min.   :1.00
 1st Qu.:202.2   1st Qu.:0.0   1st Qu.:39.25   1st Qu.:3.618   1st Qu.:2.00
 Median :362.5   Median :1.0   Median :43.00   Median :4.005   Median :3.00
 Mean   :451.6   Mean   :0.6   Mean   :43.30   Mean   :4.229   Mean   :2.70
 3rd Qu.:710.2   3rd Qu.:1.0   3rd Qu.:49.50   3rd Qu.:4.470   3rd Qu.:3.75
 Max.   :965.0   Max.   :1.0   Max.   :52.00   Max.   :6.320   Max.   :4.00
>
> colnames(election_data)
[1] "Election-id"     "Result"          "Year"            "Amount Spent"    "
Popularity Rank"
>
> plot(election_data)
>
> attach(election_data)
> election_response <- glm(Result ~ Year+`Amount Spent`+`Popularity Rank`, da
ta = election_data)
> summary(election_response)

Call:
glm(formula = Result ~ Year + `Amount Spent` + `Popularity Rank`,
    data = election_data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.36265  -0.15265  -0.09902   0.08992   0.55615

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        0.65329    1.31682   0.496   0.6375
Year               0.01021    0.02151   0.475   0.6517
`Amount Spent`     0.07523    0.12208   0.616   0.5604
`Popularity Rank` -0.30137    0.13057  -2.308   0.0604 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.1432053)

    Null deviance: 2.40000  on 9  degrees of freedom
```

```
Residual deviance: 0.85923  on 6  degrees of freedom
AIC: 13.836

Number of Fisher Scoring iterations: 2

>
> # Residual Deviance is less than Null Deviance that's mean input variable a
re significance.
>
> library(MASS)
> stepAIC(election_response) # Checking best fit model
Start:  AIC=13.84
Result ~ Year + `Amount Spent` + `Popularity Rank`

                    Df Deviance    AIC
- Year               1  0.89152 12.205
- `Amount Spent`     1  0.91361 12.449
<none>                  0.85923 13.836
- `Popularity Rank`  1  1.62217 18.191

Step:  AIC=12.2
Result ~ `Amount Spent` + `Popularity Rank`

                    Df Deviance    AIC
- `Amount Spent`     1  0.94215 10.757
<none>                  0.89152 12.205
- `Popularity Rank`  1  2.18851 19.185

Step:  AIC=10.76
Result ~ `Popularity Rank`

                    Df Deviance    AIC
<none>                  0.94215 10.757
- `Popularity Rank`  1  2.40000 18.108

Call:  glm(formula = Result ~ `Popularity Rank`, data = election_data)

Coefficients:
      (Intercept)  `Popularity Rank`
           1.5372            -0.3471

Degrees of Freedom: 9 Total (i.e. Null);  8 Residual
Null Deviance:     2.4
Residual Deviance: 0.9421     AIC: 10.76
>
> exp(coef(election_response))
      (Intercept)              Year    `Amount Spent` `Popularity Rank`
        1.9218592         1.0102668         1.0781268         0.7398019
>
> # Creating Confusion matrix to check the accuracy
>
> prob <- as.data.frame(predict(election_response, type = c("response"), elec
tion_data))
>
> final <- cbind(election_data,prob)
>
> confusion <- table(prob>0.5, election_data$Result)
```

```
> table(prob>0.5)

FALSE  TRUE
    5     5
>
> confusion

        0 1
  FALSE 4 1
  TRUE  0 5
>
>
> Accuracy <- sum(diag(confusion)/sum(confusion))
>
> Accuracy
[1] 0.9
```



Python Code –

import numpy as np

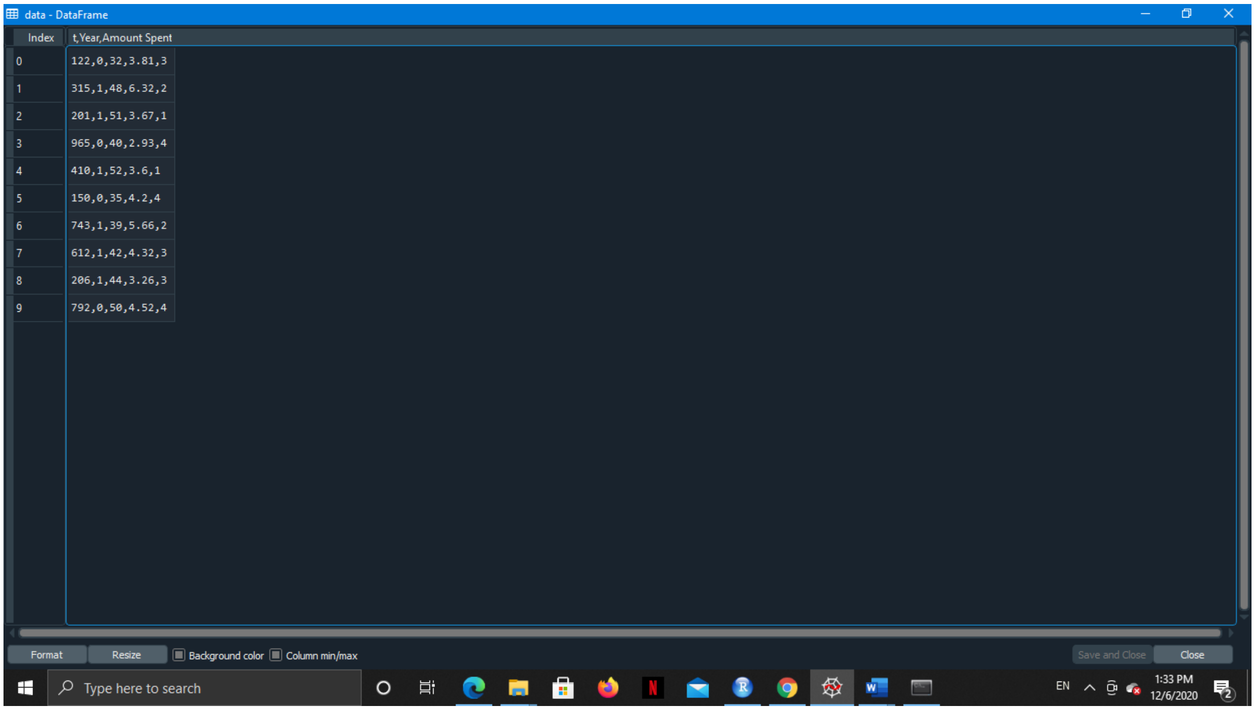
import pandas as pd

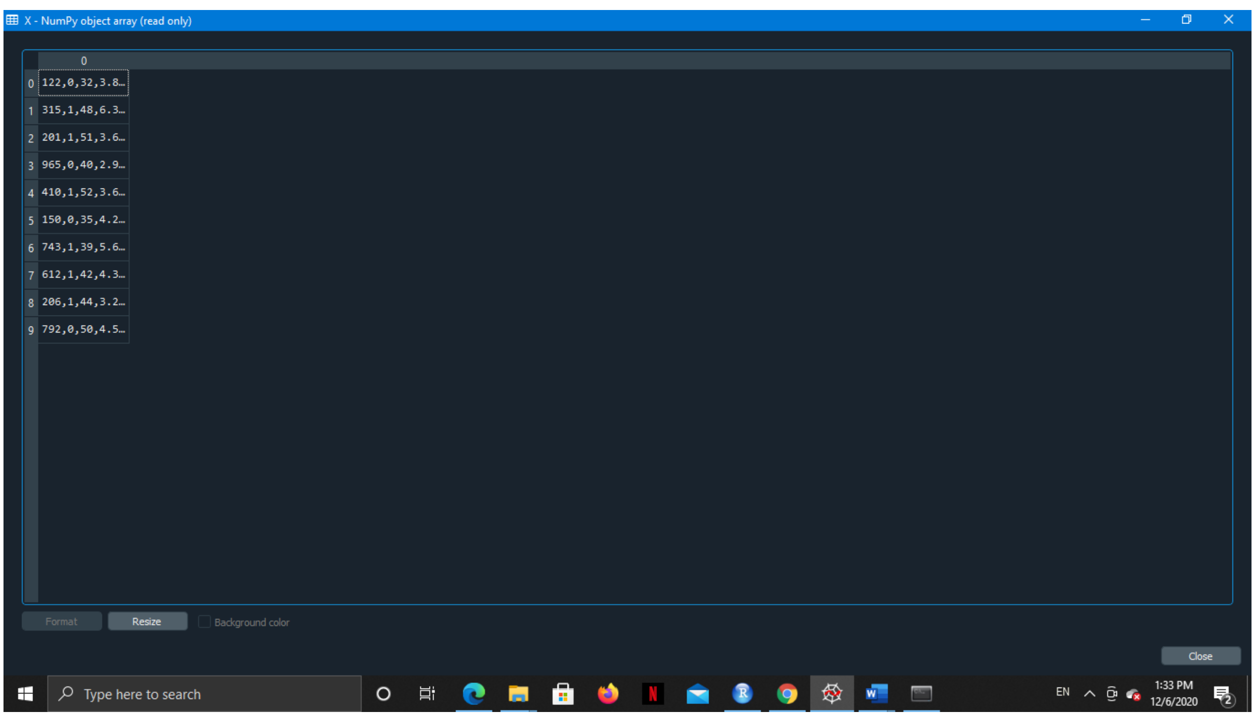
import statsmodels.api as sm

```
import matplotlib.pyplot as plt
from patsy import dmatrices
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split as split
from sklearn import metrics
from sklearn.model_selection import cross_val_score

data = pd.read_csv("election_data.csv",sep=';')
print(data.head())

print(data.head())

data = data.dropna()
print(data.shape)
print(list(data.columns))

print(data.isnull().sum())

X = data.iloc[:, :1].values
y = data.iloc[:, 9].values
# evaluate the model by splitting the data-set into train and test sets
X_train, X_test, y_train, y_test = split(X, y, test_size=0.3)

model2 = LogisticRegression()
model2.fit(X_train, y_train)



predicted = model2.predict(X_test)
print(y_test)
```

```
predicted


# generate class probabilities
probs = model2.predict_proba(X_test)
probs


# generate evaluation metrics
print(metrics.accuracy_score(y_test, predicted))
print(metrics.roc_auc_score(y_test, probs[:, 1]))

import seaborn as sns
conf_matrix = metrics.confusion_matrix(y_test, predicted)
sns.heatmap(conf_matrix, annot=True,cmap='Blues')

print(metrics.classification_report(y_test, predicted))
```

Output –


data - DataFrame
Index
t,Year,Amount Spent
0 122,0,32,3.81,3
1 315,1,48,6.32,2
2 201,1,51,3.67,1
3 965,0,40,2.93,4
4 410,1,52,3.6,1
5 150,0,35,4.2,4
6 743,1,39,5.66,2
7 612,1,42,4.32,3
8 206,1,44,3.26,3
9 792,0,50,4.52,4
Format
Resize
Background color
Column min/max
Save and Close
Close
Type here to search
EN
1:33 PM
12/6/2020


X - NumPy object array (read only)
0
0 122,0,32,3.8…
1 315,1,48,6.3…
2 201,1,51,3.6…
3 965,0,40,2.9…
4 410,1,52,3.6…
5 150,0,35,4.2…
6 743,1,39,5.6…
7 612,1,42,4.3…
8 206,1,44,3.2…
9 792,0,50,4.5…
Format
Resize
Background color
Close
Type here to search
EN
1:33 PM
12/6/2020