DATA 303-DATA473 Assignment 4

Due: 11:59 PM Sunday 4 June 2023

Intructions

- Prepare your assignment using Rmarkdown
- Submit your solutions in two files: an Rmarkdown file named `assignment4.Rmd` and the PDF file named `assignment4.pdf` that results from knitting the Rmd file.
- The YAML header of your Rmarkdown file must contain your name and ID number in the author field, and should have the output format set to `pdf_document`. For example:

```
---
title: "DATA 303 Assignment 4"
author: "Ryan Admiraal, 12345678"
date: "26 June 2020"
output: pdf_document
---
```

- While you are developing your code you may find it easiest to have the output set to `html_document` but change it to `pdf_document` when you submit.
  - A common error that occurs in producing a PDF is when unicode characters (e.g., $\beta$, α) are included in your text. The error usually looks something like “! Package inputenc Error: Unicode char \u8: not set up for use with LaTeX.” or something similar. A potential workaround is provided at https://bookdown.org/yihui/rmarkdown-cookbook/latex-unicode.html. Alternatively, if you perform a Google search for the unicode character in question (e.g., “unicode \u8”), you can more readily identify the unicode character in question, find it in your Rmarkdown file, and replace it with a more appropriate character for knitting to PDF (e.g., $\beta$ can be produced by typing “$\beta$”).
- In your submission, embed any executable R code in code chunks, and make sure both the R code and the output is displayed correctly when you knit the document.
- If there are any R code errors, then the Rmarkdown file will not knit, and no output will be created at all. If you cannot get your code to work but want to show your attempted code, then put `error = TRUE` in the header of the R code chunk that is failing.

~~~
```{r, error = TRUE}
your imperfect R code
```
~~~

- Where appropriate, make sure you include your comments in the output within the Rmarkdown document.
- **You will receive an email confirming your submission. Check the email to be sure it shows that both the Rmd file and the PDF file have been submitted.**

Background and Data

Heart disease is the annual leading cause of death worldwide, accounting for more than 25% of deaths in 2016 (World Health Organization 2018). It is also a significant economic burden for the healthcare system with Nichols et al. (2010) estimating that heart disease and other cardiovascular diseases cost an average of roughly USD $19,000 per patient, according to a study in the United States over the period of 2000-2005. Early detection of heart disease (along with many other diseases) is important in terms of reducing both mortality and costs to the healthcare system.

We will examine data on 4,240 participants in the Framingham Heart Study (Boston University and the National Heart, Lung, & Blood Institute 2020), an ongoing study that began in 1948 and has been instrumental in the identification of a number of risk factors for heart disease and other cardiovascular diseases. The data are available in the file `Framingham Heart Study.xlsx`, which can be read into R using the code below but with the path changed to point to the location of the file on your computer. A full list of variables contained in the dataset and descriptions of these variables is also provided, both here and in the Excel file.

```r
# Load the "readxl" package to read in data from an Excel file.
library(readxl)
# Read in the heart disease dataset.
hd <- read_xlsx("~/Documents/Dropbox/Courses/DATA303/Data/Framingham Heart Study.xlsx",
sheet = "Data", na = "NA")
```

Table 1: Variables and their descriptions for data contained in the file `Framingham Heart Study.xlsx`.

Variable	Description
`SEX`	Sex of the individual (0 = “Female”, 1 = “Male”).
`AGE`	Age (in years) of the individual at the time of the health exam.
`EDUC`	Highest level of education of the individual (1 = “Some high school”, 2 = “High school or Graduate Equivalency Diploma”, 3 = “Some university or vocational school”, 4 = “University”).
`SMOKER`	Indicator of whether or not the individual is a current smoker (0 = “No”, 1 = “Yes”).
`CIG`	Average number of cigarettes that the individual smokes each day.
`BP_MED`	Indicator of whether or not the individual is on blood pressure medication (0 = “No”, 1 = “Yes”).
`STROKE`	Indicator of whether or not the individual previously had a stroke (0 = “No”, 1 = “Yes”).
`HYPER`	Indicator of whether or not the individual was hypertensive (0 = “No”, 1 = “Yes”).
`DIAB`	Indicator of whether or not the individual is diabetic (0 = “No”, 1 = “Yes”).
`CHOL`	Total cholesterol level (in mg/dL).
`SBP`	Systolic blood pressure (in mmHg).
`DBP`	Diastolic blood pressure (in mmHg).
`BMI`	Body mass index.
`HR`	Resting heart rate (in beats per minute)
`GLUC`	Glucose level (in mg/dL)
`HD_RISK`	Indicator of whether the individual has 10-year risk of future coronary heart disease (0 = “No”, 1 = “Yes”)

Our focus will be on 10-year risk of coronary heart disease (CHD). Ten-year risk of CHD is a predicted risk (*i.e.*, a probability ranging between 0 and 1) of developing CHD within the next 10 years. Although this is not an observed outcome but rather an estimated value, 10-year risk of CHD is a well-established measure in the medical community. We will consider a dichotomised version of this variable which indicates whether or not a person would be considered as at risk of developing CHD within the next 10 years.

Assignment Questions

1. **Missing data and variable recode: (10 marks)**

   Although our objective will be to consider inferential and predictive models for 10-year risk of CHD, we will first ensure that we understand aspects of the underlying data as well as create a new variable that may prove useful in producing comparisons of 10-year risk of CHD for medically-meaningful blood pressure ranges. (In practice, we would want to examine each relevant variable to identify extreme observations and be sure that there are not any erroneous values. As this dataset has already been cleaned, we will not do so for this assignment.)

   a. **(2 marks) First, perform an analysis of the level of missing data for each variable. For only those variables for which there are missing data, produce a table of the form shown below, where `VARIABLE_i` is the name of the variable with missing data, n_i is the count for number of missing observations for that variable, and p_i is the proportion (to 5dp) of missing observations for that variable. Which variable has the highest level of missing data?**

Table 2: Frequency and proportion of missing values for variables with missing data.

Variable	VARIABLE_1	VARIABLE_2	...	VARIABLE_k
Frequency (n)	n_1	n_2	...	n_k
Proportion (p)	p_1	p_2	...	p_k

Summary output for individual variables can be produced by applying the `summary` function to the data frame in which we have stored the dataset, and this gives a five number summary as well as the mean and number of missing observations. Example code is shown below, although output is suppressed.

```r
# Produce a table of summary output for individual variables.
summary(hd)
```

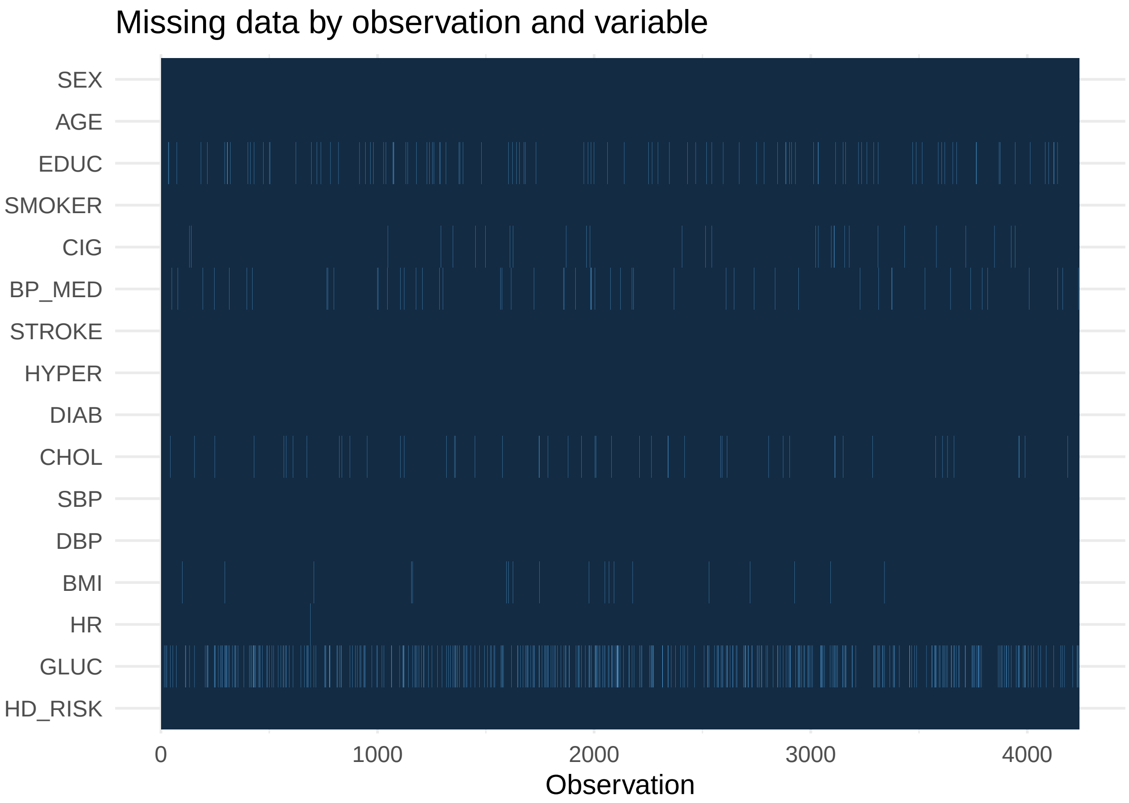
Missing data totals and proportions of observations missing by variable are as shown in the table below. A missing data plot is shown as well to simultaneously visualise missing data by observation and variable. These show the highest level of missing data occurring for glucose measurements (9.15%), followed by highest level of education (2.48%), use of blood pressure medication (1.25%), and total cholesterol level (1.18%).

Table 3: Frequency and proportion of missing values for variables with missing data.

Variable	EDUC	CIG	BP_MED	CHOL	BMI	HR	GLUC
Frequency (n)	105	29	53	50	19	1	388
Proportion (p)	0.02476	0.00684	0.0125	0.01179	0.00448	2.4×10^{-4}	0.09151

```r
# Load the "dplyr" package for data frame manipulation functionality.
library(dplyr)
# Load the "finalfit" package for visualisation of missing data by variable.
library(finalfit)
```

```
# Produce a missing data plot for the data frame.
hd %>% missing_plot(title = "Missing data by observation and variable")
```



b. **(3 marks) Create a new data frame called `hd.complete`, which only keeps people/observations that have no missing data. In total, what proportion (to 5dp) of people have been removed from the original dataset to produce this final data frame?**

The following code keeps only complete cases:

```
# Only keep observations without missing data for the predictors.
hd.complete <- hd[complete.cases(hd), ]
# Calculate the number of observations removed in producing the reduced data frame.
nrow(hd) - nrow(hd.complete)
```

```
## [1] 582
```

```
# Calculate the percentage of observations removed in producing the reduced data frame.
round((nrow(hd) - nrow(hd.complete)) / nrow(hd), 5)
```

```
## [1] 0.13726
```

In total, 582 people were removed due to missing data for at least one variable to produce the final data frame. These 582 people comprise approximately 13.726% of the original dataset size.

c. **(3 marks) Add a variable to the data frame `hd.complete` called `SBP_CAT`, which converts systolic blood pressure (`SBP`) from a numeric variable to a categorical variable according to the blood pressure ranges specified by Madell and Cherney (2018). (See references listed at the end of the assignment.) For the purposes of coding `SBP_CAT`, you can assume that the values for each blood pressure category go to just below that of the next category, as our dataset does not consist of blood pressures that are rounded to the nearest whole number. This means that, for instance, the systolic blood**

pressure range of 120 – 129 should in fact be interpreted as 120 – < 130. This should produce five levels (*i.e.*, blood pressure ranges) for `SBP_CAT`. (Note that the final level corresponds to systolic blood pressure above 180 mmHg.) Produce a table for `SBP_CAT` which shows how many observations fall into each blood pressure range.

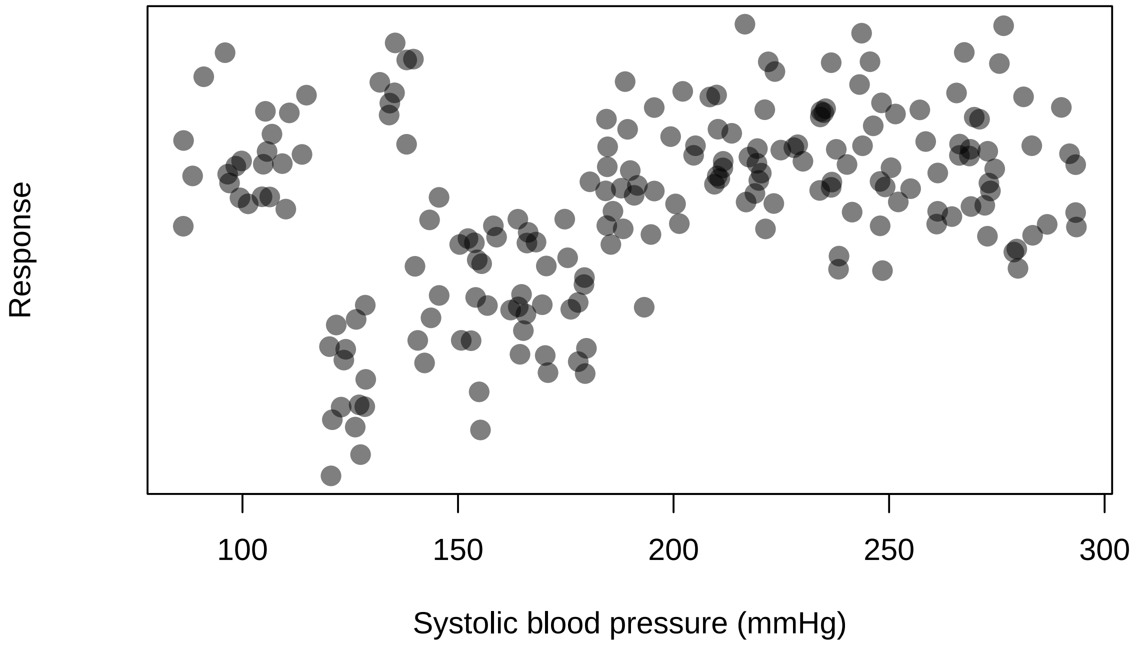
```
# Load the "memisc" package for simpler recoding of numeric variables to factors.
library(memisc)

# Construct the variable SBP_CAT and add to the data frame.
hd.complete <- hd.complete %>% mutate(SBP_CAT = memisc::recode(SBP, 1 <- range(0,
119.999), 2 <- range(120, 129.999), 3 <- range(130, 139.999), 4 <- range(140, 180), 5 <-
range(180.001, 300)))

# Load the "pander" library for pretty tables.
library(pander)
# Produce a table of counts according to the various blood pressure categories.
pander(table(hd.complete$SBP_CAT), caption = "Counts of observations by systolic blood
pressure range.", col.names = c("< 120", "120 -- < 130", "130 -- < 140", "140 -- 180", ">
180"), big.mark = ",")
```

Table 4: Counts of observations by systolic blood pressure range.

< 120	120 – < 130	130 – < 140	140 – 180	> 180
1,104	815	629	974	136

d. **(2 marks) Explain when we would expect that using the categorical variable `SBP_CAT` rather than the numeric variable `SBP` would lead to a better fit for a regression model (whether logistic regression, linear regression, or Poisson regression).**

The categorical variable for systolic blood pressure adds additional complexity to a model by introducing more parameters (four additional parameters, to be exact). This additional complexity can lead to a better model fit if the relationship between the response and `SBP` is non-linear. If the relationship between the response and `SBP` is well-approximated by a polynomial, then it may be that introduction of quadratic (and possibly cubic) terms for `SBP` can capture the non-linearity well and still result in a simpler model than using `SBP_CAT`. However, if the values of the response fluctuate in an odd manner not well-approximated by a smooth function (such as in the example plot below), then a categorical variable may be more readily able to fit these data than a polynomial function or spline.



2. **Inferential analysis: (25 marks)**

Now we will focus on 10-year risk of CHD and look at the role that blood pressure may play in whether or not someone is considered to be at risk of developing CHD within the next 10 years.

a. **(3 marks) We will first consider a logistic regression model of 10-year risk of CHD (`HD_RISK`) on systolic blood pressure (`SBP`) and diastolic blood pressure (`DBP`). Previous research suggests that the following variables are potential confounders for the true relationship between blood pressure and 10-year risk of CHD and should also be included in the logistic regression model:**

- **sex of the individual (`SEX`)**
- **age of the individual (`AGE`)**
- **highest level of education of the individual (`EDUC`)**
- **average number of cigarettes smoked per day (`CIG`)**
- **total cholesterol level (`CHOL`)**
- **body mass index (`BMI`)**
- **glucose level (`GLUC`)**

For this logistic regression model, calculate the variance inflation factors for predictors (to 3dp) to determine whether or not there is evidence of significant multicollinearity among the predictors in the model. If so, comment on which predictor(s) should be removed, and use this model for subsequent parts of this question.

```r
# Fit a logistic regression of 10-year risk of coronary heart diseases (HD_RISK) on:
# 1. systolic blood pressure (SBP)
# 2. diastolic blood pressure (DBP)
# 3. sex of the individual (SEX)
# 4. age of the individual (AGE)
# 5. highest level of education of the individual (EDUC)
# 6. average number of cigarettes smoked per day (CIG)
# 7. total cholesterol level (CHOL)
# 8. body mass index (BMI)
# 9. glucose level (GLUC)
hd.model <- glm(HD_RISK ~ SBP + DBP + factor(SEX) + AGE + factor(EDUC) + CIG + CHOL + BMI
+ GLUC, family = "binomial", data = hd.complete)
# Load the "car" package to make use of the vif() function.
library(car)
# Calculate variance inflation factors for predictors.
pander(vif(hd.model), caption = "Variance inflation factors for predictors to be included
in the logistic regression model")
```

Table 5: Variance inflation factors for predictors to be included in the logistic regression model

	GVIF	Df	GVIF^(1/(2*Df))
SBP	3.016	1	1.737
DBP	2.784	1	1.669
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
BMI	1.18	1	1.086
GLUC	1.019	1	1.01

Variance inflation factors (VIFs) are as shown in the table above. The largest VIF is approximately 3.016, which is well below 10, alleviating concerns about multicollinearity of predictors.

b. **(3 marks) Using your model from part (a), produce a table of logistic regression model output and write out the estimated logistic regression equation using the form**

$$\log\left(\frac{\widehat{p}}{1-\widehat{p}}\right) \quad = \quad \widehat{\beta}_0 + \widehat{\beta}_1 X_1 + \cdots + \widehat{\beta}_k X_k,$$

where you clearly define the variables X_1, X_2, . . . , X_k and replace $\widehat{\beta}_0$, $\widehat{\beta}_1$, . . . , $\widehat{\beta}_k$ with their estimated values (to 4dp).

Summary output for the logistic regression model is as shown in the table below.

```
# Produce a table of summary output for the logistic regression model.
pander(summary(hd.model))
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-8.963	0.5792	-15.47	5.157e-54
SBP	0.01825	0.003484	5.238	1.626e-07
DBP	-0.002798	0.006385	-0.4382	0.6612
factor(SEX)1	0.5444	0.109	4.997	5.83e-07
AGE	0.06343	0.006695	9.474	2.685e-21
factor(EDUC)2	-0.1885	0.1232	-1.53	0.1259
factor(EDUC)3	-0.1969	0.1498	-1.314	0.1888
factor(EDUC)4	-0.05213	0.1641	-0.3176	0.7508
CIG	0.01946	0.004191	4.644	3.424e-06
CHOL	0.002371	0.001127	2.105	0.0353
BMI	0.006714	0.01264	0.5313	0.5952
GLUC	0.007186	0.001674	4.293	1.766e-05

(Dispersion parameter for binomial family taken to be 1)

Null deviance:	3121 on 3657 degrees of freedom
Residual deviance:	2759 on 3646 degrees of freedom

Let variables be as defined in the following table:

Table 8: Variables used in the logistic regression model equation and description.

Variable	Description
X_1	Systolic blood pressure (in mmHg)
X_2	Diastolic blood pressure (in mmHg)
X_3	Indicator of whether the person is male (0 = “No”, 1 = “Yes”)
X_4	Age (in years)
X_5	Education level (1 = “Some high school”, 2 = “High school or Graduate Equivalency Diploma”, 3 = “Some university or vocational school”, 4 = “University”)
X_6	Average number of cigarettes smoked per day
X_7	Total cholesterol level (in mg/dL)
X_8	Body mass index
X_9	Glucose level

Then the estimated regression equation is

$$\begin{aligned}\log\left(\frac{\widehat{p}}{1-\widehat{p}}\right) &= -8.9626 + 0.0182X_1 - 0.0028X_2 + 0.5444X_3 + 0.0634X_4 - 0.1885X_{52} \\ &\quad -0.1969X_{53} - 0.0521X_{54} + 0.0195X_6 + 0.0024X_7 + 0.0067X_8 + 0.0072X_9\end{aligned}$$

c. **(6 marks) Carry out Wald tests for the coefficients for**

- **systolic blood pressure and**
- **diastolic blood pressure.**

For each coefficient, clearly state

i. **the hypotheses you are testing,**
ii. **the value of the test statistic,**
iii. **the p-value, and**
iv. **your conclusion in terms of whether the “effect” of the predictor on the response is statistically significant.**

In our model the coefficients for systolic blood pressure and diastolic blood pressure correspond to β_1 and β_2, respectively. A test of

$$\begin{aligned}\mathcal{H}_0 : \beta_1 &= 0 \\ \mathcal{H}_1 : \beta_1 &\neq 0\end{aligned}$$

produces a test statistic of

$$z \approx \frac{0.01825}{0.00348} \approx 5.2377$$

and corresponding p-value of

$$p\text{-value} = 2 \times P\left(Z > |5.2377|\right) \approx 1.626 \times 10^{-7}.$$

As the p-value is much smaller than any reasonable significance level α (*e.g.*, $\alpha = 0.05$, 0.01), we have strong evidence to suggest that β_1 is significantly different from 0, and there is a statistically significant relationship between systolic blood pressure and 10-year risk of CHD, adjusting for diastolic blood pressure, sex of the person, age of the person, highest level of education of the person, average number of cigarettes smoked per day, total cholesterol level, body mass index, and glucose level.

A test of

$$\begin{aligned} \mathcal{H}_0 : \beta_2 &= 0 \\ \mathcal{H}_1 : \beta_2 &\neq 0, \end{aligned}$$

on the other hand, produces a test statistic of

$$z \approx \frac{-0.0028}{0.00638} \approx -0.4382$$

and corresponding p-value of

$$p\text{-value} = 2 \times P\left(Z > |-0.4382|\right) \approx 0.6612.$$

As the p-value is larger than $\alpha = 0.05$, we have insufficient evidence to suggest that β_2 is significantly different from 0. Thus, there is not a statistically significant relationship between systolic blood pressure and 10-year risk of CHD, adjusting for systolic blood pressure, sex of the person, age of the person, highest level of education of the person, average number of cigarettes smoked per day, total cholesterol level, body mass index, and glucose level.

d. **(3 marks) For any significant Wald tests in part (c), provide a precise interpetation of what the estimated coefficient suggests about the "effect" of the predictor on the response, and calculate a corresponding 95% confidence interval (to 3dp) for the estimated "effect".**

From part (c), only the Wald test for the coefficient corresponding to systolic blood pressure is statistically significant. To interpret this coefficient, we exponentiate it. Similarly, a corresponding 95% confidence interval is obtained by exponentiating the endpoints for a 95% confidence interval for β_1. The exponentiated coefficient and corresponding confidence interval are as produced by the code shown below.

```r
# Produce an estimate for the odds ratio giving the "effect" for SBP.
exp(hd.model$coefficients[2])
```

```
##      SBP
## 1.018416
```

```r
# Produce a corresponding 95% confidence interval
pander(exp(confint.default(hd.model, parm = "SBP")))
```

	2.5 %	97.5 %
SBP	1.011	1.025

Interpreting this, an increase in systolic blood pressure by 1 mmHg is associated with a multiplicative change of 1.018 (95% CI: (1.011, 1.025)) in the odds of 10-year risk of CHD, adjusting for diastolic blood pressure, sex of the person, age of the person, highest level of education of the person, average number of cigarettes smoked per day, total cholesterol level, body mass index, and glucose level. (This means a roughly 1.84% increase in the odds of 10-year risk of CHD per 1 mmHg increase in systolic blood pressure.)

e. **(4 marks) A 2015 study by Wu et al. (2015) found that**

> **“cardiovascular and expanded-cardiovascular mortality risks were lowest when systolic blood pressures were 120 to 129 mm Hg, and increased significantly when systolic blood pressures (SBPs) were $\geq$ 160 mm Hg. . . ”**

Although Wu et al. (2015) considered different ranges of systolic blood pressures ($<$ 120, 120-–129, 130-–139, 140-–149, 150—159, $\geq$ 160 mmHg) than Madell and Cherney (2018), we will use those specified by Madell and Cherney (2018) in investigating whether ranges of blood pressures may differ in terms of associated 10-year risk of CHD.

Fit the same model as before, but replace `SBP` with `SBP_CAT`.

i. **Produce a table of logistic regression model output for this model.**
ii. **Based strictly on *p*-values, comment on what conclusions you would make for Wald tests based on coefficients for `SBP_CAT`. (Note that you do not need to state hypotheses or values for test statistics. You simply need to use the *p*-values to explain what these results mean about comparisons of systolic blood pressure ranges.)**
iii. **Do your results agree with the findings of Wu et al. (2015)?**

A summary of model output is shown in the table below. Only the *p*-values correspondinng to Wald tests for the coefficients of the last two levels of `SBP_CAT` (140–180 mmHg, $>$ 180 mmHg) are less than $\alpha = 0.05$, suggesting that these are the only coefficients for `SBP_CAT` that are significantly different than 0 (and, in this case, significantly greater than 0). As all coefficients represent comparisons with the reference level ($<$ 120 mmHg), this means that those with systolic blood pressures between $120 - < 130$ and $130 - < 140$ mmHg do not have a significantly different 10-year risk of CHD than those with systolic blood pressures less than 120 mmHg, whereas those with systolic blood pressures of 140–180 mmHg or more than 180 mmHg have significantly higher 10-year risk of CHD than those with systolic blood pressures less than 120 mmHg, adjusting for diastolic blood pressure, sex of the person, age of the person, highest level of education of the person, average number of cigarettes smoked per day, total cholesterol level, body mass index, and glucose level.

These results are not exactly the same as those of Wu et al. (2015). Although we do seem to see evidence that those with higher blood pressures seem to have higher 10-year risk of CHD (as based on increasing coefficients for higher systolic blood pressure ranges), which would not contradict their findings that risk was higher for those with systolic blood pressures of at least 160 mmHg, we do not have evidence that systolic blood pressures in the range of 120–129 mmHg lead to lowest 10-year risk of CHD. (We do not find a significant difference between 10-year risk of CHD for any of the three lowest systolic blood blood pressure ranges.)

Why might our results differ? First, our response variables are not identical. Wu et al. (2015) investigated actual mortalities due to heart disease and other cardiovascular diseases, whereas we are considering 10-year risk of CHD. (They also used a different class of models that are used for survival data.) Second, the data used by Wu et al. (2015) are from a Taiwanese population, whereas the data we used are from an American population. The relationship between systolic blood pressure and CHD may be different from different races. Finally, the predictors that we have considered are not identical to those used by Wu et al. (2015). They accounted for marital status, alcohol consumption, and other physiological measurements that we do not have available in our dataset.

```r
hd.model.cat <- glm(HD_RISK ~ factor(SBP_CAT) + DBP + factor(SEX) + AGE + factor(EDUC) +
CIG + CHOL + BMI + GLUC, family = "binomial", data = hd.complete)
pander(summary(hd.model.cat))
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-7.796	0.6946	-11.22	3.125e-29
factor(SBP_CAT)2	0.2344	0.1618	1.449	0.1474
factor(SBP_CAT)3	0.1886	0.1778	1.061	0.2887
factor(SBP_CAT)4	0.5934	0.1843	3.219	0.001284
factor(SBP_CAT)5	1.176	0.3017	3.897	9.731e-05
DBP	0.006337	0.005836	1.086	0.2776
factor(SEX)1	0.5178	0.1084	4.777	1.779e-06
AGE	0.06698	0.006634	10.1	5.724e-24
factor(EDUC)2	-0.1937	0.1229	-1.576	0.1151
factor(EDUC)3	-0.2164	0.1494	-1.448	0.1476
factor(EDUC)4	-0.0618	0.1642	-0.3765	0.7066
CIG	0.01976	0.004183	4.723	2.32e-06
CHOL	0.002541	0.001126	2.258	0.02397
BMI	0.006201	0.01262	0.4915	0.6231
GLUC	0.007434	0.001668	4.456	8.342e-06

(Dispersion parameter for binomial family taken to be 1)

Null deviance:	3121 on 3657 degrees of freedom
Residual deviance:	2768 on 3643 degrees of freedom

f. **(3 marks) Does the model that uses `SBP_CAT` (*i.e.*, the model fit in part (e)) provide a better fit than the model that uses `SBP` (*i.e.*, the model from part (a))?**

A likelihood ratio test using the `lrtest` function to compare the two models is as shown in the table below. The p-value of approximately 0.0271 is less than $\alpha = 0.05$, which would *seem* to suggest that the additional complexity introduced to the model by treating systolic blood pressure as categorical according to ranges of blood pressures (which adds three parameters to the model) leads to a significant improvement in model fit.

```
# Load the "lmtest" package to make use of the lrtest() function.
library(lmtest)
# Carry out a likelihood ratio test comparing the full model to the reduced model.
pander(lrtest(hd.model, hd.model.cat), caption = "Likelihood ratio test comparing a
logistic regression model with systolic blood pressure treated as numeric (reduced model)
with a logistic regression model with systolic blood pressure treated as categorical
(full model).")
```

Table 12: Likelihood ratio test comparing a logistic regression model with systolic blood pressure treated as numeric (reduced model) with a logistic regression model with systolic blood pressure treated as categorical (full model).

#Df	LogLik	Df	Chisq	Pr(>Chisq)
12	-1379	NA	NA	NA
15	-1384	3	9.172	0.02709

In fact, this is not the case. If we let M_0 denote the model including `SBP` (reduced model) and M_1 denote the model including `SBP_CAT` (full model), then, as shown in summary output for the two models, model deviances are given by

$$\begin{aligned} G^2(M_0) &\approx 2758.83 \\ G^2(M_1) &\approx 2768 \end{aligned}$$

The likelihood ratio test statistic is then given by

$$\begin{aligned} G^2 &= G^2(M_0) - G^2(M_1) \\ &\approx -9.17, \end{aligned}$$

which follows a $\chi^2_{3646-3643} = \chi^2_3$ distribution under $\mathcal{H}_0$. The p-value for this test is given by

$$p\text{-value} \approx P\left(\chi^2_3 > -9.17\right) = 1,$$

which far exceeds any reasonable significance level. Consequently, we would conclude that the fit of the reduced model using `SBP` is better than that of the full model using `SBP_CAT`.

g. **(3 marks) Finally, for the best model of the two you fit (in parts (a) and (e)), perform a Hosmer-Lemeshow test for g = 10, 20, and 30 groups, and comment on what these suggest about the goodness-of-fit of this model to the 10-year risk of CHD data.**

Using the model that uses `SBP`, we get the results for Hosmer-Lemeshow tests based on g = 10, 20, and 30 groups as shown in the tables below. For each of these tests, the p-value is much higher than $\alpha = 0.05$ (the lowest p-value is 0.2662), suggesting that the model provides a reasonable fit to the 10-year risk of CHD data.

```
# Load the "ResourceSelection" package to make use of the hoslem.test() function.
library(ResourceSelection)
# Carry out Hosmer-Lemeshow tests for g = 10, 20, and 30 groups.
pander(hoslem.test(hd.complete$HD_RISK, hd.model$fitted.values, g = 10))
pander(hoslem.test(hd.complete$HD_RISK, hd.model$fitted.values, g = 20))
pander(hoslem.test(hd.complete$HD_RISK, hd.model$fitted.values, g = 30))
```

Table 13: Hosmer-Lemeshow test for $g = 10$ groups.

Test statistic	df	P value
9.983	8	0.2662

Table 14: Hosmer-Lemeshow test for $g = 20$ groups.

Test statistic	df	P value
19.26	18	0.3758

Table 15: Hosmer-Lemeshow test for $g = 30$ groups.

Test statistic	df	P value
17.7	28	0.9335

3. **Statistical learning: (15 marks)**

Now we perform an exploratory analysis to try to identify the best set of predictors in predicting 10-year risk of CHD. Consider as predictors all variables other than the new variable that you constructed in Question 1 (`SBP_CAT`).

a. **(4 marks) Find the optimal models identified by forward and backward selection algorithms. Report the predictors included in these optimal models. If these models are different, highlight how they differ, and explain why forward and backward selection algorithms may not arrive at the same optimal model.**

First, to simplify coding later on, we ensure that all predictors that are categorical are stored as factors. We then create a new data frame that does not include the categorical variables we constructed for systolic blood pressure. The code below accomplishes this.

```
# Overwrite variables that are factors but are currently stored as numeric.
hd.complete$SEX <- as.factor(hd.complete$SEX)
hd.complete$EDUC <- as.factor(hd.complete$EDUC)
hd.complete$SMOKER <- as.factor(hd.complete$SMOKER)
hd.complete$BP_MED <- as.factor(hd.complete$BP_MED)
hd.complete$STROKE <- as.factor(hd.complete$STROKE)
hd.complete$HYPER <- as.factor(hd.complete$HYPER)
hd.complete$DIAB <- as.factor(hd.complete$DIAB)
hd.complete$SBP_CAT <- as.factor(hd.complete$SBP_CAT)

# Construct a reduced dataset that does not include the variable SBP_CAT.
hd.reduced <- hd.complete[, 1 : 16]
```

With this in place, we now perform forward and backward selection. With forward selection, we start with an empty model and sequentially add predictors that lead to the most significant improvement in fit until no predictors can be added that lead to a better model fit. The steps that were taken by the forward selection algorithm are as shown in the table below.

```
# Load the "MASS" package to make use of the stepAIC() function.
library(MASS)
# Perform forward selection for models based on the specified predictors.  Start with an
empty model.
forward.selection.hd <- stepAIC(glm(HD_RISK ~ 1, family = "binomial", data = hd.reduced),
scope = list(upper = as.formula(paste("~", paste(names(hd.reduced)[names(hd.reduced) !=
"HD_RISK"], collapse = " + "))), 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.hd$anova, caption = "Steps taken by forward selection in adding
predictors to the model.")
```

Table 16: Steps taken by forward selection in adding predictors to the model.

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
+ 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
+ HYPER	1	2.986	3650	2760	2776
+ STROKE	1	2.287	3649	2757	2775

Backward selection starts with a full model and sequentially removes predictors that produce likelihood ratio tests with the highest (non-statistically significant) p-values until no more predictors can be removed. The steps that were taken by the backward selection algorithm are as shown in the table below.

```r
# Perform backward selection for models based on the specified predictors.  Start with a
full model.
backward.selection.hd <- stepAIC(glm(as.formula(paste("HD_RISK ~",
paste(names(hd.reduced)[names(hd.reduced) != "HD_RISK"], collapse = " + "))), family =
"binomial", data = hd.reduced), scope = list(upper = as.formula(paste("~",
paste(names(hd.reduced)[names(hd.reduced) != "HD_RISK"], collapse = " + "))), 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.hd$anova, caption = "Steps taken by backward selection in
removing predictors from the model.")
```

Table 17: Steps taken by backward selection in removing predictors from the model.

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

The table below provides a comparison of the predictors that are included in the "best" models selected by forward selection and backward selection algorithms. We know that it is possible for forward and backward selection algorithms to arrive at different final models, as stepwise selection algorithms are "greedy" algorithms which simply take the optimal choice at each step, rather than exploring all possible subsets of predictors. In this case, however, forward and backward selection algorithms select the same set of predictors (`SEX`, `AGE`, `CIG`, `STROKE`, `HYPER`, `CHOL`, `SBP`, and `GLUC`).

Table 18: Predictors included (✓) in optimal models selected by forward and backward selection algorithms.

Variable	Forward selection	Backward selection
`SEX`	✓	✓
`AGE`	✓	✓
`EDUC`		
`SMOKER`		
`CIG`	✓	✓
`BP_MED`		
`STROKE`	✓	✓
`HYPER`	✓	✓
`DIAB`		
`CHOL`	✓	✓
`SBP`	✓	✓
`DBP`		
`BMI`		
`HR`		

Variable	Forward selection	Backward selection
`GLUC`	✓	✓

b. **(5 marks) Find the optimal models identified by best subset selection using AIC and BIC as selection criteria. Report the predictors included in these optimal models. If these models are different, highlight how they differ, and explain why the criteria of AIC and BIC may lead to different "best" models. If these models differ from those identified as "best" by forward and backward selection, explain why that may be the case.**

For best subset selection according to minimising AIC or minimising BIC, we must first construct a data frame that includes all of the predictors and has the response as the last variable. Our reduced dataset already has this structure, but the name of the last variable must be changed from `HD_RISK` to `y`. The code below does this.

```
# Construct a data frame to be used by the bestglm() function.
# The structure of this data frame is rigid with predictors first and the response being
placed in the last column.
# Note that the response variable MUST be named 'y' in the data frame.
predictors.for.bestglm <- data.frame(hd.reduced)
names(predictors.for.bestglm)[ncol(predictors.for.bestglm)] <- "y"
```

The tables below show predictors included in the top five models according to the criteria of minimising AIC and minimising BIC.

```
# Load the "bestglm" library to make use of the bestglm() function.
library(bestglm)
# Find the best logistic regression model based on the predictors according to the
criterion of minimising AIC.
best.logistic.AIC <- bestglm(Xy = predictors.for.bestglm, family = binomial, IC = "AIC",
method = "exhaustive")
## Show the top five models in terms of minimising AIC.
panderOptions("table.continues", "")
pander(best.logistic.AIC$BestModels, caption = "Variables selected in the top five models
according to minimising AIC.")
```

Table 19: Variables selected in the top five models according to minimising AIC. (continued below)

SEX	AGE	EDUC	SMOKER	CIG	BP_MED	STROKE	HYPER	DIAB	CHOL
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE

SBP	DBP	BMI	HR	GLUC	Criterion
TRUE	FALSE	FALSE	FALSE	TRUE	2773
TRUE	FALSE	FALSE	FALSE	TRUE	2774
TRUE	FALSE	FALSE	FALSE	TRUE	2774
TRUE	FALSE	FALSE	FALSE	TRUE	2775

SBP	DBP	BMI	HR	GLUC	Criterion
TRUE	FALSE	FALSE	TRUE	TRUE	2775

```r
# 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")
## Show the top five models in terms of minimising BIC.
pander(best.logistic.BIC$BestModels, caption = "Variables selected in the top five models
according to minimising BIC")
```

Table 21: Variables selected in the top five models according to minimising BIC (continued below)

SEX	AGE	EDUC	SMOKER	CIG	BP_MED	STROKE	HYPER	DIAB	CHOL
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE
TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE

SBP	DBP	BMI	HR	GLUC	Criterion
TRUE	FALSE	FALSE	FALSE	TRUE	2808
TRUE	FALSE	FALSE	FALSE	TRUE	2812
TRUE	FALSE	FALSE	FALSE	TRUE	2813
TRUE	FALSE	FALSE	FALSE	TRUE	2814
TRUE	FALSE	FALSE	FALSE	TRUE	2815

The table below provides a comparison of the “best” subset of predictors produced by an exhaustive subset search and using the criteria of minimising AIC and minimising BIC. For reference, the predictors included in optimal models using forward and backward selection algorithms are also shown. The optimal model based on minimising AIC includes `SEX`, `AGE`, `CIG`, `STROKE`, `HYPER`, `CHOL`, `SBP`, and `GLUC`, whereas the optimal model based on minimising BIC is the same except that it does not include `STROKE`, `HYPER`, or `CHOL`. We know that the only difference between AIC and BIC is in terms of the penalty term with the penalty term for BIC being increasingly larger than that of AIC as the sample size increases. (Here, we have a large sample size, so the penalty for BIC is much larger than that for AIC.) This means that the optimal set of predictors according to the criterion of minimising BIC will be a subset of the predictors chosen according to the criterion of minimising AIC. We note that the optimal model identified by minimising AIC matches the two models selected using stepwise selection algorithms. This is not always the case, as best subset selection does an exhaustive search of all possible combinations of predictors, whereas forward and backward selection do not.

Table 23: Predictors included (✓) in optimal models selected by forward and backward selection algorithms and best subset selection according to the criteria of minimising AIC and minimising BIC.

Variable	Forward selection	Backward selection	AIC	BIC
SEX	✓	✓	✓	✓
AGE	✓	✓	✓	✓
EDUC				
SMOKER				
CIG	✓	✓	✓	✓
BP_MED				
STROKE	✓	✓	✓	
HYPER	✓	✓	✓	
DIAB				
CHOL	✓	✓	✓	
SBP	✓	✓	✓	✓
DBP				
BMI				
HR				
GLUC	✓	✓	✓	✓

c. **(6 marks) Although it would be most appropriate to consider all possible combinations of the 15 predictor variables for a cross-validation routine to select a model based on maximising the accuracy or maximising area under the receiver operating characteristic curve (AUC), it is not feasible to do so on home computers in a reasonable amount of time. Consequently, use the predictors identified by best subset selection according to the criterion of minimising AIC from part (b). (If unable to perform the required subset selection in part (b), make note of that here and use the predictors in the optimal model identified by backward selection in part (a).) For this set of predictors, use 20 repetitions of 10-fold cross-validation to identify the optimal model(s) identified according to the criteria of**

   i. **maximising accuracy and**
   ii. **maximising AUC.**

   If the optimal model(s) identified according to these criteria are different, highlight how they differ, and explain why the criteria of maximising accuracy and maximising AUC may lead to different "best" models. If these models differ from those identified as "best" in parts (a) and (b), explain why this may be the case.

   Code to extract the column numbers for variables identified by best subset selection (according to the criterion of minimising AIC) and produce a matrix where rows represent the possible combinations of these variables is as shown below. This produces in total $2^8 - 1 = 255$ possible models.

```
# Load relevant libraries
library(caret) # Need for train().
```

```
## Loading required package: ggplot2

##
## Attaching package: 'ggplot2'

## The following object is masked from 'package:memisc':
##
##     syms
```

```
library(doParallel) # Need for registerDoParallel().
```

```
## Loading required package: foreach
```

```
##
## Attaching package: 'foreach'
```

```
## The following object is masked from 'package:memisc':
##
##     foreach
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```
library(foreach) # Need for foreach() for parallel computing.

# Specify the indices of the variables to be considered in predictive models for survival
variable.indices <- 1 : (ncol(hd.reduced) - 1)
variable.indices <- variable.indices[as.logical(best.logistic.AIC$BestModels[1,
-ncol(hd.reduced)])]

# 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, ]
```

Code for calculating accuracy and AUC for 20 repetitions of 10-fold cross-validation for these models is as shown below. Point estimates with vertical bars extending one standard error above and below these point estimates are shown to visualise the accuracy and AUC for these repetitions of 10-fold cross-validation across the models, and the predictors contained in models that are within one standard error of the “best” model in terms of maximising accuracy or maximising AUC are also presented.

```
# Set random number generator seed for replicability of results.
set.seed(1)

# Specify the number of folds and repetitions of k-fold cross-validation to use.
folds <- 10
reps <- 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 = reps, seeds =
1 : (folds * reps + 1), classProbs = TRUE, savePredictions = TRUE)

# Save estimated accuracy and standard deviation for each model type and 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(hd.reduced)[variable.indices][all.comb[i,] == 1], collapse = " + "))), data =
hd.reduced, trControl = fitControl, method = "glm", family = "binomial")$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 = reps, seeds =
1 : (folds * reps + 1), summaryFunction = twoClassSummary, classProbs = TRUE,
savePredictions = TRUE)

# Save estimated AUC and standard deviation for each model type and set of covariates,
using untransformed durations.
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(hd.reduced)[variable.indices][all.comb[i,] == 1], collapse = " + "))), data =
hd.reduced, 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.
names(hd.reduced)[variable.indices[all.comb[which.max(accuracy[, 2]), ] == 1]]
```

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

```
max(accuracy[, 2])
```

```
## [1] 0.8536234
```

```
# Determine all models within one SE of the best model.
best.models.accuracy <- (1 : nrow(all.comb))[accuracy[, 2] + accuracy[, 3] >=
max(accuracy[, 2])]

# Extract information on the simplest models that are within one SE of the best model.
simplest.equiv.models.accuracy <-
best.models.accuracy[apply(all.comb[best.models.accuracy, ], 1, sum) ==
min(apply(all.comb[best.models.accuracy, ], 1, sum))]

for(i in 1 : length(simplest.equiv.models.accuracy))
{
cat(paste("Model ", i, ":\n"))
print(names(hd.reduced)[variable.indices[all.comb[simplest.equiv.models.accuracy[i], ] ==
1]]) # Variable names
print(accuracy[simplest.equiv.models.accuracy[i], 2]) # Accuracy

cat("\n")
}
```

```
## Model  1 :
## [1] "SBP"  "GLUC"
##  Accuracy
## 0.8485794
```

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

# View the model that maximises AUC
names(hd.reduced)[variable.indices[all.comb[which.max(AUC[, 2]), ] == 1]]
```

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

```
max(AUC[, 2])
```

```
## [1] 0.733951
```

```
# Determine all models within one SE of the best model.
best.models.AUC <- (1 : nrow(all.comb))[AUC[, 2] + AUC[, 3] >= max(AUC[, 2])]

# Extract information on the simplest models that are within one SE of the best model.
simplest.equiv.models.AUC <- best.models.AUC[apply(all.comb[best.models.AUC, ], 1, sum)
== min(apply(all.comb[best.models.AUC, ], 1, sum))]

for(i in 1 : length(simplest.equiv.models.AUC))
{
cat(paste("Model ", i, ":\n"))
print(names(hd.reduced)[variable.indices[all.comb[simplest.equiv.models.AUC[i], ] == 1]])
# Variable names
print(AUC[best.models.AUC[i], 2]) # AUC

cat("\n")
}
```

```
## Model  1 :
## [1] "SEX" "AGE"
##       ROC
## 0.7006807
##
## Model  2 :
## [1] "AGE" "CIG"
##       ROC
## 0.6979719
```

```
##
## Model  3 :
## [1] "AGE"   "HYPER"
##       ROC
## 0.7051985
##
## Model  4 :
## [1] "AGE" "SBP"
##       ROC
## 0.7011235
```

As before, we present a table with the predictors included in the best model according to the criteria of maximising accuracy and maximising AUC, and we show the optimal models identified by forward and backward selection as well as best subset selection using AIC and BIC for comparison.

It is not surprising that the optimal models selected according to the criteria of maximising accuracy and maximising AUC are different. Accuracy ignores the different types of errors that occur and lumps them all together, whereas AUC accounts for both Type I and Type II errors. An imbalance in design can easily mean that a particular model can have a high accuracy but be sub-optimal in terms of Type I vs. Type II error rates.

Here, we note that the optimal models according to maximising accuracy and maximising AUC are simpler than those identified by stepwise selection algorithms and best subset selection where the criterion is minimising AIC. It is important to recall that, for these other methods, there is no distinction between the training and test sets (*i.e.*, the same data used to fit the model is used to assess model performance), and, even though there is a penalty for additional model complexity in the case of AIC, this does not guarantee that the approach is free of the problem of overfitting. Indeed, the fact that forward and backward selection as well as best subset selection using AIC all select an optimal model with more predictors than those based on a cross-validation routine where there is a clear distinction between training data and test data suggests that these methods have all overfit the data.

Variable	Forward selection	Backward selection	AIC	BIC	Accuracy	AUC
`SEX`	✓	✓	✓	✓		
`AGE`	✓	✓	✓	✓		✓
`EDUC`						
`SMOKER`						
`CIG`	✓	✓	✓	✓		
`BP_MED`						
`STROKE`	✓	✓	✓			
`HYPER`	✓	✓	✓			✓
`DIAB`						
`CHOL`	✓	✓	✓			
`SBP`	✓	✓	✓	✓	✓	
`DBP`						
`BMI`						
`HR`						
`GLUC`	✓	✓	✓	✓	✓	

Assignment total: 50 marks

References

Boston University and the National Heart, Lung, & Blood Institute. 2020. “The Framingham Heart Study.” https://framinghamheartstudy.org/.

Madell, R., and K. Cherney. 2018. “Blood Pressure Readings Explained.” *Healthline*. https://www.healthline.com/health/high-blood-pressure-hypertension/blood-pressure-reading-explained.

Nichols, G. A., T. J. Bell, K. L. Pedula, and M. O’Keeffe-Rosetti. 2010. “Medical Care Costs Among Patients with Established Cardiovascular Disease.” *The American Journal of Managed Care* 16 (3): e86–93.

World Health Organization. 2018. “The Top 10 Causes of Death.” https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death.

Wu, C.-Y., H.-Y. Hu, Y.-J. Chou, N. Huang, Y.-C. Chou, and C.-P. Li. 2015. “High Blood Pressure and All-Cause and Cardiovascular Disease Mortalities in Community-Dwelling Older Adults.” *Medicine* 94 (47): e2160. https://doi.org/10.1097/MD.0000000000002160.