<K-Chip 데이터 분석>

첨부한 phenotype.txt 파일에는 K-Chip 데이터의 여러 샘플 정보가 들어 있으며, 해당 정보는 아래 테이블과 K-Chip_CodeBook.xlsx 파일을 통해 얻을 수 있다. 이 데이터를 활용하여, 암과 같은 범주형 데이터 한 개와 비만 또는 혈압과 같은 연속형 변수 한 개를 각각 선택하여 두 개의 반응 변수에 대해 관련 있는 변수들을 설명변수로 선택하고, 데이터마이닝 수업 시간에 배운 여러 머신러닝 기법을 활용하여 예측 모델을 구축하여라. 각 모델의 예측 성능은 prediction R^2와 AUC 값을 계산하여 비교한다. 반응 변수와 설명변수를 선택한 이유를 설명하고, 데이터를 training과 test set으로 분할한 후 prediction R^2와 AUC 값을 통해 예측력이 가장 높은 모델을 선택해 보시오. 마지막으로, 리포트에 각 조원이 담당한 역할을 구체적으로 서술하여 분석 과정에 대해 기술하여라.

변수명	뜻	자료형
AGE_B	검진 당시 나이	int
SMOK_B	흡연유무 (1:비흡연자 2:과거흡연자 3:흡연자)	int
SMOKA_MOD_B	현재 흡연의 흡연량:개피/하루단위로 정리	num
ALCO_B	음주여부 1:예 2:아니오	int
ALCO_AMOUT_B	1회당 잔수로 환산	num
EXER_B	평소에 건강을 위해 규칙적으로 운동을 하고 계십니까? 1:예 2:아니오	int
MDM_B		int
MHTN_B		int
MLPD_B		int
PHTN_B		int
PDM_B		int
PLPD_B		int
HT_B	신장 cm	num
WT_B	체중 kg	num
WAIST_B	허리둘레 cm	num
SBP_B	수축기혈압 mmHg	int
DBP_B	이완기혈압 mmHg	int
CHO_B	총콜레스테롤 mg/ dl	int
LDL_B	저밀도콜레스테롤 mg/ dl	num
TG_B	중성지방 mg/ dl	int
HDL_B	고밀도콜레스테롤 mg/ dl	num

FBS_B	공복혈당 mg/ dl	int
GOT_B	AST IU/L	int
GPT_B	ALT IU/L	int
GGT_B	r-GTP IU/L	int
URIC_B	요산 mg/ dl	num
PCAN80	폐암 가족력 여부	int
PCAN81	위암 가족력 여부	int
PCAN82	직장암 가족력 여부	int
PCAN83	대장암 가족력 여부	int
PCAN84	유방암 가족력 여부	int
PCAN86	간암 가족력 여부	int
PCAN89	갑상선암 가족력 여부	int
FCAN80	폐암 가족력 여부	int
FCAN81	위암 가족력 여부	int
FCAN82	직장암 가족력 여부	int
FCAN83	대장암 가족력 여부	int
FCAN84	유방암 가족력 여부	int
FCAN86	간암 가족력 여부	int
FCAN89	갑상선암 가족력 여부	int
FEV1	1초간 뱉을 수 있는 호흡량	num
FVC	최대 숨 들이마신 후 최대한 뱉어보는 호흡량	num
BIL	총빌리루빈 mg/ dl	num
WBC	백혈구 $X10^3$/ ul	num
CREAT	크레아티닌 mg/ dl	num
STOMA	위암	int
COLON	대장암	int
LIVER	간암	int
LUNG	폐암	int
PROST	전립선암	int
THROI	갑상선암	int
BREAC	유방암	int
RECTM	직장암	int
SCOLON		num
SRECTM		num

SPROST		**num**
STHROI		**num**
SBREAC		**num**
SLUNG		**num**
SSTOMA		**num**
SLIVER		**num**
SEX1	**성별 (1:남자, 2:여자)**	**int**
CRC		**int**
SCRC		**num**

```
> colSums(is.na(d1))
       FID         IID       AGE_B      SMOK_B  SMOKA_MOD_B
         0           0           0          10          321
    ALCO_B ALCO_AMOUNT_B      EXER_B       MDM_B      MHTN_B
        35          85          73         954          880
   MLPD_B      PHTN_B       PDM_B      PLPD_B        HT_B
       972         847         933         964           2
      WT_B    WAIST_B       SBP_B      DBP_B       CHO_B
         2          29          12          11           5
   LDL_B        TG_B       HDL_B      FBS_B       GOT_B
         5           5          13           2           4
   GPT_B        GGT_B      URIC_B     PCAN80      PCAN81
         4           8          71         996         985
  PCAN82       PCAN83      PCAN84     PCAN86      PCAN89
      1000         999         992         996         982
  FCAN80       FCAN81      FCAN82     FCAN83      FCAN84
       970         931         999         977         980
  FCAN86       FCAN89        FEV1        FVC         BIL
       965         989         431         431          52
    WBC        CREAT       STOMA      COLON       LIVER
        72          38           0           0           0
   LUNG        PROST       THROI      BREAC       RECTM
         0           0           0           0           0
 SCOLON       SRECTM      SPROST     STHROI      SBREAC
         0           0           0           0           0
  SLUNG       SSTOMA      SLIVER       SEX1         CRC
         0           0           0           0           0
   SCRC
         0
```

결측치 존재 X 변수

FID, IID, AGE_B, SEX

STOMA, COLON, LIVER, LUNG, PROST, THROI, BREAC. RECTM, CRC

SCOLON, SRECTM, SPROST, STHROI, SBREAC, SLUNG, SSTOMA, SLIVER, SCRC

결측치 존재 O 변수

SMOK_B 1,2,3 : 평균값으로 대체.

SMOKA_MOD_B : SMOK_B값이 1이라면 0으로 대체, SMOK_B 값이 2라면 값들의 평균값으로 대체

PASS

ALCO_B, ALCO_AMOUNT_B, EXER_B ,MDM_B, MHTN_B, MLPD_B, PHTN_B, PDM_B, PLPD_B, FEV1, FVC

HT_B, WT_B, WAIST_B, SBP_B, DBP_B, CHO_B, LDL_B, TG_B, HDL_B, FBS_B, GOT_B, GPT_B, GGT_B, URIC_B : 평균값으로 대체

PCAN80 ~ PCAN 89, FCAN80 ~ FCN89 : 0으로 대체 (0: 과거력, 가족력 X, 1: 과거력, 가족력 O)

BIL, WBC, CREAT : 평균값으로 대체

데이터 전처리 코드

```
getwd()
setwd("C:/Users/82106/동계인턴십_Data_1000")

d1 <- read.table("phenotype_1000.txt", header = T)
str(d1)

# 이상치 처리 (NA값을 평균값 or 0으로 대체)
colSums(is.na(d1))
# SMOK_B 이상치 처리
d1$SMOK_B <- ifelse(!is.na(d1$SMOK_B), d1$SMOK_B, round(mean(d1$SMOK_B, na.rm=T))) # 10개의 이상치, 2로 대체

# SMOKA_MOD_B 이상치 처리
length(which(d1$SMOK_B == 1 & is.na(d1$SMOKA_MOD_B)))
length(which(d1$SMOK_B == 2 & is.na(d1$SMOKA_MOD_B)))
length(which(d1$SMOK_B == 3 & is.na(d1$SMOKA_MOD_B)))

d1$SMOKA_MOD_B <- ifelse(d1$SMOK_B == 1 & is.na(d1$SMOKA_MOD_B), 0, d1$SMOKA_MOD_B) # 276개의 이상치, 0로 대체
round(mean(d1$SMOKA_MOD_B[d1$SMOK_B==2])) # 41개의 이상치, 13으로 대체
d1$SMOKA_MOD_B <- ifelse(d1$SMOK_B == 2 & is.na(d1$SMOKA_MOD_B), round(mean(d1$SMOKA_MOD_B[d1$SMOK_B==2], na.rm=T)), d1$SMOKA_MOD_B)
round(mean(d1$SMOKA_MOD_B[d1$SMOK_B==3])) # 4개의 이상치, 16으로 대체
d1$SMOKA_MOD_B <- ifelse(d1$SMOK_B == 3 & is.na(d1$SMOKA_MOD_B), round(mean(d1$SMOKA_MOD_B[d1$SMOK_B==3], na.rm=T)), d1$SMOKA_MOD_B)

# pass
d1$ALCO_B
d1$ALCO_AMOUNT_B
d1$EXER_B
d1$MDM_B
d1$MHTN_B
d1$MLPD_B
d1$PHTN_B
d1$PDM_B
d1$PLPD_B

# HT_B, WT_B, WAIST_B 이상치 처리
d1$HT_B <- ifelse(!is.na(d1$HT_B), d1$HT_B, round(mean(d1$HT_B, na.rm=T))) # 2개의 이상치, 166로 대체
d1$WT_B <- ifelse(!is.na(d1$WT_B), d1$WT_B, round(mean(d1$WT_B, na.rm=T))) # 2개의 이상치, 66로 대체
d1$WAIST_B <- ifelse(!is.na(d1$WAIST_B), d1$WAIST_B, round(mean(d1$WAIST_B, na.rm=T))) # 29개의 이상치, 82로 대체
round(mean(d1$WAIST_B, na.rm=T))

# SBP_B, DBP_B, CHO_B, LDL_B, TG_B, HDL_B, FBS_B, GOT_B, GPT_B, GGT_B, URIC_B 이상치 처리
d1$SBP_B <- ifelse(!is.na(d1$SBP_B), d1$SBP_B, round(mean(d1$SBP_B, na.rm=T))) # 124개의 이상치, 121로 대체
round(mean(d1$SBP_B, na.rm=T))
d1$DBP_B <- ifelse(!is.na(d1$DBP_B), d1$DBP_B, round(mean(d1$DBP_B, na.rm=T))) # 11개의 이상치, 75로 대체
round(mean(d1$DBP_B, na.rm=T))
d1$CHO_B <- ifelse(!is.na(d1$CHO_B), d1$CHO_B, round(mean(d1$CHO_B, na.rm=T))) # 5개의 이상치, 192로 대체
round(mean(d1$CHO_B, na.rm=T))
d1$LDL_B <- ifelse(!is.na(d1$LDL_B), d1$LDL_B, round(mean(d1$LDL_B, na.rm=T))) # 5개의 이상치, 116로 대체
round(mean(d1$LDL_B, na.rm=T))
d1$TG_B <- ifelse(!is.na(d1$TG_B), d1$TG_B, round(mean(d1$TG_B, na.rm=T))) # 5개의 이상치, 137로 대체
round(mean(d1$TG_B, na.rm=T))
d1$HDL_B <- ifelse(!is.na(d1$HDL_B), d1$HDL_B, round(mean(d1$HDL_B, na.rm=T))) # 13개의 이상치,
```

```
51로 대체
round(mean(d1$HDL_B, na.rm=T))
d1$FBS_B <- ifelse(!is.na(d1$FBS_B), d1$FBS_B, round(mean(d1$FBS_B, na.rm=T))) # 2개의 이상치, 93로 대체
round(mean(d1$FBS_B, na.rm=T))
d1$GOT_B <- ifelse(!is.na(d1$GOT_B), d1$GOT_B, round(mean(d1$GOT_B, na.rm=T))) # 4개의 이상치, 25로 대체
round(mean(d1$GOT_B, na.rm=T))
d1$GPT_B <- ifelse(!is.na(d1$GPT_B), d1$GPT_B, round(mean(d1$GPT_B, na.rm=T))) # 4개의 이상치, 26로 대체
round(mean(d1$GPT_B, na.rm=T))
d1$GGT_B <- ifelse(!is.na(d1$GGT_B), d1$GGT_B, round(mean(d1$GGT_B, na.rm=T))) # 8개의 이상치, 43로 대체
round(mean(d1$GGT_B, na.rm=T))
d1$URIC_B <- ifelse(!is.na(d1$URIC_B), d1$URIC_B, round(mean(d1$URIC_B, na.rm=T))) # 71개의 이상치, 5로 대체
round(mean(d1$URIC_B, na.rm=T))

# PCAN80 ~ PCAN89, FCAN80 ~ FCAN90 이상치 대체
d1$PCAN80 <- ifelse(!is.na(d1$PCAN80), d1$PCAN80, 0) # 996개의 이상치, 0으로 대체
d1$PCAN81 <- ifelse(!is.na(d1$PCAN81), d1$PCAN81, 0) # 985개의 이상치, 0으로 대체
d1$PCAN82 <- ifelse(!is.na(d1$PCAN82), d1$PCAN82, 0) # 1000개의 이상치, 0으로 대체
d1$PCAN83 <- ifelse(!is.na(d1$PCAN83), d1$PCAN83, 0) # 999개의 이상치, 0으로 대체
d1$PCAN84 <- ifelse(!is.na(d1$PCAN84), d1$PCAN84, 0) # 992개의 이상치, 0으로 대체
d1$PCAN86 <- ifelse(!is.na(d1$PCAN86), d1$PCAN86, 0) # 996개의 이상치, 0으로 대체
d1$PCAN89 <- ifelse(!is.na(d1$PCAN89), d1$PCAN89, 0) # 982개의 이상치, 0으로 대체

d1$FCAN80 <- ifelse(!is.na(d1$FCAN80), d1$FCAN80, 0) # 970개의 이상치, 0으로 대체
d1$FCAN81 <- ifelse(!is.na(d1$FCAN81), d1$FCAN81, 0) # 931개의 이상치, 0으로 대체
d1$FCAN82 <- ifelse(!is.na(d1$FCAN82), d1$FCAN82, 0) # 999개의 이상치, 0으로 대체
d1$FCAN83 <- ifelse(!is.na(d1$FCAN83), d1$FCAN83, 0) # 977개의 이상치, 0으로 대체
d1$FCAN84 <- ifelse(!is.na(d1$FCAN84), d1$FCAN84, 0) # 980개의 이상치, 0으로 대체
d1$FCAN86 <- ifelse(!is.na(d1$FCAN86), d1$FCAN86, 0) # 965개의 이상치, 0으로 대체
d1$FCAN89 <- ifelse(!is.na(d1$FCAN89), d1$FCAN89, 0) # 989개의 이상치, 0으로 대체

# FEV1, FVC
d1$FEV1
d1$FVC

# BIL, WBC, CREAT
d1$BIL <- ifelse(!is.na(d1$BIL), d1$BIL, round(mean(d1$BIL, na.rm=T),2)) # 52개의 이상치, 0.88로 대체
round(mean(d1$BIL, na.rm=T),2)
d1$WBC <- ifelse(!is.na(d1$WBC), d1$WBC, round(mean(d1$WBC, na.rm=T),2)) # 72개의 이상치, 6.09로 대체
round(mean(d1$WBC, na.rm=T),2)
d1$CREAT <- ifelse(!is.na(d1$CREAT), d1$CREAT, round(mean(d1$CREAT, na.rm=T),2)) # 38개의 이상치, 0.98로 대체
round(mean(d1$CREAT, na.rm=T),2)

colSums(is.na(d1))
```