다변량분석 과제 4

2015170378

정은영

[Q1] Logistic Regression 모형 구축을 위해 필요하지 않은 변수는 어떤 것들이 있는가? 왜 그렇게 생각하는가?

```
df <- read.csv('Admission_Predict.csv')
str(df)
head(df,3)  #데이터 확인
length(unique(df$Serial.No.)) == nrow(df) #Serial number 가 중복되는 것이 없는지 확인
sum(is.na(df)) #missing value 있는지 확인
#중복이나 결측값 없음.

#[Q1]

df[1] <- NULL #Serial number 는 admission의 변화에 영향 주지 않기 때문에 제거.
```

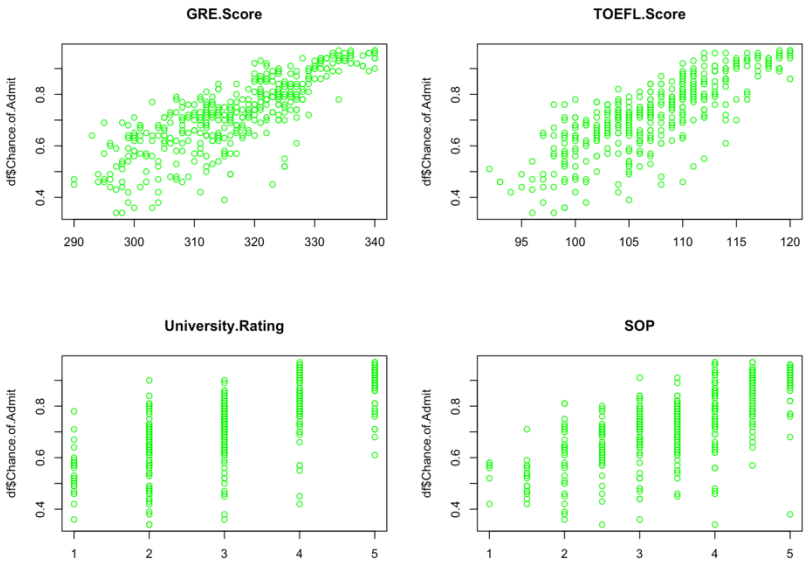
-> 변수 중 Serial number 가 admission 의 변화에 영향을 주지 않기 때문에 제거한다.

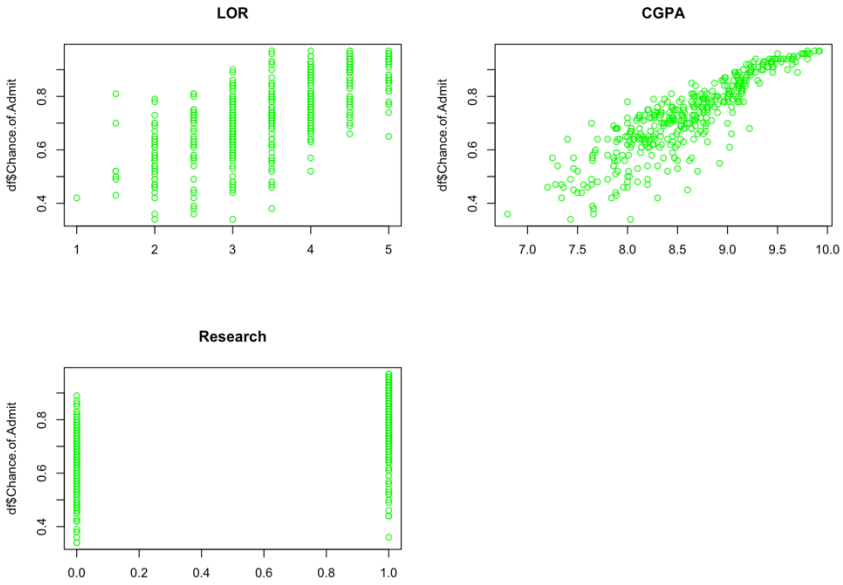
다음 물음에 대해서는 [Q1]에서 선택한 변수들은 제외하고 답변하시오.

[Q2] 개별 입력 변수들에 대하여 각각 다음과 같은 단변량 통계량을 계산하고 Box plot 을 도시하시오: Mean, Standard deviation, Skewness, Kurtosis. 전체 변수 중에서 정규분포를 따른다고 할 수 있는 변수 들은 몇 개인가? 정규분포를 따른다고 가정한 근거는 무엇인가?

```
#[Q2]

par(mfrow = c(2,2))
for(i in 1:(length(colnames(df))-1)){
  plot(df[,i], df$Chance.of.Admit, main = names(df[i]), ylab = names(df$Chance.of.Admit), xlab
= "", col = 'green')
}
```

-> y 축을 Chance of admit 로 하고 나머지 변수들을 input 으로 하여 그래프를 그려본다.





데이터프레임을 참고하고 그래프를 그려본 결과 변수 University.Rating, SOP, LOR, Research 가 discrete 변수라고 생각된다.

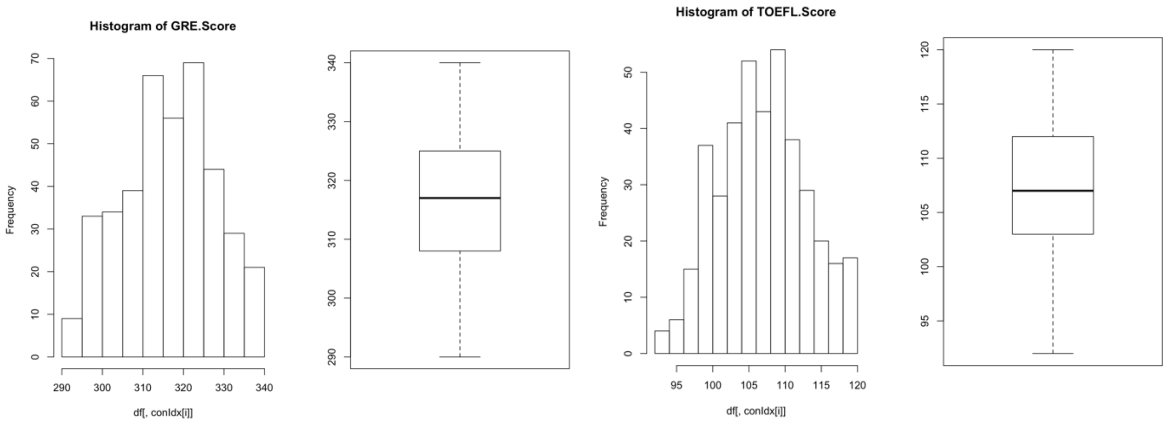
#따라서 이 변수들의 column index 를 구해보면 3,4,5,7 이다.

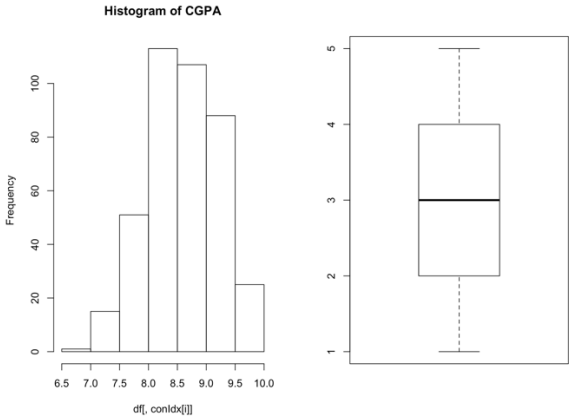
#또한 그래프를 그려본 결과 input 변수들이 종속변수와 양의 상관관계를 가질 것이라고 예상된다.

```
cateIdx <- c(3,4,5,7) #categorical index
conIdx = c(1:(length(colnames(df))-1))[!(c(1:(length(colnames(df))-1)) %in% cateIdx
)] #non categorical index

# mean, standard deviation, skewness, kurtosis 포함 matrix 정의
nMat <- matrix(c(1:length(conIdx)*4),nrow=length(conIdx),ncol=4)
colnames(nMat)<- c("mean", "std","skewness","kurtosis")
rownames(nMat)<- colnames(df[conIdx])

#Continuous value 에 대해 histogram그리고, matrix 채우기.
par(mfrow=c(1,2))
for(i in 1:length(conIdx)){
  hist(df[,conIdx[i]], main = paste("Histogram of" , colnames(df[conIdx[i]])))
  boxplot(df[,i])
  nMat[i,1] <- mean(unlist(df[,conIdx[i]]))
  nMat[i,2] <- sqrt(var(df[,conIdx[i]]))
  nMat[i,3] <- skewness(df[,conIdx[i]])
  nMat[i,4] <- kurtosis(df[,conIdx[i]])
}
```

-> 범주형과 아닌것으로 나누어 연속형 값에 대해 histogram 을 그리고 box plot 을 먼저 그려본다.

-> 또한 mean, sqrt, skewness, kurtosis 값을 matrix 로 구해본다.





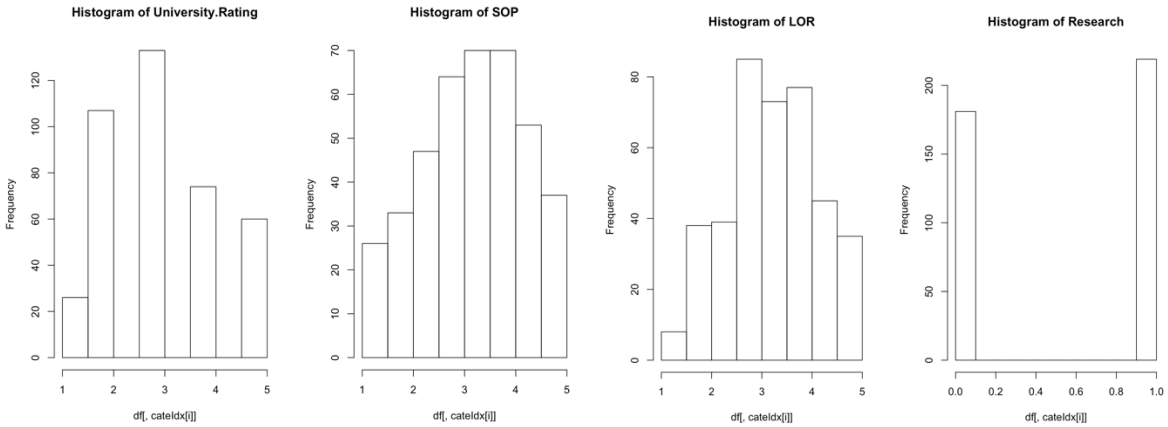
non-categorical value 들의 histogram 을 그려본 결과 대부분 정규분포와 같이 퍼져있고 outlier 도 거의 관측되지 않음을 확인할 수 있다.

	mean	std	skewness	kurtosis
GRE.Score	316.807500	11.4736461	-0.06265736	2.293273
TOEFL.Score	107.410000	6.0695138	0.05700113	2.413468
CGPA	8.598925	0.5963171	-0.06574282	2.532273

matrix 의 결과를 봐도 skewness 가 거의 0 에 가깝고 kurtosis 가 2 와 3 사이임을 확인하여 정규분포를 따른다고 볼 수 있다. 표준정규분포의 skewness 는 0, kurtosis 는 3 임을 고려했을 때 연속형 변수 모두 정규분포를 거의 따른다고 할 수 있다.

```
#Categorical Value에 대해 Summary 확인
for(i in 1:length(cateIdx)){
  print(colnames(df[cateIdx[i]]))
  print(summary(factor(df[,cateIdx[i]])))
  hist(df[,cateIdx[i]], main = paste("Histogram of" , colnames(df[cateIdx[i]])))
}
```

-> discrete, 범주형 변수들에 대해서는 summary 를 확인해보고 histogram 을 통해 대략적인 데이터 분포를 확인한다.




```
[1] "University.Rating"
  1   2   3   4   5
 26 107 133  74  60
[1] "SOP"
  1 1.5   2 2.5   3 3.5   4 4.5   5
  6  20  33  47  64  70  70  53  37
[1] "LOR"
  1 1.5   2 2.5   3 3.5   4 4.5   5
  1   7  38  39  85  73  77  45  35
[1] "Research"
  0   1
181 219
```

히스토그램과 summary 를 보고 대략적인 파악을 하면 University.Rating, SOP, LOR 은 중간으로 갈수록 값이 커지며, 데이터 범주도 고르게 퍼져있어 정규분포의 모양과 유사함을 확인할 수 있다. 또한 Research 는 0,1 의 binary 값으로 이루어져 있음을 알 수 있다.

[Q3] [Q2]의 Box plot 을 근거로 각 변수들에 대한 이상치(너무 값이 크거나 작은 객체) 조건을 정의하고, 해당하는 객체들을 데이터셋에서 제거해 보시오.

#categorical 변수들에 대해서는 위 문제의 summary 를 확인했을 때 이상치가 없다고 볼 수 있다.
#Continuous 변수들은 Box plot 근거로 upper_range, lower_range 정하여 outliers 탐색.

```
64 for(i in conIdx){
65   print(colnames(df[i]))
66   summary(df[,i])
67   q1 <- quantile(df[,i], c(0.25))
68   q3 <- quantile(df[,i], c(0.75))
69   IQR <- q3 - q1
70   upper_range <- q3 + 1.5*IQR
71   lower_range <- q1 - 1.5*IQR
72   print(nrow(df[df[,i] > upper_range,])+nrow(df[df[,i] < lower_range,])) #outliers
     개수
73   df[,i]<- ifelse(df[,i] > upper_range, NA, df[,i]) #outlier 제거(윗부분)
74   df[,i]<- ifelse(df[,i] > upper_range, NA, df[,i]) #outlier 제거(윗부분)
75   df<- na.omit(df) #outlier 해당 데이터 데이터셋에서 제거
76 }
```

-> 각 범주형 변수 아닌 변수들에 대해 upper range, lower range 를 벗어나는 데이터들을 outlier 로 간주하여 제거한다. outlier 개수를 프린트한다.

```
[1] "GRE.Score"
[1] 0
[1] "TOEFL.Score"
[1] 0
[1] "CGPA"
[1] 1
```

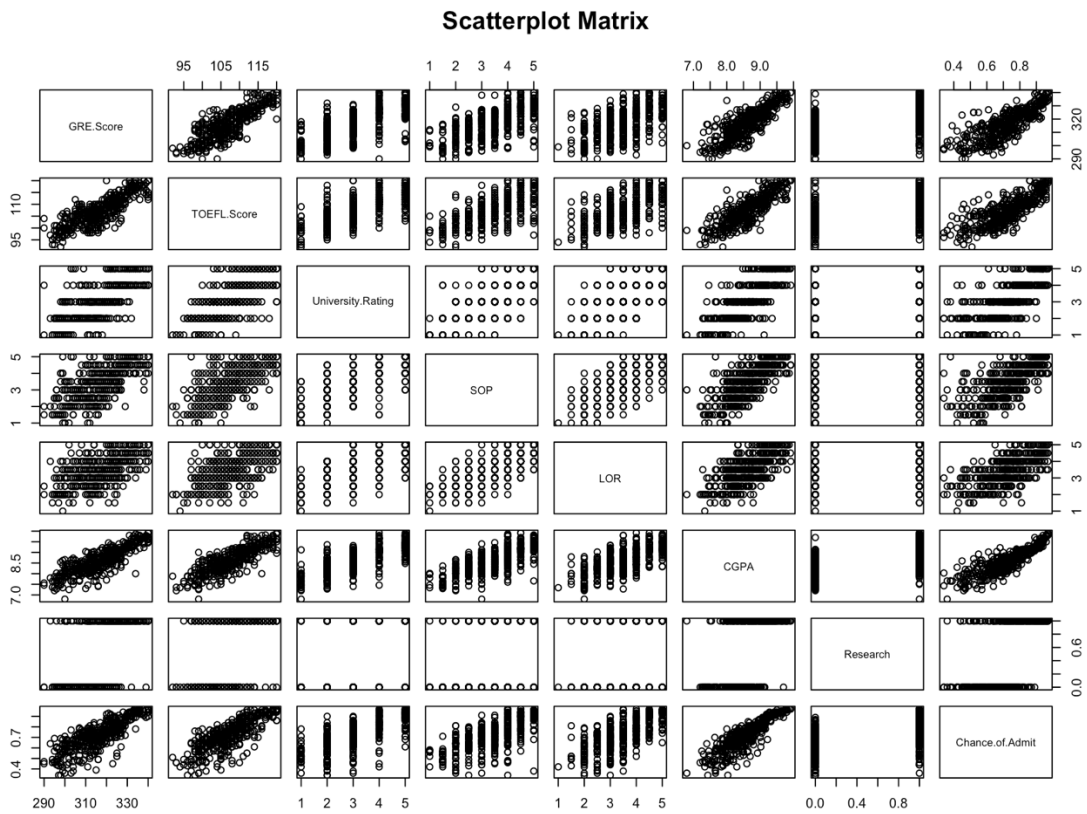
#GRE.Score, TOEFL.Score 은 outliers 없고, CGPA 는 1 개의 outlier 가 있어 제거한다.

다음 각 물음에 대해서는 [Q3]에서 제거된 객체들을 제외시킨 데이터프레임을 사용하여 답하시오.

[Q4] 가능한 모든 두 쌍의 입력 변수 조합에 대한 산점도(scatter plot) 및 Correlation plot (hint: "corrplot" 패키지의 corrplot( ) 함수 사용) 상관관계를 계산해 보시오. 어떤 두 조합의 변수들이 서로 강 한 상관관계가 있다고 할 수 있는가?

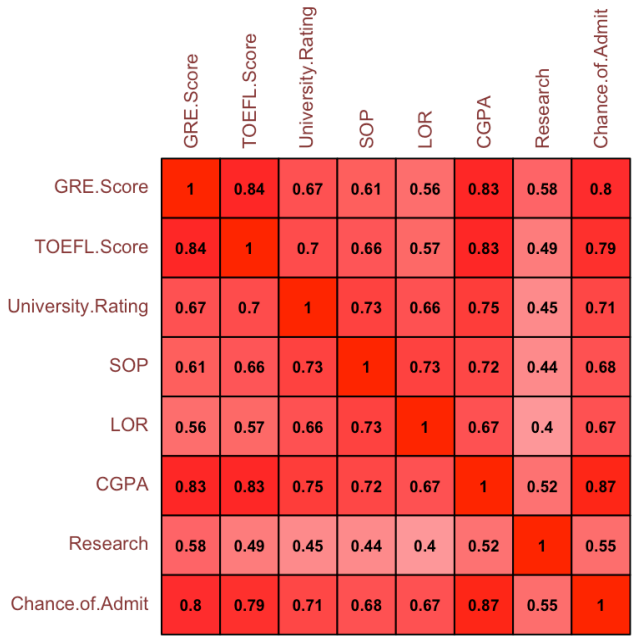
```
79  # Basic Scatterplot Matrix
80  pairs(~.,data=df,
81        main="Scatterplot Matrix")
```

-> scatter plot 그리기



```
79  # correlation plot
80  corr <- cor(df)
81  corrplot(corr, method = "color", outline = T, cl.pos = 'n', rect.col = "black",  tl
    .col = "indianred4", addCoef.col = "black", number.digits = 2, number.cex = 0.60, tl
    .cex = 0.7, cl.cex = 1, col = colorRampPalette(c("green4","white","red"))(100))
```

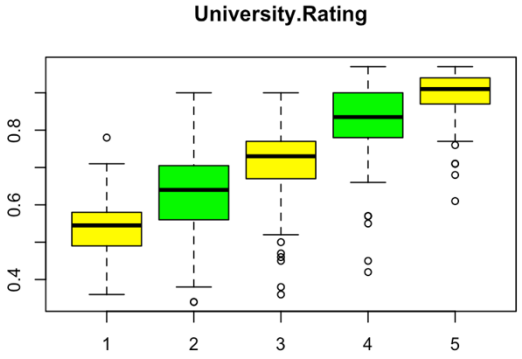
-> correlation plot 그리기

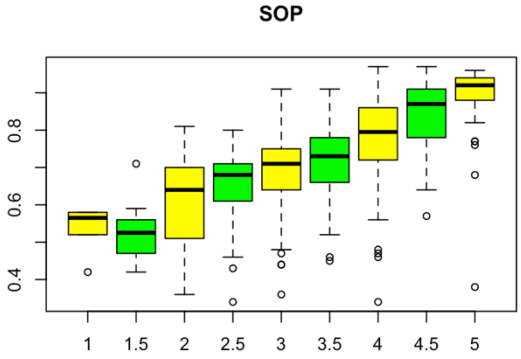


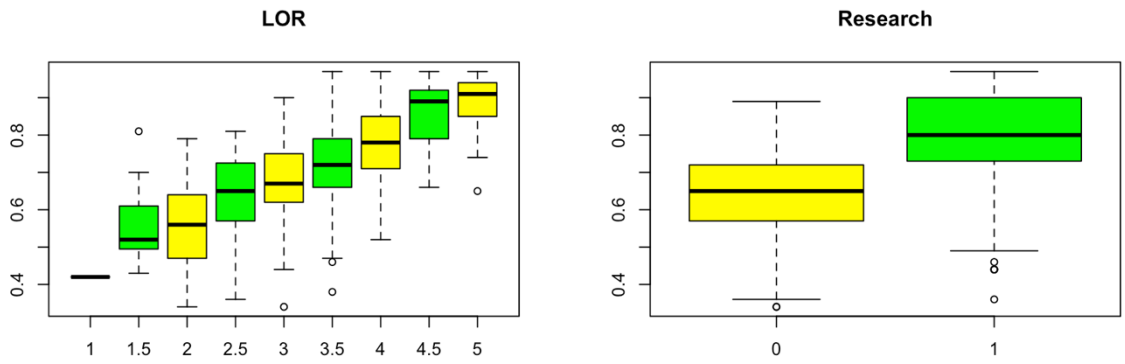
#범주형 변수 아닌 GRE.Score, TOEFL.Score, CGPA, Chance of Admit 가 correlation 이 0.8 정도로 강한 상관관계를 보인다.

```
89  #Discrete value 에 대해 y축을 chance of admit로 boxplot 그리기.
90  par(mfrow=c(2,2))
91  for(i in cateIdx){
92    boxplot(df[,length(colnames(df))]~df[,i], xlab='', main=names(df[i]), col=c
    ("yellow","green"))
93  }
```

-> discrete value 에 대해 y 축을 입학기회로 boxplot 을 그려보면 다음과 같다.







#box plot 을 그렸을 때 데이터의 트렌드를 보면 대학수준(University Rating), 학업계획서(SOP), 추천서(LOR) 가 수준이 높아질수록 입학기회가 높아짐(높은 상관관계)을 확인할 수 있다. 또한 연구를 하면 입학기회가 높아진다는 것을 확인할 수 있다.

[Q5] 종속변수인 Change of Admit 은 원래 데이터에서는 0 부터 1 사이의 확률 값으로 표현되어 있다. 이를 0.8 을 기준으로 하여 0.8 을 초과하는 경우 1 (positive class), 0.8 이하인 경우 0 (negative class)의 값을 갖 는 binary target variable 로 변환하시오. 이후 전체 데이터셋을 70%의 학습 데이터와 30%의 테스트 데이 터로 무작위(random)로 분할한 후 모든 변수를 사용하여 Logistic Regression 모델을 학습해 보시오. 유의수준(Significance level) 0.1 에서 Change of Admit 에 유의미하게 영향을 주는 변수들은 어떤 것들이 있는가?

```
100  table(df$Chance.of.Admit > 0.8) #False=283, True=117, 즉 입학기회 0.8보다 적은 것이
     높은 것보다 많다.
101  df$Chance.of.Admit = as.factor(ifelse(df$Chance.of.Admit > 0.8,1,0)) #0,1을 갖는
     값으로 target variable 변환.
```

-> 0.8 을 기준으로 했을 때 큰 것을 true, 아닌 것을 false 로 하여 해당 데이터 개수를 구해보고, 0.8 보다 크면 1, 아니면 0 의 값을 갖도록 target variable 을 변환한다.

```
FALSE  TRUE
  283   117
```

입학기회가 0.8 보다 큰 것은 117 개, 아닌것은 283 개임을 확인할 수 있다.

	SOP	LOR	CGPA	Research	Chance.of.Admit
4	4.5	4.5	9.65	1	1
4	4.0	4.5	8.87	1	0
3	3.0	3.5	8.00	1	0
3	3.5	2.5	8.67	1	0
2	2.0	3.0	8.21	0	0

-> 입학기회(합격확률)가 1,0 으로(합격, 불합격) 바뀐 것을 확인할 수 있다.

```
# Conduct the normalization
input_idx <- c(1:7)
target_idx <- 8 #Chance of admit

df_input <- df[,input_idx]
df_input <- scale(df_input, center = TRUE, scale = TRUE) #round error 발생할 수 있어 SCALE 맞춤.
df_target <- df[,target_idx]
df_scaled <- data.frame(df_input, df_target)
```

-> 각 변수에 대해 scale 맞춰준다.

```
# Split the data into the training/validation sets
set.seed(12345)
trn_idx <- sample(1:nrow(df_scaled), round(0.7*nrow(df_scaled)))
df_trn <- df_scaled[trn_idx,]
df_tst <- df_scaled[-trn_idx,]

# Train the Logistic Regression Model with all variables
full_lr <- glm(df_target ~ ., family=binomial, df_trn) #GLM_generalize linear model
summary(full_lr) #유의수준. 이상
#CGPA 의 p-value가 0.01보다 작고, TOEFL점수의 p-value가 0.1보다 작다.
```

-> 70% 를 training data로, 나머지를 test data로 분리하고 Logistic regression model 을 학습한다.

```
Call:
glm(formula = df_target ~ ., family = binomial, data = df_trn)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2704  -0.1706  -0.0336   0.0433   3.3749

Coefficients:
                  Estimate Std. Error z value       Pr(>|z|)
(Intercept)        -3.4512     0.5540  -6.230 0.000000000467 ***
GRE.Score           0.1824     0.7247   0.252         0.8013
TOEFL.Score         1.1917     0.6364   1.872         0.0612 .
University.Rating   0.6392     0.5037   1.269         0.2044
SOP                -0.3686     0.6265  -0.588         0.5563
LOR                 0.2836     0.4355   0.651         0.5148
CGPA                3.8149     0.9758   3.910 0.000092453302 ***
Research            0.4537     0.3436   1.320         0.1867
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 329.375  on 279  degrees of freedom
Residual deviance:  92.353  on 272  degrees of freedom
AIC: 108.35

Number of Fisher Scoring iterations: 8
```

summary 를 확인해 본 결과 CGPA 의 p-value 가 0.01 보다 작고, TOEFL 점수의 p-value 가 0.1 보다 작아 유의수준 0.1 보다 작은 변수인 CGPA(학점평균)과 TOEFL.Score(토플점수)가 대학원 입학유무(Chance of admit)에 유의미한 영향을 준다는 것을 확인할 수 있다.

[Q6] Test 데이터셋에 대하여 예측을 수행하고 Confusion Matrix 를 생성한 뒤, True Positive Rate, True Negative Rate, False Positive Rate, False Negative Rate, Simple Accuracy, Balanced Correction Rate, F1-Measure 를 각각 구하고 그 의미를 해석하시오.

```
#prediction
lr_response <- predict(full_lr, type = "response", newdata = df_tst) #예측, 숫자_sample번호, 값들의 CUTOFF를 결정해서 이걸 나눔.
lr_target <- df_tst$df_target
lr_predicted <- rep(0, length(lr_target))
lr_predicted[which(lr_response >= 0.5)] <- 1 #로지스틱 cutoff를 0.5로 결정하겠다.
cm_full <- table(lr_target, lr_predicted) #실제정답, 예측 CONFUSION MATRIX 생성.
cm_full
```

-> test data 에 대해 예측 수행 후 logistic cutoff 를 가장 많이 사용되는 0.5 로 설정하여 confusion matrix 를 생성한다.

```
         lr_predicted
lr_target  0  1
        0 76  4
        1  3 37
```

#logistic cutoff 0.5 로 했을 시 타겟을 제대로 예측하는 개수가 0-0 은 76 개, 1-1 은 37 개임, 또한 제대로 못한 경우 0-1 은 4 개, 1-0 은 3 개임. 즉, 대학원 합격을 제대로 예측하는 개수 37 개, 못하는 개수 3 개, 대학원 불합격 제대로 예측 개수 76 개, 아닌것 4 개이다.

```
#perference evaluation funtion
perf_eval2 <- function(cm){

  # True positive rate: TPR (Recall)
  TPR <- cm[2,2]/sum(cm[2,]) #cm=>confusion matrix,R-script는 오름차순이어서 TPR이 2행.
  # True negative rate: TNR
  TNR <- cm[1,1]/sum(cm[1,])
  # False Positive Rate: FPR
  FPR <- cm[1,2]/sum(cm[1,])
  # False Negative Rate: FNR
  FNR <- cm[2,1]/sum(cm[2,])

  # Precision
  PRE <- cm[2,2]/sum(cm[,2])
  # Simple Accuracy
  ACC <- (cm[1,1]+cm[2,2])/sum(cm)
  # Balanced Correction Rate
  BCR <- sqrt(TPR*TNR)
  # F1-Measure
  F1 <- 2*TPR*PRE/(TPR+PRE)

  return(c(TPR, TNR, FPR, FNR, PRE, ACC, BCR, F1))
}

# Initialize the performance matrix
perf_mat <- matrix(0, 1, 8)
colnames(perf_mat) <- c("TPR (Recall)", "TNR","FPR","FNR","Precision", "ACC", "BCR", "F1")
rownames(perf_mat) <- "Logstic Regression"
```

-> perference evalution factor 들을 계산하는 함수를 생성하고, 결과값을 넣을 matrix 생성한다.

```
162  #결과
163  perf_mat[1,] <- perf_eval2(cm_full)
164  perf_mat
```

-> confusion matrix를 함수에 넣어 결과를 계산한다.

```
          lr_predicted
lr_target  0  1
        0 76  4
        1  3 37
```

-> 이 행렬을 기반으로 계산된 결과는 다음과 같다.

```
                   TPR (Recall)  TNR  FPR   FNR Precision       ACC       BCR        F1
Logstic Regression        0.925 0.95 0.05 0.075  0.902439 0.9416667 0.9374167 0.9135802
```

#TPR: 실제로 '예'일 때, 얼마나 자주 '예'라고 예측하는가? -> 92.5%

#TNR: 실제로 '아니오'일 때, 얼마나 자주 '아니오'를 예측하는가? -> 95%

#FPR: 실제로 '아니오'일 때, 얼마나 자주 '예'라고 예측하는가? -> 5%

#FNR: 실제로 '예'일 때, 얼마나 자주 '아니오'를 예측하는가? -> 7.5%

#Precision: '예'라고 예측했을 때, 얼마나 자주 정확한가? -> 90%

#Accuracy: 전반적으로 얼마나 자주 분류가 정확한가? -> 94.2%

#BCR(balanced correction rate): TPR, TNR 의 곱을 제곱근 취한 것으로 하나가 0 이면 0 이 되어 하나만 커도 과대평가되는 것을 막아준다 -> 93.7%

#FI-Measure: 정밀도(Precision: 찾아야할 것이라고 예측한 것 중 진짜 그런 것)과 재현율(TPR: 찾아야 할 것 중 실제로 찾은 비율)의 평균으로 성능을 평가할 때 자주쓰임. -> 91.4%

TPR, TNR, Precision, Accuracy, BCR, FI-Measure 이 모두 90% 이상으로 높은 것으로 보아 분류가 비교적 잘 될 것이라는 것을 알 수 있다.

TPR 보다 Accuracy 가 더 큰데 이는 대학원 불합격한 수가 더 많다는 것을 알 수 있다.

[Q7] Test 데이터셋에 대한 AUROC 를 산출하는 함수를 직접 작성하고, random seed 를 변경해가면서 학습-테스트를 5 회 반복하여 산출된 AUROC 값의 변화를 확인해보시오.

```
#AUROC 산출 함수_교수님 프린트를 기반으로 x,y축에 평행한 선들로 구성된 ROC 커브로 간주.
AUROC <- function(seed_num){

  # Split the data into the training/validation sets
  set.seed(seed_num)
  trn_idx <- sample(1:nrow(df_scaled), round(0.7*nrow(df_scaled)))
  df_trn <- df_scaled[trn_idx,]
  df_tst <- df_scaled[-trn_idx,]

  # 학습
  full_lr <- glm(df_target ~ ., family=binomial, df_trn) #GLM_generalize linear model

  # 예측
  lr_response <- predict(full_lr, type = "response", newdata = df_tst)
  lr_target <- df_tst$df_target
  lr_predicted <- rep(0, length(lr_target))

  #TPR, FPR 담을 벡터 생성 및 필요한 변수 초기화
  TPR_vec <- c(length(lr_response))
  FPR_vec <- c(length(lr_response))
  dummy <- matrix(0,nrow = 2,ncol = 1)

  dFPR <- 0
  dTPR <- 0
  AUROC <- 0

  #TPR, FPR 기반으로 AUROC 계산
  for(i in 1:length(lr_response)){
    lr_predicted[which(lr_response >= lr_response[i])] <- 1 #cutoff를 하나씩 옮기기
    cm_1 <- cbind(table(lr_target, lr_predicted),dummy)
    #confusion matrix의 2열이 나오지 않을 때를 대비하여(2열이 0,0 일때 표시안됨) dummy로 0,0을 추가.
    TPR_vec[i] <- cm_1[2,2]/sum(cm_1[2,]) #TPR
    FPR_vec[i] <- cm_1[1,2]/sum(cm_1[1,]) #FPR
    if(i>1){
      dFPR <- FPR_vec[i]-FPR_vec[(i-1)] #FPR이 처음에 0, FPR의 차이 계산
    }
    AUROC <- AUROC + TPR_vec[i] * dFPR #막대들 더하기
  }

  return(AUROC)
}
```

-> AUROC 값을 계산하는 함수를 작성한다. input 은 seed number 이다.

```
#AUROC(seed number) 로 계산한 AOROC 값.
AUROC(12345)
AUROC(11111)
AUROC(1000)
AUROC(123)
AUROC(54321)
```

```
> AUROC(12345)
[1] 0.95
> AUROC(11111)
[1] 0.9883721
> AUROC(1000)
[1] 0.9782609
> AUROC(123)
[1] 0.9354839
> AUROC(54321)
[1] 0.958551
```

Seed number 를 바꿔가면서 AUROC 를 계산했을 때, 5 번 다 0.95 를 넘어 1 에 가깝기 때문에 classification performance 가 좋다고 볼 수 있다.

[Q8] 이 외 웹이나 기타 자료들을 통해 재미있는 데이터셋(fun dataset)을 찾아 나름대로의 로지스틱 회귀 분석 모형 구축 및 결과 해석을 수행하시오.

'미쳐라' 쇼핑몰에서 파이썬 코드를 통해 직접 긁어온 바지사이즈를 포함한 정보 데이터로 바지사이즈 S,M,L 를 종속변수로 한 multiple logistic regression 모델을 구축해보았다.

코드는 교수님 실습코드 중 다중로지스틱회귀 부분을 참고하였다.

	ID	name	price	size	waist	thigh	length	image	url
1	0	메디와이드PT	19900	S	32.5	28.5	88.0	http://cdn-michyeora.bizhost.kr/files/goods/38327/...	http://www.michyeora.com/shop/view.php?index_no...
2	1	메디와이드PT	19900	M	35.0	29.5	89.0	http://cdn-michyeora.bizhost.kr/files/goods/38327/...	http://www.michyeora.com/shop/view.php?index_no...
3	2	메디와이드PT	19900	L	37.5	30.5	90.0	http://cdn-michyeora.bizhost.kr/files/goods/38327/...	http://www.michyeora.com/shop/view.php?index_no...
4	4	(세.편.바)컷팅면5부PT	14300	S	32.5	28.5	39.5	http://cdn-michyeora.bizhost.kr/files/goods/38181/...	http://www.michyeora.com/shop/view.php?index_no...
5	5	(세.편.바)컷팅면5부PT	14300	M	35.0	29.5	40.5	http://cdn-michyeora.bizhost.kr/files/goods/38181/...	http://www.michyeora.com/shop/view.php?index_no...
6	6	(세.편.바)컷팅면5부PT	14300	L	37.5	30.5	41.5	http://cdn-michyeora.bizhost.kr/files/goods/38181/...	http://www.michyeora.com/shop/view.php?index_no...
7	7	보드바이청반PT	22900	S	32.0	28.0	32.5	http://cdn-michyeora.bizhost.kr/files/goods/38542/...	http://www.michyeora.com/shop/view.php?index_no...
8	8	보드바이청반PT	22900	M	34.0	30.0	34.0	http://cdn-michyeora.bizhost.kr/files/goods/38542/...	http://www.michyeora.com/shop/view.php?index_no...
9	9	보드바이청반PT	22900	L	36.0	32.0	35.5	http://cdn-michyeora.bizhost.kr/files/goods/38542/...	http://www.michyeora.com/shop/view.php?index_no...

데이터는 다음과 같이 ID, name, price, size, waist(허리둘레), thigh(허벅지둘레), image(이미지 url), url(바지 페이지 url)로 구성되어있다.

```
install.packages("nnet")
library(nnet)

#데이터 불러오기
pants <- read.csv('Michyeora_Pants_Info_UTF_8.csv')
str(pants)
head(pants,3)  #데이터 확인

pants[,c(1,2,8,9)] <- NULL #ID, name, Image, URL 은 size에 영향을 주지 않기 때문에 제거
```

-> ID, name, Image, URL 은 size 에 영향을 주지 않으므로 제거한다.

```
# 평가지표 구하는 함수
perf_eval3 <- function(cm){

  # Simple accuracy
  ACC <- sum(diag(cm))/sum(cm)

  # ACC for each class, 대각선 따로 계산한 것.
  A1 <- cm[1,1]/sum(cm[1,])
  A2 <- cm[2,2]/sum(cm[2,])
  A3 <- cm[3,3]/sum(cm[3,])
  BCR <- (A1*A2*A3)^(1/3) #세제곱 루트

  return(c(ACC, BCR))
}
```

-> 평가지표 구하는 함수를 미리 작성한다. 각각 class 의 accuracy 를 구하여 전체 accuracy 를 구하고, BCR 을 구하는 함수이다.

```
# Define the baseline class
pants$size <- as.factor(pants$size) #CLASS가 종속변수, 문자로 되어있는데 범주로 바꾸어줘야함.
pants$size <- relevel(pants$size, ref = "L") #기준 범주를 선택해주는 작업, L로 선택.
```

-> size 가 S, M, L 의 문자로 되어 있는데 이를 범주로 바꾸어준다.

```
252  # 학습-평가 데이터 분류
253  trn_idx <- sample(1:nrow(pants), round(0.7*nrow(pants)))
254  pants_trn <- pants[trn_idx,]
255  pants_tst <- pants[-trn_idx,]
256
257  # Train multinomial logistic regression
258  ml_logit <- multinom(size ~ ., data = pants_trn)
```

-> 트레이닝-테스트 데이터로 분류하고, 학습시킨다. multinom 을 사용한다.

```
260  # Check the coefficients
261  summary(ml_logit) #S,M 각각은 p(Y=S)/p(Y=L), p(Y=M)/P(Y=L)
262  t(summary(ml_logit)$coefficients)
```

-> logistic model 의 summary 를 확인한다. coefficients 를 따로 확인해본다.

```
Call:
multinom(formula = size ~ ., data = pants_trn)

Coefficients:
  (Intercept)         price     waist      thigh      length
M    55.93211 0.00004226303 -1.155415 -0.4669938 -0.02635820
S   102.37139 0.00007537687 -2.245455 -0.7828960 -0.04863928

Std. Errors:
   (Intercept)         price      waist      thigh     length
M 0.0006336068 0.00007369282 0.02391267 0.02100596 0.01146571
S 0.0008549704 0.00009170950 0.02890094 0.02858462 0.01490973

Residual Deviance: 125.5286
AIC: 145.5286
```

```
> t(summary(ml_logit)$coefficients)
                          M              S
(Intercept) 55.93210899302 102.37139260523
price        0.00004226303   0.00007537687
waist       -1.15541515583  -2.24545492987
thigh       -0.46699380542  -0.78289599244
length      -0.02635819861  -0.04863927936
```

#S,M 각각이 p(Y=S)/p(Y=L), p(Y=M)/P(Y=L) 로 계산된다.

#즉, coefficient 를 보면 waist 의 경우 M, S 둘다 (-) 값을 가지는데 이는 허리둘레가 증가하면 log(p(y=M)/p(y=L)), log(p(y=M)/p(y=L)) 가 감소되는 것이다.

p-value 가 계산되지 않기 때문에 직접 계산한다.

```
264 # Conduct 2-tailed z-test to compute the p-values
265 z_stats <- summary(ml_logit)$coefficients/summary(ml_logit)$standard.errors #P-VALUE를
    따로 계산하기 위해
266 t(z_stats)
267
268 p_value <- (1-pnorm(abs(z_stats), 0, 1))*2
269 options(scipen=10)
270 t(p_value) #P-VALUE
271
272 cbind(t(summary(ml_logit)$coefficients), t(p_value)) #회귀계수랑 P-VALUE를 S,M에 대해
    보여줌(L을 기준으로 확률)
```

-> p-value 를 계산하기 위해 z 통계량을 이용한다. z 통계량을 확인한다. z 통계량으로 p-value 를 계산하고 확인한다. 한번에 보기 위해 coefficient 와 p-value 를 동시에 확인한다.

```
> t(z_stats)
                        M             S
(Intercept) 88275.7356998 119736.7675439
price           0.5735026      0.8219091
waist         -48.3181105    -77.6948669
thigh         -22.2314859    -27.3887139
length         -2.2988720     -3.2622499
> p_value <- (1-pnorm(abs(z_stats), 0, 1))*2
> options(scipen=10)
> t(p_value) #P-VALUE
                    M           S
(Intercept) 0.00000000 0.000000000
price       0.56630443 0.411128650
waist       0.00000000 0.000000000
thigh       0.00000000 0.000000000
length      0.02151221 0.001105317
> cbind(t(summary(ml_logit)$coefficients), t(p_value)) #회귀계수랑 P-VALUE를 S,M에 대해 보여줌(L을 기준으로
확률)
                          M               S          M           S
(Intercept) 55.93210899302 102.37139260523 0.00000000 0.000000000
price        0.00004226303   0.00007537687 0.56630443 0.411128650
waist       -1.15541515583  -2.24545492987 0.00000000 0.000000000
thigh       -0.46699380542  -0.78289599244 0.00000000 0.000000000
length      -0.02635819861  -0.04863927936 0.02151221 0.001105317
```

유의수준을 0.05 로 잡았을 때 Price 는 p-value 가 M, S 모두 0.05 를 넘어 가격이 바지 사이즈를 구분 짓는 데 (M 과 L, S 와 L) 에 영향을 주지 않음을 알 수 있다.

waist 와 thigh 의 경우 p-value 가 모두 0 에 가까워 유의수준 0.05 보다 작으며, 이는 허리둘레와 허벅지 둘레가 M 과 L, S 와 L 를 구분하는 데에 영향을 줌을 알 수 있다.

length 의 경우 p-value 가 모두 0.05 보다 작아 M 과 L, S 와 L 을 구분하는데 유의미하다는 것을 알 수 있다. 다만 허리둘레와 허벅지둘레보다 p-value 가 큰 이유는 데이터에 반바지와 긴바지가 섞여있기 때문이라는 것을 알 수 있다. 만약 반바지만 있거나, 긴바지만 있다면 0 에 가까울 것이라 추정된다.

waist, thigh, length 의 coefficient 가 모두 (-)인것으로 보아 허리둘레가 증가할수록, 허벅지둘레가 증가할수록, 바지길이가 증가할수록 log(p(y=M)/p(y=L)), log(p(y=M)/p(y=L)) 가 감소함을 알 수 있고 이는 로그 안의 p((y=M)/p(y=L)), p((y=S)/p(y=L)) 이 감소함을 알 수 있다. 이는

독립변수들이 줄어들수록 상대적으로 종속변수들의 확률의 비율 중 분자보다 분모가 커진다는 것을 의미하는데 즉, 허리둘레, 허벅지 둘레, 바지길이가 증가할수록 S,M 에 비해 L 의 비율이 높아진다는 것을 의미한다. 다시 말하면 허리둘레, 허벅지 둘레, 바지길이가 증가할수록 사이즈가 커질 것이라는 것을 추정할 수 있다.

```
274  # Predict the class probability
275  ml_logit_haty <- predict(ml_logit, type="probs", newdata = pants_tst)
276  ml_logit_haty[1:10,]
277
278  # Predict the class label
279  ml_logit_prey <- predict(ml_logit, newdata = pants_tst)
280
281  cfmatrix <- table(pants_tst$size, ml_logit_prey)
282  cfmatrix
283  perf_eval3(cfmatrix) #단순 정확도, 균형정확도
```

-> class 비율을 계산하고, test 데이터에 대해 예측하고 confusion matrix를 만들어 performance를 evaluation 한다.

```
> cfmatrix
   ml_logit_prey
     L  M  S
  L 16  3  0
  M  3  3  5
  S  0  2 13
> perf_eval3(cfmatrix) #단순 정확도, 균형정확도
[1] 0.7111111 0.5838694
```

#test data 에 대한 confusion matrix 는 다음과 같다.

#제대로 분류할 경우의 수가 L,S 의 경우에는 많이 나왔으나(즉 제대로 분류한 비율이 높음) M 의 경우에 제대로 구분하지 못했음을 알 수 있다.

#단순정확도는 71.1%정도로, 균형정확도는 58,4% 로 나왔다. 균형정확도는 하나가 제대로 분류를 못할 경우 확 낮아지게 되기 때문에 M 의 영향인 것으로 보인다.

#결과적으로 input 변수(허리둘레, 허벅지 둘레, 바지길이)로 L, S 를 잘 구분지을 확률이 높다고 할 수 있다.