Problem Set 5

Tate Mason

Question 1

Part A

Let's first make our sample

```
rm(list = ls())

library(haven)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```
# load data
data <- read_dta('~/SchoolWork/Sem2/Metrics/PSets/PS5/cps09mar.dta')

male_data <- data %>% filter(female == 0)

X <- male_data %>%
  transmute(
    intercept = 1,
```

```r
    age = age,
    education = education,
    black = as.numeric(race == 2),
    hispanic = as.numeric(hisp == 1)
  ) %>%
  as.matrix()
y <- male_data$union
```

Now, we can conduct analysis using the glm function to establish a baseline of estimates.

```r
probit_glm <- glm(union ~ age + education + I(race == 2) + I(hisp == 1),
              data = male_data,
              family = binomial(link = "probit"))
summary(probit_glm)
```

```
Call:
glm(formula = union ~ age + education + I(race == 2) + I(hisp ==
    1), family = binomial(link = "probit"), data = male_data)

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -1.953818   0.107176 -18.230  < 2e-16 ***
age               0.007925   0.001419   5.584 2.35e-08 ***
education        -0.025505   0.006221  -4.100 4.14e-05 ***
I(race == 2)TRUE -0.054083   0.060004  -0.901    0.367
I(hisp == 1)TRUE -0.297745   0.056865  -5.236 1.64e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6282.0  on 29139  degrees of freedom
Residual deviance: 6210.7  on 29135  degrees of freedom
AIC: 6220.7

Number of Fisher Scoring iterations: 6
```

Now, implementing our probit:

```r
dnorm_vec <- function(x) dnorm(x)
pnorm_vec <- function(x) pnorm(x)

probit_loglik <- function(beta, X, y) {
  xb <- X %*% beta
  sum(y*log(pnorm_vec(xb) + 1e-10))
}

probit_neglik <- function(beta, X, y) {
  -probit_loglik(beta, X, y)
}

probit_grad <- function(beta, X, y) {
  xb <- X %*% beta
  phi_xb <- dnorm_vec(xb)
  Phi_xb <- pnorm_vec(xb)

  lambda <- ifelse(y == 1,
                   phi_xb/(Phi_xb+1e-10),
                   -phi_xb/(1-phi_xb+1e1-0))
  -colSums(lambda*X)
}

init_beta <- rep(0, ncol(X))
probit_opt <- optim(init_beta,
                    fn = probit_neglik,
                    gr = probit_grad,
                    X = X,
                    y = y,
                    method = "BFGS",
                    hessian = TRUE)

beta_hat <- probit_opt$par
hessian <- probit_opt$hessian
var_beta <- solve(hessian)
se_beta <- sqrt(diag(var_beta))

probit_results <- data.frame(
  Variable = c("Intercept", "Age", "Education", "Black", "Hispanic"),
  Coefficient = beta_hat,
  Std.Error = se_beta
)
```

```r
print(probit_results)
```

```
   Variable   Coefficient   Std.Error
1 Intercept -1.470143e-17 0.366392418
2       Age -5.798083e-16 0.004542339
3 Education -2.062481e-16 0.021723204
4     Black -1.248879e-18 0.180365013
5  Hispanic -3.270044e-18 0.174865466
```

Comparative analysis of our model vs the built in one.

```r
compare <- data.frame(
  Variable = c("Intercept", "Age", "Education", "Black", "Hispanic"),
  Manual_Coef = beta_hat,
  GLM_Coef = coef(probit_glm),
  Manual_SE = se_beta,
  GLM_SE = summary(probit_glm)$coefficients[,2]
)

print(compare)
```

```
                  Variable   Manual_Coef     GLM_Coef   Manual_SE      GLM_SE
(Intercept)      Intercept -1.470143e-17 -1.953818176 0.366392418 0.107176068
age                    Age -5.798083e-16  0.007925402 0.004542339 0.001419363
education        Education -2.062481e-16 -0.025504590 0.021723204 0.006221006
I(race == 2)TRUE     Black -1.248879e-18 -0.054082971 0.180365013 0.060004104
I(hisp == 1)TRUE  Hispanic -3.270044e-18 -0.297744914 0.174865466 0.056864898
```

Part B

```r
calculate_ape <- function(beta, X) {
  xb <- X %*% beta
  phi_xb <- dnorm(xb)

  apes <- numeric(length(beta) - 1)

  for (j in 2:length(beta)) {
    apes[j-1] <- mean(phi_xb*beta[j])
```

```r
  }
  return(apes)
}

apes <- calculate_ape(beta_hat, X)

ape_results <- data.frame(
  Variable = c("Age", "Education", "Black", "Hispanic"),
  APE = apes
)

print(ape_results)
```

```
   Variable           APE
1       Age -2.313101e-16
2 Education -8.228107e-17
3     Black -4.982306e-19
4  Hispanic -1.304559e-18
```

Parts C and D

```r
calculate_ape_se <- function(beta, X, var_beta) {
  xb <- X %*% beta
  phi_xb <- dnorm(xb)
  n <- nrow(X)
  p <- ncol(X)

  ape_se <- numeric(p-1)

  for (j in 2:p) {
    grad <- numeric(p)
    grad[j] <- mean(phi_xb)

    phi_prime_xb <- -xb * phi_xb
    for (k in 1:p) {
      grad[k] <- grad[k] + mean(X[,k]*phi_prime_xb*beta[j])
    }

    ape_se[j-1] <- sqrt(t(grad) %*% var_beta %*% grad)
  }
```

```r
  return(ape_se)
}

ape_se <- calculate_ape_se(beta_hat, X, var(var_beta))

ape_results$Analytical_SE <- ape_se
print(ape_results)
```

```
   Variable           APE Analytical_SE
1       Age -2.313101e-16  0.0001702294
2 Education -8.228107e-17  0.0012183082
3     Black -4.982306e-19  0.0058898661
4  Hispanic -1.304559e-18  0.0065029696
```

Part E

```r
bootstrap_ape_se <- function(X, y, beta_init, B = 100) {
  n <- nrow(X)
  p <- ncol(X)

  bootstrap_apes <- matrix(0, nrow = B, ncol = p-1)
  probit_neglik <- function(beta, X, y) {
    xb <- X %*% beta
    p <- pnorm(xb)
    -sum(y*log(p + 1e-10) + (1-y)*log(1-p+1e-10))
  }

  for (b in 1:B) {
    boot_indices <- sample(1:n, n, replace=TRUE)
    X_boot <- X[boot_indices, ]
    y_boot <- y[boot_indices[]]

    tryCatch({
      boot_opt <- optim(
        par = beta_init,
        fn = probit_neglik,
        X = X_boot,
        y = y_boot,
        method = "Nelder-Mead",
```

```r
        control = list(maxit = 2000)
      )
      boot_beta <- boot_opt$par

      if (all(abs(boot_beta) < 1e-8)) {
        boot_beta <- beta_init
      }

      xb_boot <- X_boot %*% boot_beta
      phi_xb_boot <- dnorm(xb_boot)

      for (j in 2:p) {
        bootstrap_apes[b, j-1] <- mean(phi_xb_boot*boot_beta[j])
        }
      }, error = function(e) {
        cat("Bootstrap iteration", b, "failed with error:", e$message, "\n")
        bootstrap_apes[b, ] <- NA
    })
  }
  bootstrap_se <- apply(bootstrap_apes, 2, sd)
  return(bootstrap_se)
}

set.seed(19)

bs_ape_se <- bootstrap_ape_se(X, y, beta_hat, B = 100)

ape_results <- data.frame(
  Variable = c("Age", "Education", "Black", "Hispanic"),
  APE = calculate_ape(beta_hat, X),
  Analytical_SE = ape_se,
  Bootstrap_SE = bs_ape_se
)

print(ape_results)
```

```
   Variable           APE Analytical_SE Bootstrap_SE
1       Age -2.313101e-16  0.0001702294            0
2 Education -8.228107e-17  0.0012183082            0
3     Black -4.982306e-19  0.0058898661            0
4  Hispanic -1.304559e-18  0.0065029696            0
```