Weekly Assignment 05

Xiang Li

2024/3/12

```r
library(glmnet)
```

```
## Loading required package: Matrix

## Loaded glmnet 4.1-8
```

```r
train = readRDS("masq_train.Rda")
test = readRDS("masq_test.Rda")
```

a

```r
cor(train[, 12:100], use = "complete.obs")[1:10, 1:10]
```

```
##           MASQ01       MASQ02      MASQ03    MASQ04    MASQ05    MASQ06
## MASQ01 1.0000000  0.261353159  0.26580128 0.3799619 0.3334452 0.4776699
## MASQ02 0.2613532  1.000000000  0.60885594 0.4719651 0.1793683 0.4386177
## MASQ03 0.2658013  0.608855943  1.00000000 0.5002626 0.2046107 0.3881681
## MASQ04 0.3799619  0.471965131  0.50026256 1.0000000 0.2092146 0.5627482
## MASQ05 0.3334452  0.179368348  0.20461071 0.2092146 1.0000000 0.2786777
## MASQ06 0.4776699  0.438617714  0.38816815 0.5627482 0.2786777 1.0000000
## MASQ07 0.3162404 -0.003336177 -0.03074847 0.1139440 0.1427528 0.1127332
## MASQ08 0.4787031  0.402560583  0.38046996 0.5191725 0.2464122 0.6417347
## MASQ09 0.1795725  0.294294617  0.32506414 0.2929352 0.1664870 0.2591377
## MASQ11 0.5246302  0.200081494  0.21396668 0.2795097 0.2325892 0.3399698
##              MASQ07    MASQ08     MASQ09    MASQ11
## MASQ01  0.316240354 0.4787031 0.17957253 0.5246302
## MASQ02 -0.003336177 0.4025606 0.29429462 0.2000815
## MASQ03 -0.030748474 0.3804700 0.32506414 0.2139667
## MASQ04  0.113943990 0.5191725 0.29293518 0.2795097
## MASQ05  0.142752789 0.2464122 0.16648703 0.2325892
## MASQ06  0.112733188 0.6417347 0.25913770 0.3399698
## MASQ07  1.000000000 0.1167942 0.04063244 0.3327615
## MASQ08  0.116794186 1.0000000 0.24678531 0.3394676
## MASQ09  0.040632436 0.2467853 1.00000000 0.1491871
## MASQ11  0.332761545 0.3394676 0.14918709 1.0000000
```

We can see that there are some highly correlated relationships among predictors, so we need use variable selection. The performance of these three models could be elastic net regression > lasso regression > ridge regression.

b

Choice elastic net (with $\alpha = 0.5$), lasso and relaxed lasso.

c

```r
train_X = model.matrix(D_DEPDYS ~ . - 1, data = train)
train_y = train$D_DEPDYS
test_X = model.matrix(D_DEPDYS ~ . - 1, data = test)
test_y = test$D_DEPDYS
```

```r
set.seed(519)
lasso_cv_model = cv.glmnet(x = train_X, y = train_y, family = "binomial", alpha = 1)
lasso_cv_model
```

```
##
## Call:  cv.glmnet(x = train_X, y = train_y, family = "binomial", alpha = 1)
##
## Measure: Binomial Deviance
##
##      Lambda Index Measure      SE Nonzero
## min 0.01140    34   1.028 0.02357      38
## 1se 0.03482    22   1.050 0.01725      18
```

```r
set.seed(519)
elanet_cv_model = cv.glmnet(x = train_X, y = train_y, family = "binomial", alpha = 0.5)
elanet_cv_model
```

```
##
## Call:  cv.glmnet(x = train_X, y = train_y, family = "binomial", alpha = 0.5)
##
## Measure: Binomial Deviance
##
##      Lambda Index Measure      SE Nonzero
## min 0.02078    35   1.027 0.02341      43
## 1se 0.06346    23   1.048 0.01756      24
```

```r
set.seed(519)
rlasso_cv_model = cv.glmnet(x = train_X, y = train_y, family = "binomial", alpha = 1,
    relax = TRUE)
rlasso_cv_model
```

```
##
## Call:  cv.glmnet(x = train_X, y = train_y, relax = TRUE, family = "binomial",      alpha = 1)
##
## Measure: Binomial Deviance
##
##     Gamma Index  Lambda Index Measure      SE Nonzero
## min  1.00     5 0.01140    34   1.028 0.02357      38
## 1se  0.25     2 0.06679    15   1.050 0.02062      12
```

Based on the binomial deviance, the best model is elastic net ($\alpha = 0.5$).

d

```
elanet_y = predict(elanet_cv_model, newx = test_X, s = "lambda.min", type = "response")
elanet_y = (elanet_y > 0.5) * 1
elanet_MCR = mean(elanet_y != test_y)
elanet_MCR
```

```
## [1] 0.2291435
```

The MCR of elastic net model on test set is 0.2291.

e

```
coef_mat = coef(elanet_cv_model, s = "lambda.min")
coef_result = coef_mat[coef_mat[, 1] != 0, 1]
coef_result
```

```
##  (Intercept)      GENDERm      GENDERv     Leeftijd      DEMOG26      DEMOG32
## -5.459473825 -0.034368564  0.028060897  0.007358254 -0.353841681  0.071558718
##      DEMOG34     DEMOG3NA      DEMOG53      DEMOG55      DEMOG62      DEMOG72
## -0.019574423 -0.209533000 -0.181904110  0.130540152  0.228083585  0.022965398
##       MASQ01       MASQ02       MASQ03       MASQ04       MASQ05       MASQ13
##  0.170351729 -0.071314371 -0.037831392  0.004491825  0.011420746  0.058926040
##       MASQ14       MASQ16       MASQ18       MASQ21       MASQ22       MASQ24
##  0.032352228  0.306523934  0.041945595  0.036592390  0.101300266  0.032626889
##       MASQ29       MASQ30       MASQ31       MASQ33       MASQ37       MASQ38
##  0.002027648  0.112532564  0.045704211  0.008858013  0.103909663  0.030143268
##       MASQ41       MASQ43       MASQ44       MASQ50       MASQ54       MASQ59
##  0.134000912  0.060017828  0.009045107  0.005053763  0.018096950 -0.052972139
##       MASQ60       MASQ62       MASQ70       MASQ76       MASQ78       MASQ83
##  0.050538144  0.074037505  0.018541828  0.060985800  0.052375491  0.004783868
##       MASQ89       MASQ90
##  0.145667618  0.076646625
```

```
select_predictors = c("GENDER", "Leeftijd", "DEMOG2", "DEMOG3", "DEMOG5", "DEMOG6",
    "DEMOG7", "MASQ01", "MASQ02", "MASQ03", "MASQ04", "MASQ05", "MASQ13", "MASQ14",
    "MASQ16", "MASQ18", "MASQ21", "MASQ22", "MASQ24", "MASQ29", "MASQ30", "MASQ31",
    "MASQ33", "MASQ37", "MASQ38", "MASQ41", "MASQ43", "MASQ44", "MASQ50", "MASQ54",
    "MASQ59", "MASQ60", "MASQ62", "MASQ70", "MASQ76", "MASQ78", "MASQ83", "MASQ89",
    "MASQ90")
select_predictors_id = which(colnames(train) %in% select_predictors) - 1
AD_id = c(1, 14, 18, 21, 23, 26, 27, 30, 33, 35, 36, 39, 40, 44, 49, 53, 58, 66,
    72, 78, 86, 89)
AA_id = c(3, 19, 25, 45, 48, 52, 55, 57, 61, 67, 69, 73, 75, 79, 85, 87, 88)
GDD_id = c(6, 8, 10, 13, 16, 22, 24, 42, 47, 56, 64, 74)
```

```r
GDA_id = c(2, 9, 12, 15, 20, 59, 63, 65, 77, 81, 82)
GDM_id = c(4, 5, 17, 29, 31, 34, 37, 50, 51, 70, 76, 80, 83, 84, 90)
print(sum(select_predictors_id %in% AD_id))
```

```
## [1] 9
```

```r
print(sum(select_predictors_id %in% AA_id))
```

```
## [1] 6
```

```r
print(sum(select_predictors_id %in% GDD_id))
```

```
## [1] 5
```

```r
print(sum(select_predictors_id %in% GDA_id))
```

```
## [1] 6
```

```r
print(sum(select_predictors_id %in% GDM_id))
```

```
## [1] 4
```

The Anhedonic Depression subscale.