DA_Lab6_HW

Na SeungChan

2023-10-23

Exercises

1. Select 10 variables to predict cesd and explain why you choose those variables.

```
df_exs <- HELPrct %>%
  select(id, cesd, age, female, avg_drinks, dayslink, drugrisk, mcs, pcs, pss_fr, racegrp, treat)
```

2. Handle various values and NAs before we start analysis.

```
levels(df_exs$racegrp)
```

```
## [1] "black"    "hispanic" "other"    "white"
```

```
levels(df_exs$treat)
```

```
## [1] "no"  "yes"
```

우선, `treat`는 2개의 levels를 가지고 있으므로 'yes'이면 1로, 'no'이면 0으로 coding한다. 다음으로 4개의 level을 가지는 treat `racegrp`에는 one-hot coding을 적용한다.(사실 transmute로 적용한 ifelse문도 one-hot coding이긴 하다…)

```
exs_one_hot <- model.matrix(~0+df_exs$racegrp)

df_temp <- df_exs %>%
  cbind(exs_one_hot) %>%
  mutate(istreat = ifelse(treat == 'yes', 1, 0)) %>%
  rename('isBlack' = 'df_exs$racegrpblack', 'isHisph' = 'df_exs$racegrphispanic', 'isOther' = 'df_exs$ra
  select(-racegrp, -treat)
```

`female`은 여성일 때 1, 남성일 때 0을 가지는 binary variable이므로 별도의 처리를 할 필요가 없다.(data 차원에서 이미 `sex`에 대해 별도의 처리를 해 놓은 변수임) 나머지 변수는 모두 별도의 처리를 하지 않아도 회귀분석 가능하다.

```
sum(is.na(df_temp$dayslink))
```

```
## [1] 22
```

```r
sum(is.na(df_temp$drugrisk))
```

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

```r
temp_mean <- mean(df_temp$dayslink, na.rm = TRUE)
temp_mean2 <- mean(df_temp$drugrisk, na.rm = TRUE)
df_exs2 <- df_temp %>%
  mutate(dayslink = ifelse(is.na(dayslink), temp_mean, dayslink), drugrisk = ifelse(is.na(drugrisk), ter

sum(is.na(df_exs2$dayslink))
```

```
## [1] 0
```

```r
sum(is.na(df_exs2$drugrisk))
```

```
## [1] 0
```

한편… NA를 처리하는 방법으로 크게 두 가지가 있는데, 사실 딱히 평균으로 NA를 대체해야 할 근거는 없다. 단, id가 존재하는 data인데 싸그리 drop하면 id에서 1,3,4, … 하는 식으로 빈 id가 생기는 것이 불편해서 해당 분석에서는 na를 모두 평균으로 대치하는 방식을 적용하였다.

3. Perform multiple linear regression and explain the result.

```r
model_q3 <- lm(cesd ~ ., data = df_exs2 %>% select(-id, -isOther))
summary(model_q3)
```

```
##
## Call:
## lm(formula = cesd ~ ., data = df_exs2 %>% select(-id, -isOther))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.1877  -5.9438  -0.0423   5.4447  25.8138
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 62.992780   3.761877  16.745  < 2e-16 ***
## age         -0.038293   0.056346  -0.680  0.49711
## female       2.855186   0.991337   2.880  0.00417 **
## avg_drinks   0.055110   0.021752   2.534  0.01164 *
## dayslink    -0.001960   0.003083  -0.636  0.52534
## drugrisk     0.048991   0.101397   0.483  0.62922
## mcs         -0.604496   0.033526 -18.030  < 2e-16 ***
## pcs         -0.213836   0.040772  -5.245 2.44e-07 ***
## pss_fr      -0.243891   0.103985  -2.345  0.01945 *
## isBlack      0.318672   1.826655   0.174  0.86159
## isHisph      1.807018   2.100821   0.860  0.39018
## isWhite      1.954062   1.843182   1.060  0.28965
```

```
## istreat     -0.076481   0.903918  -0.085  0.93261
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.628 on 440 degrees of freedom
## Multiple R-squared:  0.5373, Adjusted R-squared:  0.5247
## F-statistic: 42.58 on 12 and 440 DF,  p-value: < 2.2e-16
```

id를 뺀 이유? 아무 의미가 없는 변수라서… 또한, isOther을 넣으면 특정 열이 다른 열의 선형결합으로 표현되어 X matrix가 full column rank가 아니게 되고, 이에 따라 역행렬이 존재하지 않게 되는 문제가 되어 Hat matrix를 찾을 수 없게 된다.

아무튼, `mcs`와 `pcs`는 모두 매우 유의하며, `being female` 역시 꽤나 유의하다.

4. Perform stepwise selection methods to find the best model and justify your result.

```
m_both <- ols_step_both_aic(model_q3)
summary(m_both)
```

```
##            Length Class  Mode
## predictors 6      -none- character
## method     6      -none- character
## steps      1      -none- numeric
## arsq       6      -none- numeric
## aic        6      -none- numeric
## ess        6      -none- numeric
## rss        6      -none- numeric
## rsq        6      -none- numeric
```

```
model_q4 <- lm(cesd ~ mcs + pcs + avg_drinks + female + pss_fr + isBlack, data = df_exs2)
```

앞의 model에서 p-value 낮은 유의성 높은 변수들은 모두(*이 들어가 있던) model에 들어갔다. 한편, `isBlack`도 들어갔는데… 다른 변수에 비해 p-value가 그나마 높은 것은 사실이다.

한편, 해당 stepwise selection에서는 addition method만이 사용되었다. 즉 이 결과는 forward selection과 같을 것이다…

```
m_for <- ols_step_forward_aic(model_q3)
m_for$predictors
```

```
## [1] "mcs"        "pcs"        "avg_drinks" "female"     "pss_fr"
## [6] "isBlack"
```

실제로 같다. 반면 backward selection을 해 보면

```
m_back <- ols_step_backward_aic(model_q3)
m_back$predictors
```

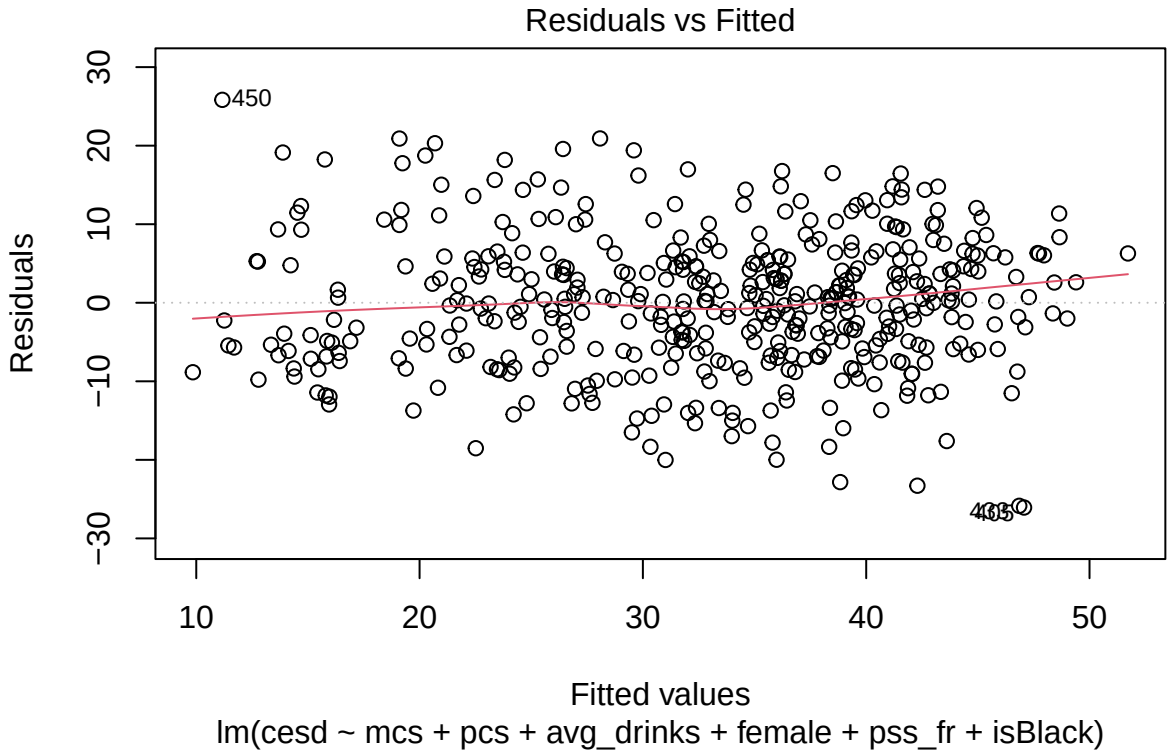
```
## [1] "istreat"  "isBlack"  "drugrisk" "dayslink" "age"      "isHisph"
```

대단히 다르다. 이는…. 잘 모르겠다! 이런 결과가 나올 수도 있다는 건 알겠는데.

5. Use the model in 4, perform regression diagnostic procedures (linearity, homoscedasticity, normality, independence). Explain the result for each assumption. If the assumption does not hold, give an alternative method to satisfy the result.

Linearity, Homoscedastity

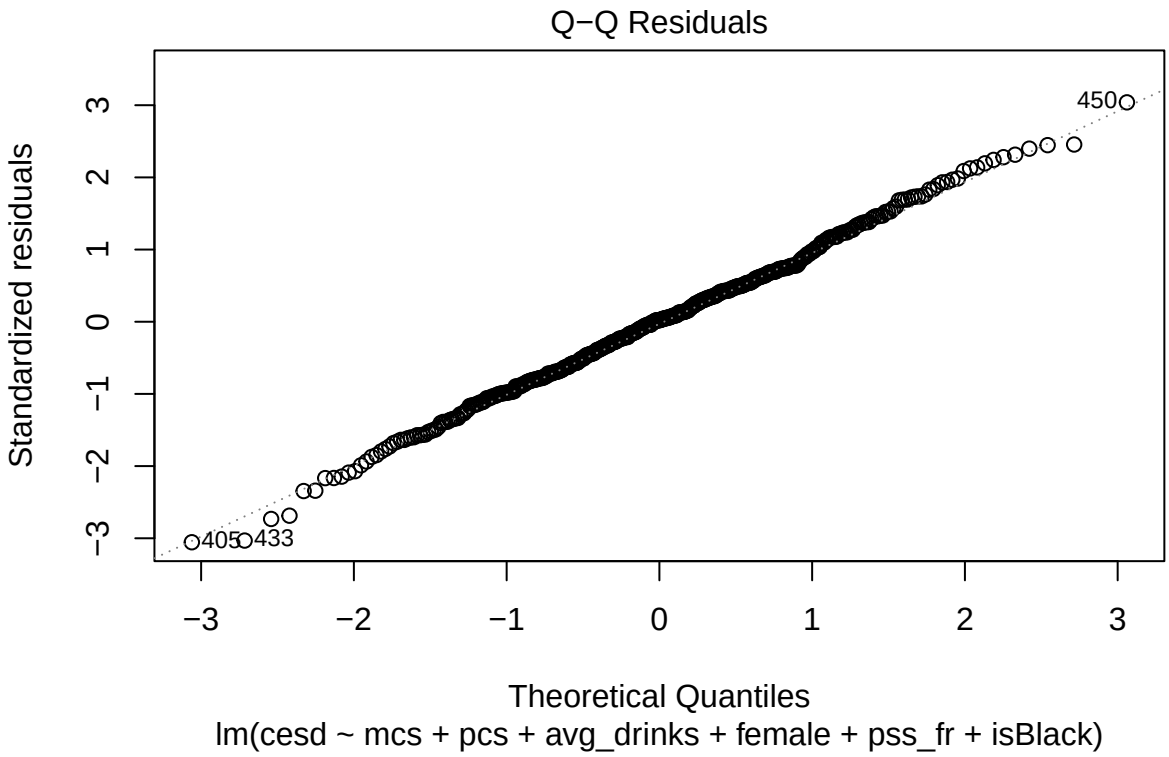
```
plot(model_q4, which = 1)
```



선형성 문제는 없는 것 같고… 등분산성 문제도 크지 않은 것 같다.

Normality

```
plot(model_q4, which = 2)
```



정규성 문제는 적어 보인다.

Normality

```
dwtest(cesd ~ mcs + pcs + avg_drinks + female + pss_fr + isBlack, data = df_exs2, alternative = "two.si
```

```
##
##  Durbin-Watson test
##
## data:  cesd ~ mcs + pcs + avg_drinks + female + pss_fr + isBlack
## DW = 2.0999, p-value = 0.2846
## alternative hypothesis: true autocorrelation is not 0
```

DW의 값이 2 주변이 나오므로 자기상관관계가 없다는 귀무가설을 기각하기 어렵다.

6. Select only 2 variables that were the most significant in 3. Compare two models, (1) multiple linear regression not considering the interaction term and (2) multiple linear regression considering the interaction term. Discuss the result.

```
model_q6a <- lm(cesd ~ mcs + pcs, data = df_exs2)
model_q6b <- lm(cesd ~ mcs*pcs, data = df_exs2)
summary(model_q6a)
```

```
##
## Call:
## lm(formula = cesd ~ mcs + pcs, data = df_exs2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -24.2855  -6.2421   0.1138   5.4071  26.0520
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 65.42136    2.05035  31.907  < 2e-16 ***
## mcs         -0.64098    0.03227 -19.865  < 2e-16 ***
## pcs         -0.25536    0.03841  -6.647 8.65e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.754 on 450 degrees of freedom
## Multiple R-squared:  0.5129, Adjusted R-squared:  0.5107
## F-statistic: 236.9 on 2 and 450 DF,  p-value: < 2.2e-16
```

```
summary(model_q6b)
```

```
##
## Call:
## lm(formula = cesd ~ mcs * pcs, data = df_exs2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -24.7672  -6.2379   0.1837   5.4547  25.3745
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 70.844891   5.379485  13.169  < 2e-16 ***
## mcs         -0.815887   0.163612  -4.987 8.79e-07 ***
## pcs         -0.363825   0.106624  -3.412 0.000703 ***
## mcs:pcs      0.003466   0.003179   1.090 0.276094
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.752 on 449 degrees of freedom
## Multiple R-squared:  0.5141, Adjusted R-squared:  0.5109
## F-statistic: 158.4 on 3 and 449 DF,  p-value: < 2.2e-16
```

여전히 두 term 모두 상호작용을 고려하는 model과 고려하지 않는 model 모두에서 유의하다. 그러나 p-value의 값은 상호작용을 고려하는 model에서 크게 감소한다.

7. Select only 1 variable that were the most significant in 3. To fit the multiple regression $y=\beta0+\beta1x+\varepsilon$ using the L1 loss function, choose a proper grid range of (β0,β1) and find the values of β0,β1 minimizing the total L1 loss.

```r
model_q7 <- lm(cesd ~ mcs, data = df_exs2)
summary(model_q7)

glmnet(df_exs2$mcs, df_exs2$cesd)
```

multiple regression y=β0+β1x+ε은 오타겠죠…? 근데 glmnet 함수는 muliple regression에서 Lasso regression을 하기 위한 거고… lm 함수에서 loss function을 조정하는 방법이 따로 있는지