Machine Learning Unit Summary

Yogindra Raghav

December 7, 2018

Load all the necessary packages.

```
library(NHANES)
library(dplyr)
library(mdsr)
library(glmnet)
library(rpart)
library(rpart.plot)
```

Package `NHANES` contains survey data collected by the US National Center for Health Statistics (NCHS) which has conducted a series of health and nutrition surveys since the early 1960's. Since 1999 approximately 5,000 individuals of all ages are interviewed in their homes every year and complete the health examination component of the survey. The health examination is conducted in a mobile examination centre (MEC).

Marijuana $\implies$ hard drugs?

Are people who have tried marijuana more likely to use hard drugs? In the `NHANES` data set, the variable `Marijuana` and `HardDrugs` are defined as follows:

- `Marijuana`: Participant has tried marijuana. Reported for participants aged 18 to 59 years as Yes or No.
- `HardDrugs`: Participant has tried cocaine, crack cocaine, heroin or methamphetamine. Reported for participants aged 18 to 69 years as Yes or No.

To simplify the problem, we will only use 7 variables (predictors) in the data set:

Gender, HealthGen, Depressed, SleepHrsNight, AlcoholDay, SmokeNow, Marijuana.

You may use `?NHANES` to check the definition of each variable.

Models that you should build are:

- Logistic regression model
- Decision tree

Step 1: Select the variables and remove observations of missing values

```
data1<- NHANES %>%
  select(HardDrugs,Gender,HealthGen,Depressed,SleepHrsNight,AlcoholDay, SmokeNow, Marijuana) %>%
  na.omit()
glimpse(data1)
```

```
## Observations: 1,704
## Variables: 8
## $ HardDrugs     <fct> Yes, No, No, Yes, Yes, No, Yes, Yes, Yes, No, No...
## $ Gender        <fct> female, male, female, male, male, male, male, fe...
## $ HealthGen     <fct> Good, Fair, Excellent, Fair, Excellent, Excellen...
## $ Depressed     <fct> Several, None, None, None, Several, None, None, ...
## $ SleepHrsNight <int> 8, 6, 8, 6, 6, 6, 8, 4, 4, 6, 6, 6, 6, 6, 5, 8, ...
## $ AlcoholDay    <int> 2, 3, 1, 6, 3, 12, 5, 3, 3, 7, 7, 7, 7, 1, 4, 1,...
## $ SmokeNow      <fct> Yes, No, No, Yes, Yes, Yes, Yes, Yes, Yes, Yes, ...
## $ Marijuana     <fct> Yes, No, Yes, No, Yes, Yes, Yes, Yes, Yes, Yes, ...
```

Step 2: Create a distribution table for `HardDrugs`.

Show percentage of each category.

```
tally(~HardDrugs, data = NHANES, format = "percent")
```

```
## HardDrugs
##    No   Yes  <NA>
## 47.00 10.65 42.35
```

Step 3: Split the data to training and testing sets.

Just as what we have done before, 80% training and 20% testing.

```
set.seed(99)
train1 <- data1 %>% sample_frac(size = 0.8)
test1 <- data1 %>% setdiff(train1)
```

Model 1. Logistic Regression Model

We will do a regularized logistic regression model. So, first, use the `glmnet` package to find the `optimal` lambda value through cross-valuation ($k = 10$, default). Plot the fit result of cross-valuation.

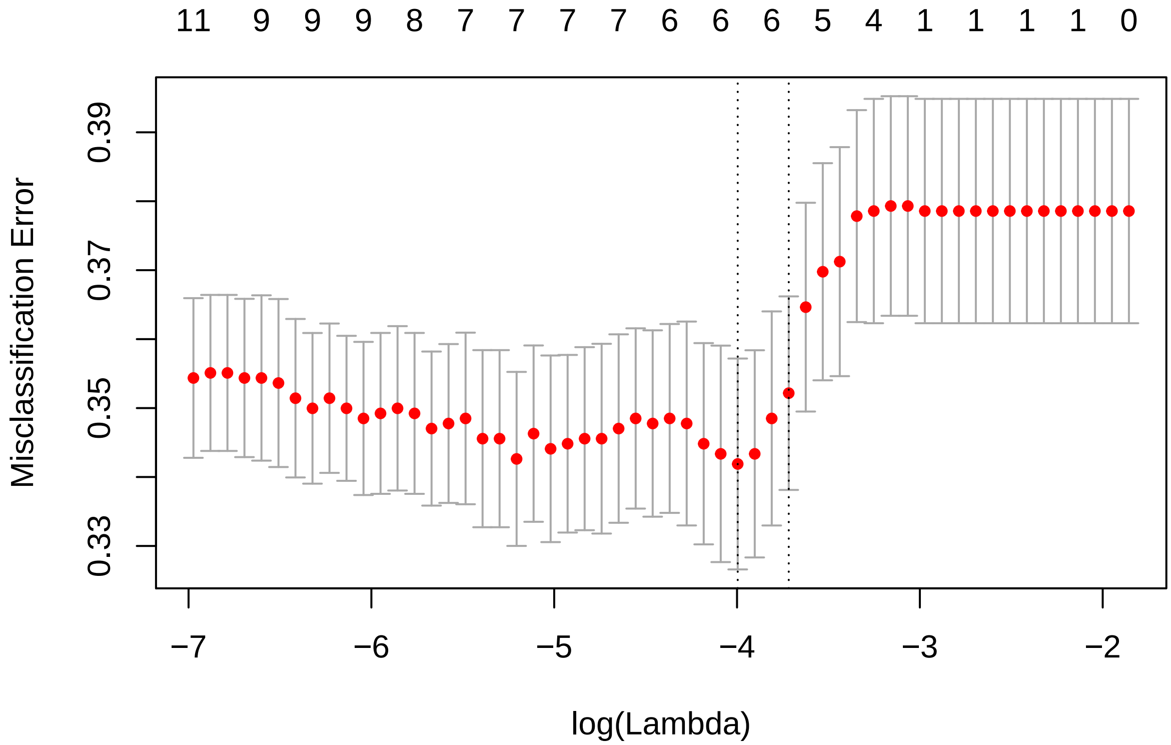
```
formula = as.formula("HardDrugs~ Gender + HealthGen + Depressed + SleepHrsNight + AlcoholDay + SmokeNow

predictors = model.matrix(formula, data = train1)

fit = cv.glmnet(predictors, train1$HardDrugs, family = "binomial", type = "class")

fit
```

```
## $lambda
##  [1] 0.1562243785 0.1423458353 0.1297002236 0.1181780132 0.1076794043
##  [6] 0.0981134628 0.0893973332 0.0814555205 0.0742192366 0.0676258041
## [11] 0.0616181141 0.0561441307 0.0511564408 0.0466118436 0.0424709759
## [16] 0.0386979715 0.0352601503 0.0321277358 0.0292735963 0.0266730107
## [21] 0.0243034539 0.0221444021 0.0201771544 0.0183846716 0.0167514281
## [26] 0.0152632774 0.0139073299 0.0126718410 0.0115461096 0.0105203850
## [31] 0.0095857830 0.0087342085 0.0079582855 0.0072512933 0.0066071084
## [36] 0.0060201511 0.0054853375 0.0049980352 0.0045540235 0.0041494566
## [41] 0.0037808304 0.0034449518 0.0031389118 0.0028600595 0.0026059797
## [46] 0.0023744717 0.0021635301 0.0019713281 0.0017962007 0.0016366312
## [51] 0.0014912374 0.0013587600 0.0012380515 0.0011280664 0.0010278521
## [56] 0.0009365405
##
## $cvm
##  [1] 0.3785767 0.3785767 0.3785767 0.3785767 0.3785767 0.3785767 0.3785767
##  [8] 0.3785767 0.3785767 0.3785767 0.3785767 0.3785767 0.3785767 0.3793103
## [15] 0.3793103 0.3785767 0.3778430 0.3712399 0.3697726 0.3646368 0.3521643
## [22] 0.3484960 0.3433602 0.3418929 0.3433602 0.3448276 0.3477623 0.3484960
## [29] 0.3477623 0.3484960 0.3470286 0.3455613 0.3455613 0.3448276 0.3440939
## [36] 0.3462949 0.3426266 0.3455613 0.3455613 0.3484960 0.3477623 0.3470286
## [43] 0.3492296 0.3499633 0.3492296 0.3484960 0.3499633 0.3514307 0.3499633
## [50] 0.3514307 0.3536317 0.3543654 0.3543654 0.3550990 0.3550990 0.3543654
##
```

```
## $cvsd
##  [1] 0.01627135 0.01627135 0.01627135 0.01627135 0.01627135 0.01627135
##  [7] 0.01627135 0.01627135 0.01627135 0.01627135 0.01627135 0.01627135
## [13] 0.01627135 0.01592332 0.01592332 0.01627135 0.01537158 0.01660694
## [19] 0.01574647 0.01513969 0.01403698 0.01552451 0.01502331 0.01529337
## [25] 0.01569822 0.01459452 0.01478450 0.01369662 0.01351553 0.01307159
## [31] 0.01366576 0.01376158 0.01327381 0.01288177 0.01353289 0.01278871
## [37] 0.01261821 0.01284481 0.01284481 0.01244382 0.01152011 0.01117255
## [43] 0.01166922 0.01192102 0.01166922 0.01110538 0.01051583 0.01083090
## [49] 0.01092537 0.01149277 0.01218475 0.01198794 0.01147704 0.01131527
## [55] 0.01131527 0.01158027
##
## $cvup
##  [1] 0.3948480 0.3948480 0.3948480 0.3948480 0.3948480 0.3948480 0.3948480
##  [8] 0.3948480 0.3948480 0.3948480 0.3948480 0.3948480 0.3948480 0.3952337
## [15] 0.3952337 0.3948480 0.3932146 0.3878469 0.3855190 0.3797765 0.3662013
## [22] 0.3640205 0.3583835 0.3571863 0.3590585 0.3594221 0.3625468 0.3621926
## [29] 0.3612778 0.3615676 0.3606944 0.3593228 0.3588351 0.3577094 0.3576268
## [36] 0.3590836 0.3552448 0.3584061 0.3584061 0.3609398 0.3592824 0.3582012
## [43] 0.3608989 0.3618843 0.3608989 0.3596013 0.3604791 0.3622616 0.3608887
## [50] 0.3629234 0.3658164 0.3663533 0.3658424 0.3664143 0.3664143 0.3659456
##
## $cvlo
##  [1] 0.3623053 0.3623053 0.3623053 0.3623053 0.3623053 0.3623053 0.3623053
##  [8] 0.3623053 0.3623053 0.3623053 0.3623053 0.3623053 0.3623053 0.3633870
## [15] 0.3633870 0.3623053 0.3624714 0.3546330 0.3540261 0.3494971 0.3381274
## [22] 0.3329715 0.3283369 0.3265995 0.3276620 0.3302331 0.3329778 0.3347993
## [29] 0.3342468 0.3354244 0.3333629 0.3317997 0.3322874 0.3319458 0.3305610
## [36] 0.3335062 0.3300083 0.3327165 0.3327165 0.3360521 0.3362422 0.3358561
## [43] 0.3375604 0.3380423 0.3375604 0.3373906 0.3394475 0.3405998 0.3390379
## [50] 0.3399379 0.3414469 0.3423774 0.3428883 0.3437838 0.3437838 0.3427851
##
## $nzero
##  s0  s1  s2  s3  s4  s5  s6  s7  s8  s9 s10 s11 s12 s13 s14 s15 s16 s17
##   0   1   1   1   1   1   1   1   1   1   1   1   1   2   3   4   4   5
## s18 s19 s20 s21 s22 s23 s24 s25 s26 s27 s28 s29 s30 s31 s32 s33 s34 s35
##   5   5   5   6   6   6   6   6   6   6   7   7   7   7   7   7   7   7
## s36 s37 s38 s39 s40 s41 s42 s43 s44 s45 s46 s47 s48 s49 s50 s51 s52 s53
##   7   7   7   7   8   8   8   9   9   9   9   9   9   9   9   9   9   9
## s54 s55
##  10  11
##
## $name
##                      class
## "Misclassification Error"
##
## $glmnet.fit
##
## Call:  glmnet(x = predictors, y = train1$HardDrugs, family = "binomial")
##
##       Df       %Dev    Lambda
##  [1,]  0 -1.724e-14 0.1562000
##  [2,]  1  1.345e-02 0.1423000
##  [3,]  1  2.497e-02 0.1297000
##  [4,]  1  3.486e-02 0.1182000
##  [5,]  1  4.340e-02 0.1077000
##  [6,]  1  5.078e-02 0.0981100
##  [7,]  1  5.719e-02 0.0894000
##  [8,]  1  6.275e-02 0.0814600
##  [9,]  1  6.759e-02 0.0742200
## [10,]  1  7.180e-02 0.0676300
## [11,]  1  7.548e-02 0.0616200
## [12,]  1  7.868e-02 0.0561400
## [13,]  1  8.148e-02 0.0511600
## [14,]  2  8.466e-02 0.0466100
## [15,]  3  8.947e-02 0.0424700
## [16,]  4  9.511e-02 0.0387000
## [17,]  4  1.000e-01 0.0352600
## [18,]  5  1.043e-01 0.0321300
## [19,]  5  1.080e-01 0.0292700
## [20,]  5  1.112e-01 0.0266700
## [21,]  5  1.140e-01 0.0243000
## [22,]  6  1.169e-01 0.0221400
## [23,]  6  1.194e-01 0.0201800
## [24,]  6  1.216e-01 0.0183800
## [25,]  6  1.235e-01 0.0167500
## [26,]  6  1.250e-01 0.0152600
## [27,]  6  1.264e-01 0.0139100
## [28,]  6  1.275e-01 0.0126700
## [29,]  7  1.285e-01 0.0115500
## [30,]  7  1.294e-01 0.0105200
## [31,]  7  1.301e-01 0.0095860
## [32,]  7  1.307e-01 0.0087340
## [33,]  7  1.313e-01 0.0079580
## [34,]  7  1.317e-01 0.0072510
## [35,]  7  1.321e-01 0.0066070
## [36,]  7  1.324e-01 0.0060200
## [37,]  7  1.327e-01 0.0054850
## [38,]  7  1.329e-01 0.0049980
## [39,]  7  1.331e-01 0.0045540
## [40,]  7  1.332e-01 0.0041490
## [41,]  8  1.334e-01 0.0037810
## [42,]  8  1.335e-01 0.0034450
## [43,]  8  1.336e-01 0.0031390
## [44,]  9  1.337e-01 0.0028600
## [45,]  9  1.338e-01 0.0026060
## [46,]  9  1.338e-01 0.0023740
## [47,]  9  1.339e-01 0.0021640
## [48,]  9  1.339e-01 0.0019710
## [49,]  9  1.340e-01 0.0017960
## [50,]  9  1.340e-01 0.0016370
## [51,]  9  1.340e-01 0.0014910
## [52,]  9  1.340e-01 0.0013590
## [53,]  9  1.340e-01 0.0012380
## [54,]  9  1.341e-01 0.0011280
## [55,] 10  1.341e-01 0.0010280
## [56,] 11  1.341e-01 0.0009365
## [57,] 11  1.341e-01 0.0008533
```

```
##
## $lambda.min
## [1] 0.01838467
##
## $lambda.1se
## [1] 0.02430345
##
## attr(,"class")
## [1] "cv.glmnet"
```

```
plot(fit)
```



In the output, `cvm` is the mean cross-validated error – a vector of length `length(lambda)`; `lambda.min` is the value of lambda that gives minimum cvm. `lambda.1se` is the largest value of lambda such that error is within 1 standard error of the minimum.

Question:

1. What is the 'optimal' lambda based on the cross-validation?

```
fit$lambda.min
```

```
## [1] 0.01838467
```

Let us use the 'optimal' value of lambda to regulate the logistic regression model.

```
regulated = glmnet(predictors, train1$HardDrugs, family = "binomial", lambda = fit$lambda.min)

regulated$beta
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                            s0
## (Intercept)       .
## Gendermale        0.33669618
## HealthGenVgood    .
## HealthGenGood     .
## HealthGenFair     .
## HealthGenPoor     .
## DepressedSeveral  0.31319072
## DepressedMost     0.63349407
## SleepHrsNight     .
## AlcoholDay        0.01263815
## SmokeNowYes      -0.07626068
## MarijuanaYes      2.21050628
```

Now, fit a logistic regression model without regulation.

```r
mod_lr2 <- glm(HardDrugs~Gender+Depressed+AlcoholDay+SmokeNow+Marijuana, data=train1,family=binomial)
summary(mod_lr2)
```

```
##
## Call:
## glm(formula = HardDrugs ~ Gender + Depressed + AlcoholDay + SmokeNow +
##     Marijuana, family = binomial, data = train1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5931  -1.0258  -0.3005   1.1677   2.6010
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -3.71459    0.34967 -10.623  < 2e-16 ***
## Gendermale        0.56162    0.12806   4.385 1.16e-05 ***
## DepressedSeveral  0.60392    0.14975   4.033 5.51e-05 ***
## DepressedMost     1.16331    0.24009   4.845 1.26e-06 ***
## AlcoholDay        0.03912    0.01887   2.073  0.03813 *
## SmokeNowYes      -0.35157    0.12631  -2.783  0.00538 **
## MarijuanaYes      3.01945    0.33250   9.081  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1808.3  on 1362  degrees of freedom
## Residual deviance: 1566.8  on 1356  degrees of freedom
## AIC: 1580.8
##
## Number of Fisher Scoring iterations: 5
```

Questions:

2. From the output of the final model, which variable is the most significant?

“Marijuana” is the variable that is most significant based on output from final model.

3. What is the relationship between `Marijuana` and `HardDrugs`? Describe it in terms of direction and strength.

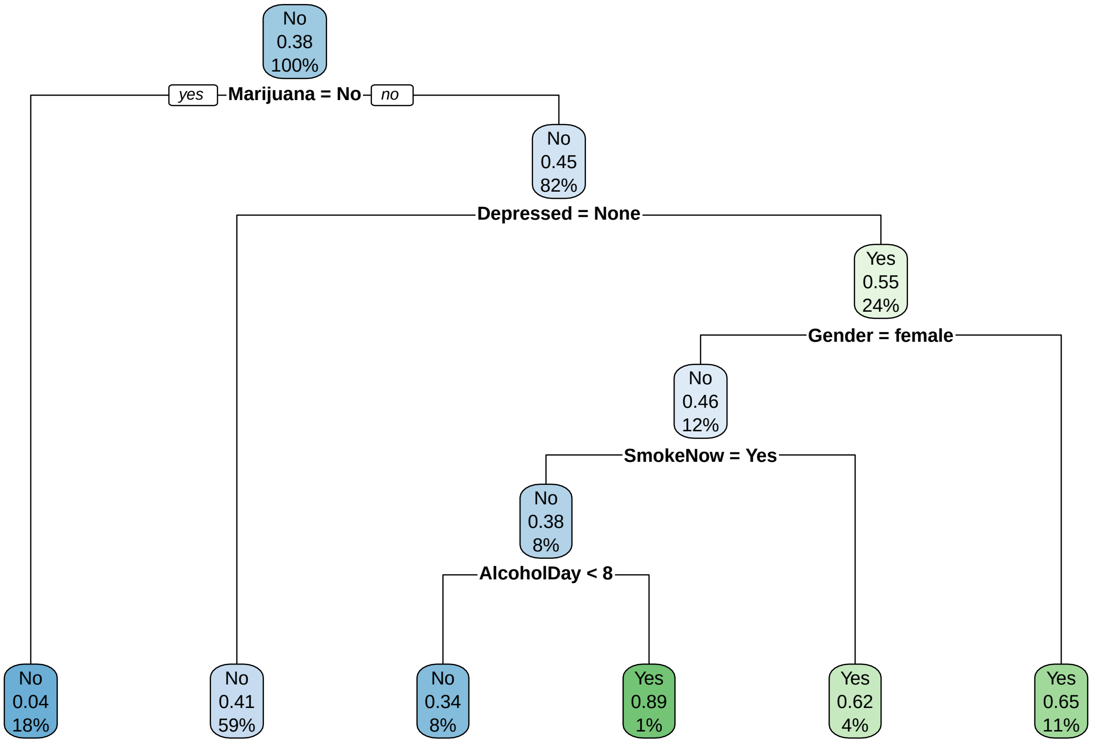
The relationship/correlation between “Marijuana” and “HardDrugs” is a strong and positive one.

4. Males or Female are more likely to use hard drugs?

Males are more likely to use hard drugs.

Model 2: Decision Tree

```
tree = rpart(formula, data = train1)

rpart.plot(tree)
```



From the decision tree, what conclusion can you make? In particular, answer the following questions:

5. Which one seems to be the most important factor among the 7 predictors?

Among the 7 predictors, “Marijuana” usage seems to be the most important factor.

6. Do you think people who have tried Marijuana are more likely to try hard drugs, such as cocaine, heroin, etc.?

Yes, I do personally believe that people who have tried Marijuana are more likely to try hard drugs, such as cocaine, heroine, etc.