# APPENDIX

# Preparation

library(here)

## here() starts at /Users/amalinaismail/Desktop/DrPH/ SEM 2/ADVANCED CATEGORICAL/ASSIGNMENT

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(haven)  
library(gtsummary)  
library(VGAM)

## Loading required package: stats4  
## Loading required package: splines

library(nnet)  
library(broom)  
library(knitr)  
library(kableExtra)

##   
## Attaching package: 'kableExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## group\_rows

library(tibble)  
library(purrr)  
library(gt)  
library(ggplot2)  
library(ggeffects)  
library(reshape2)

##   
## Attaching package: 'reshape2'  
##   
## The following object is masked from 'package:tidyr':  
##   
## smiths

library(data.table)

##   
## Attaching package: 'data.table'  
##   
## The following objects are masked from 'package:reshape2':  
##   
## dcast, melt  
##   
## The following objects are masked from 'package:lubridate':  
##   
## hour, isoweek, mday, minute, month, quarter, second, wday, week,  
## yday, year  
##   
## The following objects are masked from 'package:dplyr':  
##   
## between, first, last  
##   
## The following object is masked from 'package:purrr':  
##   
## transpose

library(ordinal)

##   
## Attaching package: 'ordinal'  
##   
## The following objects are masked from 'package:VGAM':  
##   
## dgumbel, dlgamma, pgumbel, plgamma, qgumbel, rgumbel, wine  
##   
## The following object is masked from 'package:dplyr':  
##   
## slice

library(foreign)  
library(brant)

# Methods

## Read data

dat.o <- read\_csv('hpt.csv')

## Rows: 200 Columns: 8  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (8): id, age, height, weight, cholesterol, smoke, exercise, bp  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

summary(dat.o)

glimpse(dat.o)

## Convert Variable

dat.o <-   
 dat.o %>%   
 mutate(across(where(is.labelled), ~as\_factor(.x)))  
summary(dat.o)

## Cut and label variable

dat.o <- dat.o %>%  
 mutate(bp4 = cut(bp,   
 breaks = c(-Inf, 120, 140, 160, Inf),  
 labels = c('=<120','121-140','141-160','>160'),  
 right = TRUE))

## Check grouping

dat.o %>%   
 select(bp4, bp) %>%  
 group\_by(bp4) %>%   
 summarize\_at(vars(bp),   
 c(min = min, max = max))

## Reverse order

lev <- c('>160','141-160','121-140','=<120')  
lev

dat.o <- dat.o %>%   
 mutate(bp4a = fct\_relevel(bp4, lev)) %>%  
 mutate(bp4a = ordered(bp4a, levels = lev))

check

str(dat.o$bp4a)  
levels(dat.o$bp4)  
table(dat.o$bp4)  
levels(dat.o$bp4a)  
table(dat.o$bp4a)

## Exploratory Data Analysis

# Histogram of age by blood pressure category

p1 <- ggplot(dat.o, aes(x = age, fill = bp4)) +

geom\_histogram(binwidth = 5, position = "dodge", alpha = 0.7) +

facet\_wrap(~ bp4, scales = "free\_y") +

labs(title = "Distribution of Age by Blood Pressure Category",

x = "Age",

y = "Count",

fill = "Blood Pressure Category") +

theme\_minimal()

# Histogram of height by blood pressure category

p2 <- ggplot(dat.o, aes(x = height, fill = bp4)) +

geom\_histogram(binwidth = 5, position = "dodge", alpha = 0.7) +

facet\_wrap(~ bp4, scales = "free\_y") +

labs(title = "Distribution of Height by Blood Pressure Category",

x = "Height",

y = "Count",

fill = "Blood Pressure Category") +

theme\_minimal()

# Histogram of weight by blood pressure category

p3 <- ggplot(dat.o, aes(x = weight, fill = bp4)) +

geom\_histogram(binwidth = 5, position = "dodge", alpha = 0.7) +

facet\_wrap(~ bp4, scales = "free\_y") +

labs(title = "Distribution of Weight by Blood Pressure Category",

x = "Weight",

y = "Count",

fill = "Blood Pressure Category") +

theme\_minimal()

# Histogram of cholesterol levels by blood pressure category

p4 <- ggplot(dat.o, aes(x = cholesterol, fill = bp4)) +

geom\_histogram(binwidth = 10, position = "dodge", alpha = 0.7) +

facet\_wrap(~ bp4, scales = "free\_y") +

labs(title = "Distribution of Cholesterol Levels by Blood Pressure Category",

x = "Cholesterol Levels",

y = "Count",

fill = "Blood Pressure Category") +

theme\_minimal()

# Bar plot of smoking status by blood pressure category

p5 <- ggplot(dat.o, aes(x = bp4, fill = smoke)) +

geom\_bar(position = "dodge", alpha = 0.7) +

labs(title = "Distribution of Smoking Status by Blood Pressure Category",

x = "Blood Pressure Category",

y = "Count",

fill = "Smoking Status") +

theme\_minimal()

# Bar plot of exercise frequency by blood pressure category

p6 <- ggplot(dat.o, aes(x = bp4, fill = exercise)) +

geom\_bar(position = "dodge", alpha = 0.7) +

labs(title = "Distribution of Exercise Frequency by Blood Pressure Category",

x = "Blood Pressure Category",

y = "Count",

fill = "Exercise Frequency") +

theme\_minimal()

# Combine all plots into a single visualization

combined\_plot <- (p1 + p2) / (p3 + p4) / (p5 + p6)

# Print the combined plot

print(combined\_plot)

dat.o <- dat.o %>%

mutate(smoke = as.factor(smoke))

summary\_table <- dat.o %>%

select(bp4, age, height, weight, cholesterol, smoke, exercise) %>%

tbl\_summary(

by = bp4,

statistic = list(

all\_continuous() ~ "{mean} ({sd})",

all\_categorical() ~ "{n} ({p}%)"

),

type = list(

where(is.logical) ~ "categorical"

),

label = list(

age ~ "age",

height ~ "height (cm)",

weight ~ "weight (kg)",

cholesterol ~ "cholesterol (mmol/L)",

smoke ~ "smoke",

exercise ~ "exercise"

),

missing\_text = "Missing"

) %>%

modify\_caption("\*\*Table 1. Participant Characteristics by Blood Pressure Group\*\*") %>%

modify\_header(label ~ "\*\*Variable\*\*") %>%

modify\_spanning\_header(

starts\_with("stat\_") ~ "\*\*Blood Pressure Group\*\*"

) %>%

modify\_footnote(all\_stat\_cols() ~ "Mean (SD) or Frequency (%)") %>%

bold\_labels() %>%

as\_gt()

# Print the summary table

print(summary\_table)

## Estimation

### Adjacent-category model or multinomial or baseline logit model

Generate new variable without ordering

dat.o <-   
 dat.o %>%   
 mutate(bp4b = fct\_relevel(bp4, lev))

mlogit1 <-   
 vglm(bp4 ~ exercise, multinomial, data = dat.o)  
summary(mlogit1)

RRR

exp(coef(mlogit1))

### Continuation-ratio

1. Compare **bp4 == >160** vs. **bp4 == 141-160**

table(dat.o$bp4) ; table(dat.o$bp4a)

dat.oa <-   
 dat.o %>%   
 filter(bp4 == '>160' | bp4 == '141-160')  
cr1 <- glm(bp4a ~ exercise, family = binomial(link ='logit'),  
 data = dat.oa)  
summary(cr1)

1. Compare **bp4 == >160 and bp4 == 141-160** vs. **bp4 == 121-140**

dat.ob <- dat.o %>%   
 filter(bp4 == '>160' |   
 bp4 == '141-160'|   
 bp4 == '121-140')  
table(dat.ob$bp4a)

dat.ob <- dat.ob %>%   
 mutate(bp4b = ifelse(bp4a == ">160", 0,   
 ifelse(bp4a == "141-160",0,1)))  
table(dat.ob$bp4a) ; table(dat.ob$bp4b)

And run the next CR model:

cr2 <-   
 glm(bp4b ~ exercise, family = binomial(link ='logit'),   
 data = dat.ob)  
summary(cr2)

1. Compare **bp4 == >160 and bp4 == 141-160 and bp4 == 121-140** vs. **bp4 == =<120**

table(dat.o$bp4) ; table(dat.o$bp4a)

dat.ob <-   
 dat.o %>%   
 mutate(bp4c = fct\_recode(bp4a,   
 gp0 = '>160',  
 gp0 = '141-160',  
 gp0 = '141-160',  
 gp0 = '121-140'))

table(dat.ob$bp4a); table(dat.ob$bp4b) ; table(dat.ob$bp4c)

And we can obtain logit 3

cr3 <-   
 glm(bp4c ~ exercise, family = binomial(link = 'logit'),  
 data = dat.ob)  
summary(cr3)

### Cumulative link logit model

mean\_bp\_by\_category <- dat.o %>%  
 group\_by(bp4) %>%  
 summarize(mean\_bp = mean(bp, na.rm = TRUE))  
  
# Display the result  
print(mean\_bp\_by\_category)

levels(dat.o$bp4)

* o.age <- clm(bp4 ~ age, data=dat.o)  
  summary(o.age)

log odds and odds ratio

* tidy(o.age, conf.int = TRUE)
* tidy(o.age, exponentiate = TRUE ,conf.int = TRUE)

## Prediction

### Predicted probability

**Use polr** package

polr\_cr3 <- MASS::polr(bp4 ~ age, data = dat.o, Hess = TRUE)  
summary(polr\_cr3)

prob\_polr <- predict(polr\_cr3, type = 'probs')  
head(prob\_polr) ; tail(prob\_polr)

### Manual calculation for prediction

summary(o.age)

### New data

newData <- expand.grid(age = unique(dat.o$age))  
head(newData)

lp.o.age <- predict(o.age, newdata = newData, type = 'linear.predictor')  
lp.o.age

### Coefficients

coef.o.age<- coef(o.age)  
coef.o.age

age\_value <- 32  
lp.o1.bx <- coef.o.age['age'] \* age\_value

### Probabilities

pMat <- cbind(  
 p1 = pLeq1, # Probability of being in the first category  
 p2 = pLeq2 - pLeq1, # Probability of being in the second category  
 p3 = pLeq3 - pLeq2, # Probability of being in the third category  
 p4 = 1 - pLeq3 # Probability of being in the fourth category  
)  
  
pMat

### Confirm the prediction

predict(o.age, newdata = newData, type = 'prob')

### Plot

# Create a new data frame with age, smoke, and predicted probabilities

newdat <- cbind(dat.o[, c("age", "smoke","exercise")], prob\_polr)

# Reshape the data to long format

lnewdat <- melt(newdat, id.vars = c("age", "smoke", "exercise"),

variable.name = "BP\_Level", value.name = "Probability")

# View the first few rows

head(lnewdat)

ggplot(lnewdat, aes(x = age, y = Probability, colour = BP\_Level)) +

geom\_line() + facet\_grid(smoke ~ exercise, labeller="label\_both")

### Checking proportional odds assumption

brant(polr\_cr3)