Assignment 4

Michael Fry 300570669

2023-06-04

Question 1

a)

```
library(pander)

# Identify variables with missing data
vars_with_missing <- names(hd)[colSums(is.na(hd)) > 0]

# Create a table of missing data
missing_data_table <- data.frame(Variable = vars_with_missing,
          Frequency = colSums(is.na(hd[, vars_with_missing])),
          Proportion = round(colSums(is.na(hd[, vars_with_missing])) / nrow(hd), 5))

# Sort the table by the proportion of missing data (in descending order)
missing_data_table <- missing_data_table[order(-missing_data_table$Proportion), ]
missing_data_table <- t(missing_data_table)

# Print the table
pander(missing_data_table)
```

	GLUC	EDUC	BP_MED	CHOL	CIG	BMI	HR
Variable	GLUC	EDUC	BP_MED	CHOL	CIG	BMI	HR
Frequency	388	105	53	50	29	19	1
Proportion	0.09151	0.02476	0.01250	0.01179	0.00684	0.00448	0.00024

```
pander(missing_data_table[,1])
```

Variable	Frequency	Proportion
GLUC	388	0.09151

The variable with the highest level of missing data is GLUC, with 388 missing observations, and proportion of missing observations of 0.09151 or 9.2% missing observations.

b)

```
# Create a new data frame (hd.complete) without missing data
hd.complete <- hd[complete.cases(hd), ]
```

```r
# Calculate the proportion of removed observations
proportion_removed <- round((nrow(hd) - nrow(hd.complete)) / nrow(hd), 5)
```

The proportion of people that have been removed from the data set is 0.13726 or 13.73 percent.

c)

```r
# Create the SBP_CAT variable and categorize SBP readings
hd.complete$SBP_CAT <- cut(hd.complete$SBP,
          breaks = c(0, 120, 130, 140, 180, Inf),
          labels = c("normal", "elevated", "high_stage_1",
                     "high_stage_2", "hypertensive"), right = FALSE)

# Create a table showing the count of observations in each blood pressure range
table_SBP_CAT <- table(hd.complete$SBP_CAT)

# Print the table
pander(table_SBP_CAT)
```

normal	elevated	high_stage_1	high_stage_2	hypertensive
1104	815	629	965	145

d)

There are multiple times when transforming SBP to a categorical variable could lead to a better fit for a regression model.

When the relationship between SBP and the response variable is non-linear. By grouping SBP values into categories based on clinically relevant thresholds or risk levels, the regression model can better capture the relationship between SBP and the response variable

When there are irregular or sparce data in the SBP variable. When there are significant gaps in levels of numeric SBP, categorizing SBP can address the issue by assigning all values to a specific category which increases the number of observations in each group.

When outliers are present. Numeric variables like SBP may contain extreme values that deviate significantly from the overall model By grouping SBP values into categories, outliers are contained within groups, reducing their influence on the model.

Question 2

a)

```r
library(car)
```

```
## Loading required package: carData
```

```r
# Fit the logistic regression model
logistic_model <- glm(factor(HD_RISK) ~ factor(SEX) + AGE + factor(EDUC) +
                        CIG + CHOL + SBP + DBP + GLUC+ BMI,
                        data = hd.complete, family = binomial)

# Calculate the VIF for predictors
vif_values <- vif(logistic_model)
```

```
# Round the VIF values to 3 decimal places
vif_values <- round(vif_values, 3)

# Print the VIF values
pander(vif_values)
```

	GVIF	Df	GVIF^(1/(2*Df))
factor(SEX)	1.242	1	1.115
AGE	1.287	1	1.134
factor(EDUC)	1.104	3	1.017
CIG	1.241	1	1.114
CHOL	1.065	1	1.032
SBP	3.016	1	1.737
DBP	2.784	1	1.669
GLUC	1.019	1	1.01
BMI	1.18	1	1.086

All the predictors have VIF values well below the threshold of 10, and all are even well below a conservative cutoff of 5, suggesting that there is no evidence of significant multicollinearity among the predictors in the model.

b)

```
model_summary <- summary(logistic_model)

# Print the model output table
model_summary
```

```
##
## Call:
## glm(formula = factor(HD_RISK) ~ factor(SEX) + AGE + factor(EDUC) +
##     CIG + CHOL + SBP + DBP + GLUC + BMI, family = binomial, data = hd.complete)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9444  -0.5969  -0.4262  -0.2843   2.9063
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -8.962559   0.579182 -15.475  < 2e-16 ***
## factor(SEX)1   0.544414   0.108953   4.997 5.83e-07 ***
## AGE            0.063433   0.006695   9.474  < 2e-16 ***
## factor(EDUC)2 -0.188485   0.123171  -1.530   0.1259
## factor(EDUC)3 -0.196924   0.149848  -1.314   0.1888
## factor(EDUC)4 -0.052129   0.164125  -0.318   0.7508
## CIG            0.019463   0.004191   4.644 3.42e-06 ***
## CHOL           0.002371   0.001127   2.105   0.0353 *
## SBP            0.018249   0.003484   5.238 1.63e-07 ***
## DBP           -0.002798   0.006385  -0.438   0.6612
## GLUC           0.007186   0.001674   4.293 1.77e-05 ***
## BMI            0.006714   0.012639   0.531   0.5952
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3121.2  on 3657  degrees of freedom
## Residual deviance: 2758.8  on 3646  degrees of freedom
## AIC: 2782.8
##
## Number of Fisher Scoring iterations: 5
```

$log(p/1-p) = -8.9626(intercept) + 0.5444 * SEX + 0.0634 * AGE - 0.1885 * Education_2 - 0.1969 * Education_3 - 0.0521 * Education_4 + 0.0195 * CIG + 0.0024 * CHOL + 0.0182 * SBP - 0.0028 * DBP + 0.0072 * GLUC + 0.0067 * BMI$

where:

SEX represents the factor(SEX)1 (1 for male, 0 for female)

AGE represents the age variable

Education_2 represents the factor(EDUC)2 (2 for high school)

Education_3 represents the factor(EDUC)3 (3 for some college)

Education_4 represents the factor(EDUC)4 (4 for college graduate)

CIG represents the cigarettes smoked per day variable

CHOL represents the cholesterol level variable

SBP represents the systolic blood pressure variable

DBP represents the diastolic blood pressure variable

GLUC represents the glucose level variable

BMI represents the body mass index variable

c)

```r
# Extract the Wald test statistics for SBP and DBP coefficients
wald_sbp <- model_summary$coefficients["SBP", "Pr(>|z|)"]
wald_dbp <- model_summary$coefficients["DBP", "Pr(>|z|)"]

wald_sbp
```

```
## [1] 1.625958e-07
```

```r
wald_dbp
```

```
## [1] 0.6612366
```

Wald Test for SBP:

$H_0 : \beta_8 = 0$

$H_2 : \beta_8 \neq 0$

$z \approx \frac{0.003484}{0.018249} \approx 5.238$

$p - value = 2 \times P(Z > |5.238|) \approx 1.63 \times 10^{-7}$

As the p-value is much smaller than any reasonable significance level, we have sufficient evidence to suggest that Beta8 (SBP) is significantly different from 0, and there is a statistically significant relationship between SBP and HD_RISK, adjusting for all other variables.

Wald Test for DBP

$H_0 : \beta_9 = 0$

$H_2 : \beta_9 \neq 0$

$z \approx \frac{0.006385}{-0.002798} \approx -0.438$

$p - value = 2 \times P(Z > |-0.438|) \approx 0.6612$

As the p-value is much larger than any reasonable significance level, we have sufficient evidence to suggest that Beta9 (DBP) is not significantly different from 0, and there is not a statistically significant relationship between DBP and HD_RISK, adjusting for all other variables.

d)

To interpret the "effects" corresponding to the coefficient for SBP , we must exponentiate the estimated coefficient.

$\beta_8 \approx 0.018249$

$\exp(\beta_8) \approx 1.018417$

An increase in SBP by one mmHg is associated with an estimated multiplicative change of 1.018417 (95% CI: (1.011, 1.025)) in the odds of 10-year risk of future coronary heart disease, adjusting for all other variables.

```
pander(exp(confint.default(logistic_model, parm = 'SBP')))
```

	2.5 %	97.5 %
SBP	1.011	1.025

Note: Estimate of Beta 8 is significantly larger than 1 as the 95% confidence interval for Beta 8 does not include 1.

e)

```
# Fit the logistic regression model with SBP_CAT
logistic_model_cat <- glm(factor(HD_RISK) ~ factor(SEX) + AGE +
                            factor(EDUC) + CIG + CHOL +factor(SBP_CAT) +
                            DBP + GLUC + BMI, data = hd.complete, family = binomial)

# Get the summary of the logistic regression model with SBP_CAT
model_summary_cat <- summary(logistic_model_cat)

# Print the model output table for the model with SBP_CAT
print(model_summary_cat)
```

```
##
## Call:
## glm(formula = factor(HD_RISK) ~ factor(SEX) + AGE + factor(EDUC) +
##     CIG + CHOL + factor(SBP_CAT) + DBP + GLUC + BMI, family = binomial,
##     data = hd.complete)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max  
## -1.7466  -0.6025  -0.4283  -0.2843   2.8458  
## 
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)                 -7.790357   0.696157 -11.191  < 2e-16 ***
## factor(SEX)1                 0.517430   0.108387   4.774 1.81e-06 ***
## AGE                          0.067004   0.006634  10.100  < 2e-16 ***
## factor(EDUC)2               -0.190932   0.122913  -1.553 0.120329    
## factor(EDUC)3               -0.215698   0.149497  -1.443 0.149068    
## factor(EDUC)4               -0.061899   0.164188  -0.377 0.706173    
## CIG                          0.019659   0.004184   4.699 2.62e-06 ***
## CHOL                         0.002539   0.001126   2.256 0.024097 *  
## factor(SBP_CAT)elevated      0.234938   0.161846   1.452 0.146609    
## factor(SBP_CAT)high_stage_1  0.189517   0.177903   1.065 0.286746    
## factor(SBP_CAT)high_stage_2  0.592431   0.184339   3.214 0.001310 ** 
## factor(SBP_CAT)hypertensive  1.155536   0.298192   3.875 0.000107 ***
## DBP                          0.006231   0.005864   1.063 0.287955    
## GLUC                         0.007432   0.001667   4.457 8.29e-06 ***
## BMI                          0.006294   0.012617   0.499 0.617921    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 3121.2  on 3657  degrees of freedom
## Residual deviance: 2768.2  on 3643  degrees of freedom
## AIC: 2798.2
## 
## Number of Fisher Scoring iterations: 5
```

The p-value for the category “elevated” is approximately 0.147, indicating that there is no strong evidence to suggest a significant association between the “elevated” blood pressure range and the 10-year risk of CHD when compared to the “normal” systolic blood pressure.

The p-value for the category “high_stage_1” is approximately 0.287, suggesting that there is no strong evidence of a significant association between the “high_stage_1” systolic blood pressure range and the 10-year risk of CHD when compared to the “normal” systolic blood pressure.

The p-value for the category “high_stage_2” is approximately 0.001, indicating strong evidence of a significant association between the “high_stage_2” systolic blood pressure range and the 10-year risk of CHD when compared to the “normal” systolic blood pressure.

The p-value for the category “hypertensive” is approximately 0.0001, suggesting strong evidence of a significant association between the “hypertensive” systolic blood pressure range and the 10-year risk of CHD when compared to the “normal” systolic blood pressure.

These results align with the findings of Wu et al. (2015) you can clearly see that there is a relationship with the SBP and HD_RISK. The higher the SBP is compared to normal, the more significant the relationship with HD_RISK. These estimates increase with SBP indicating an alignment with that of Wu et al. (2015)

f)

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(zoo)
# Perform the likelihood ratio test
lr_test <- lrtest(logistic_model, logistic_model_cat)
pander(lr_test)
```

Table 6: Likelihood ratio test

#Df	LogLik	Df	Chisq	Pr(>Chisq)
12	-1379	NA	NA	NA
15	-1384	3	9.35	0.02498

```
?lrtest
```

Since this p-value is less than the conventional significance level of 0.05, we can conclude that the model using SBP_CAT provides a significantly better fit than the model using SBP.

Therefore, the model with SBP_CAT (model fit in part (e)) is considered to provide a better fit compared to the model with SBP (model from part (a)).

g)

```
library(ResourceSelection)
```

```
## ResourceSelection 0.3-5   2019-07-22
```

```
# Perform the Hosmer-Lemeshow test for g = 10
hoslem_10 <- hoslem.test(hd.complete$HD_RISK, logistic_model_cat$fitted.values, g = 10)
hoslem_20 <- hoslem.test(hd.complete$HD_RISK, logistic_model_cat$fitted.values, g = 20)
hoslem_30 <- hoslem.test(hd.complete$HD_RISK, logistic_model_cat$fitted.values, g = 30)

hoslem_table <- data.frame(
  G = c(10, 20, 30),
  p_value = c(hoslem_10$p.value, hoslem_20$p.value, hoslem_30$p.value)
)

# Print the table
pander(hoslem_table)
```

G	p_value
10	0.4073
20	0.4988
30	0.9349

The P-Values for all tests are significantly above 0.05. This suggests that the model provides a reasonable fit to the data.

Question 3

a)

```
library(MASS)



forward.selection <- stepAIC(glm(factor(HD_RISK) ~ 1, family = "binomial", data =
              hd.complete), scope = list(upper = ~ factor(SEX) + AGE + factor(EDUC)
              + factor(SMOKER) + CIG + factor(BP_MED) + factor(STROKE) +
              factor(HYPER) + factor(DIAB) + CHOL + SBP + DBP + BMI + HR + GLUC, lower
               = ~1), direction = "forward", trace = FALSE )
# Output the steps that were taken in the forward selection algorithm
# to produce the final model.
pander(forward.selection$anova)
```

Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
	NA	NA	3657	3121	3123
+ AGE	1	200.6	3656	2921	2925
+ SBP	1	65.01	3655	2856	2862
+ factor(SEX)	1	49.6	3654	2806	2814
+ CIG	1	19.99	3653	2786	2796
+ GLUC	1	19.12	3652	2767	2779
+ CHOL	1	4.081	3651	2763	2777
+ factor(HYPER)	1	2.986	3650	2760	2776
+ factor(STROKE)	1	2.287	3649	2757	2775

```
backward.selection <- stepAIC(glm(factor(HD_RISK) ~ factor(SEX) + AGE +
                factor(EDUC) + factor(SMOKER) + CIG + factor(BP_MED)
                + factor(STROKE) + factor(HYPER) + factor(DIAB) + CHOL
                + SBP + DBP + BMI + HR + GLUC, family = "binomial",
                data = hd.complete), scope = list(upper =
              ~ factor(SEX) + AGE + factor(EDUC) + factor(SMOKER) +
                CIG + factor(BP_MED) + factor(STROKE) +
                    factor(HYPER) + factor(DIAB) + CHOL + SBP + DBP + BMI +
                HR + GLUC, lower = ~1), direction = "backward", trace = FALSE)

# Output the steps that were taken in the backward selection algorithm
# to produce the final model.
pander(backward.selection$anova)
```

Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
	NA	NA	3640	2752	2788
- factor(EDUC)	3	3.244	3643	2755	2785
- factor(DIAB)	1	0.01826	3644	2755	2783
- factor(SMOKER)	1	0.2102	3645	2756	2782
- BMI	1	0.3619	3646	2756	2780
- DBP	1	0.407	3647	2756	2778
- factor(BP_MED)	1	0.4915	3648	2757	2777
- HR	1	0.5925	3649	2757	2775

```r
pander(forward.selection$coefficients)
```

Table 10: Table continues below

(Intercept)	AGE	SBP	factor(SEX)1	CIG	GLUC	CHOL	factor(HYPER)1	factor(STROKE)1
-8.746	0.06541	0.01422	0.5533	0.01958	0.007317	0.002257	0.2258	0.7517

```r
pander(backward.selection$coefficients)
```

Table 12: Table continues below

(Intercept)	factor(SEX)1	AGE	CIG	factor(STROKE)1	factor(HYPER)1	CHOL	SBP	GLUC
-8.746	0.5533	0.06541	0.01958	0.7517	0.2258	0.002257	0.01422	0.007317

Forward Selection Best Subset:

SEX, AGE, CIG, STROKE, HYPER, CHOL, SBP, GLUC

Backwards Selection Best Subset:

SEX, AGE, CIG, STROKE, HYPER, CHOL, SBP, GLUC

Both Forward and Backward selection algorithms included the same predictors in their optimal models.

b)

```r
library(bestglm)
```

```
## Loading required package: leaps
```

```r
# Create a data frame with predictors and response variable
predictors.for.bestglm <- data.frame(SEX = as.factor(hd.complete$SEX),
            AGE = hd.complete$AGE, EDUC =
           as.factor(hd.complete$EDUC), SMOKER = as.factor(hd.complete$SMOKER),
          CIG = hd.complete$CIG, PB_MED = as.factor(hd.complete$BP_MED),
          STROKE = as.factor(hd.complete$STROKE), HYPER = as.factor(hd.complete$HYPER),
          DIAB = as.factor(hd.complete$DIAB), CHOL = hd.complete$CHOL,
          SBP = hd.complete$SBP,DBP = hd.complete$DBP,BMI = hd.complete$BMI,
          HR = hd.complete$HR, GLUC = hd.complete$GLUC, y = as.factor(hd.complete$HD_RISK))

best.logistic.AIC <- bestglm(Xy = predictors.for.bestglm, family = binomial,
                             IC = "AIC", method = "exhaustive")
```

```
## Morgan-Tatar search since family is non-gaussian.
## Note: factors present with more than 2 levels.
```

```
## Show the top five models in terms of minimising AIC.
pander(best.logistic.AIC$BestModels)
```

Table 14: Table continues below

SEX	AGE	EDUC	SMOKER	CIG	PB_MED	STROKE	HYPER	DIAB	CHOL	SBP	DBP	BMI	HR	GLUC	Criterion
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	2773
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	2774
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	2774
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	2775
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	2775

```
# Find the best logistic regression model based on the predictors according
# to the criterion of  #minimising BIC.
best.logistic.BIC <- bestglm(Xy = predictors.for.bestglm, family = binomial,
                             IC = "BIC", method = "exhaustive")
```

```
## Morgan-Tatar search since family is non-gaussian.
## Note: factors present with more than 2 levels.
```

```
## Show the top five models in terms of minimising BIC.
pander(best.logistic.BIC$BestModels)
```

Table 16: Table continues below

SEX	AGE	EDUC	SMOKER	CIG	PB_MED	STROKE	HYPER	DIAB	CHOL	SBP	DBP	BMI	HR	GLUC	Criterion
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	2808
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	2812
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	2813
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	2814
TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	2815

AIC Selection Best Subset:

SEX, AGE, CIG, STROKE, HYPER, CHOL, SBP, GLUC

BIC Selection Best Subset:

SEX, AGE, CIG, STROKE, SBP, GLUC

AIC and BIC selection criterion models are nested. AIC included all predictors that BIC did, with the addition of HYPER and CHOL.

The optimal model produced by AIC is identical to that of both the Forward and Backward selection algorithms in part (a).

The difference in selection between BIC and AIC is likely down to the way the penalties are implemented between AIC and BIC with AIC being more lenient towards model complexity.

c)

Predictors identified in AIC selection above: SEX AGE CIG STROKE HYPER CHOL SBP GLUC

```
q3_dataset <- base::data.frame(HD_RISK = as.factor(hd.complete$HD_RISK),
              SEX = as.factor(hd.complete$SEX),
              AGE = hd.complete$AGE, CIG = hd.complete$CIG, STROKE =
              as.factor(hd.complete$STROKE), HYPER = as.factor(hd.complete$HYPER),
              CHOL = hd.complete$CHOL, SBP = hd.complete$SBP,
              GLUC = hd.complete$GLUC)

best_model_aic <- glm(factor(HD_RISK) ~ factor(SEX) + AGE + CIG +
                        factor(STROKE) + factor(HYPER) + CHOL + SBP +
                        GLUC, family = "binomial", data = hd.complete)

# Specify the indices of the variables to be considered in predictive models for survival
variable.indices <- 2 : 9

# Produce a matrix that represents all possible combinations of variables.
# Remove the first row, which is the null model (i.e., no predictors).
all.comb <- expand.grid(as.data.frame(matrix(rep(0 : 1,
                        length(variable.indices)), nrow = 2)))[-1, ]



library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```
library(doParallel)
```

```
## Loading required package: foreach

## Loading required package: iterators

## Loading required package: parallel
```

```
# Load the "foreach" package to allow for splitting loops.
library(foreach)

# Specify the number of folds to be considered in k-fold cross-validation.
folds <- 10
# Specify the number of repetitions of cross-validation to carry out.
```

```
nrep <- 20

# Fire up 75% of cores for parallel processing.
nclust <- makeCluster(detectCores() * 0.75)
registerDoParallel(nclust)

##############
## Accuracy ##
##############

# Specify settings for repeated 10-fold cross-validation for accuracy.
# This includes specifying seeds for consistency when splitting across cores.

fitControl <- trainControl(method = "repeatedcv", number = folds, repeats = nrep,
                           seeds = 1 :(folds * nrep + 1), classProbs = TRUE,
                           savePredictions = TRUE)

# Save estimated accuracy and standard errors for each set of covariates.
accuracy <- foreach(i = 1 : nrow(all.comb), .combine = "rbind",
                    .packages = "caret") %dopar%
{
c(i, unlist(train(as.formula(paste("make.names(HD_RISK) ~",
paste(names(q3_dataset)[variable.indices][all.comb[i,] == 1], collapse = " + "))), data
= q3_dataset, trControl = fitControl, method = "glm", family = "binomial", metric =
"Accuracy")$results[c(2, 4)]))
}

rownames(accuracy) <- NULL

##############################
## Area under the ROC curve ##
##############################

# Specify settings for repeated 10-fold cross-validation for AUC.
# This includes specifying seeds for consistency when splitting across cores.

fitControl <- trainControl(method = "repeatedcv", number = folds, repeats = nrep,
                           seeds = 1 :(folds * nrep + 1), summaryFunction =
                             twoClassSummary, classProbs = TRUE, savePredictions = TRUE)

# Save estimated AUC and standard errors for each set of covariates.
AUC <- foreach(i = 1 : nrow(all.comb), .combine = "rbind", .packages = "caret") %dopar%
{
c(i, unlist(train(as.formula(paste("make.names(HD_RISK) ~",
paste(names(q3_dataset)[variable.indices][all.comb[i,] == 1], collapse = " + "))), data
= q3_dataset, trControl = fitControl, method = "glm", family = "binomial", metric =
"ROC")$results[c(2, 5)]))
}

rownames(AUC) <- NULL

# Shut down cores.
stopCluster(nclust)
```

```
##############
## Accuracy ##
##############

# View the model that maximises accuracy.
max_accurace_variables <-
  names(q3_dataset)[variable.indices[all.comb[which.max(accuracy[, 2]), ] == 1]]

##############################
## Area under the ROC curve ##
##############################

# View the model that maximises AUC
max_auc_variables <-
  names(q3_dataset)[variable.indices[all.comb[which.max(AUC[, 2]), ] == 1]]

max_accurace_variables
```

```
## [1] "SEX"   "AGE"   "CIG"   "HYPER" "CHOL"  "SBP"   "GLUC"
```

```
max_auc_variables
```

```
## [1] "SEX"    "AGE"    "CIG"    "STROKE" "HYPER"  "CHOL"   "SBP"    "GLUC"
```

The optimal model identified with 20 repetition 10 fold cross validation maximizing accuracy included:

SEX, AGE, CIG, HYPER, CHOL, SBP, GLUC

This is identical to the BIC selection criterion with the addition of CHOL as a predictor.

The optimal model identified with 20 repetition 10 fold cross validation maximizing AUC included:

SEX, AGE, CIG, STROKE, HYPER, CHOL, SBP, GLUC.

This is identical to the AIC selection criterion seen in part (b), as well as the Forward and Backwards subset selection methods in part (a).

Maximizing accuracy and maximizing AUC as criteria for selecting the "best" models can lead to different outcomes because they focus on different aspects of model performance.

Maximizing accuracy aims to find the model that predicts the outcome with the highest overall correctness. It considers the proportion of correctly classified instances and disregards the balance between true positives and true negatives.

On the other hand, maximizing the Area Under the Receiver Operating Characteristic Curve (AUC) focuses on the trade-off between the true positive rate (sensitivity) and the false positive rate (1-specificity). AUC measures the ability of the model to discriminate between positive and negative instances and provides an overall assessment of the model's performance.

The differences in the "best" models identified by accuracy and AUC compared to the models in parts (a) and (b) can be attributed to the evaluation criteria and the dataset characteristics.

In the case of accuracy, the inclusion of CHOL as a predictor in the optimal model suggests that it contributes to improving the overall correctness of the predictions, regardless of its specific impact on true positives or true negatives. This indicates that CHOL has an influence on the correct classification of instances, even if it may not be strongly associated with the outcome of interest.

In the case of AUC, the inclusion of STROKE as a predictor suggests that it plays a significant role in the model's ability to discriminate between positive and negative instances. STROKE might not have been selected in the AIC or BIC models because its contribution to the overall goodness of fit was relatively small, but it has a noticeable impact on the model's discriminatory power.