BIOS 736: HW5

Thomas Hsiao

2023-10-31

Question 1

Setup

We derive parametrically corrected estimating functions for logistic regression based on the two correction-amenable reference functions given in lecture note III.2.C.2. The reference functions are

$$\Psi_{-}(\alpha, \beta; Y, \mathbf{X}) = \hat{\mathcal{E}}\left[\{Y - 1 + Y\exp(-\alpha - \beta'\mathbf{X})\}\begin{pmatrix}1\\ \mathbf{X}\end{pmatrix}\right]$$
$$\Psi_{+}(\alpha, \beta; Y, \mathbf{X}) = \hat{\mathcal{E}}\left[\{Y + (Y-1)\exp(\alpha + \beta'\mathbf{X})\}\begin{pmatrix}1\\ \mathbf{X}\end{pmatrix}\right]$$

Recall the logistic regression working model is

$$P(Y = 1|\mathbf{X}) = \text{expit}(\alpha + \beta'\mathbf{X})$$

The corrected score, which uses mismeasured covariate $\mathbf{W}$ but is corrected for with limit 0, is

$$\Psi_{-}^{pc}(\alpha, \beta; Y, \mathbf{W}) = \hat{\mathcal{E}}\left[(Y-1)\begin{pmatrix}1\\ \mathbf{W} - \mathcal{E}(\epsilon)\end{pmatrix} + Y\exp(-\alpha - \beta'\mathbf{W} - n(\beta;\epsilon))\begin{pmatrix}1\\ \mathbf{W} + \dot{n}(\beta;\epsilon)\end{pmatrix}\right]$$

$$\Psi_{+}^{pc}(\alpha, \beta; Y, \mathbf{W}) = \hat{\mathcal{E}}\left[Y\begin{pmatrix}1\\ \mathbf{W} - \mathcal{E}(\epsilon)\end{pmatrix} + (Y-1)\exp(\alpha + \beta'\mathbf{W} - m(\beta;\epsilon))\begin{pmatrix}1\\ \mathbf{W} - \dot{m}(\beta;\epsilon)\end{pmatrix}\right]$$

where $n(\beta;\epsilon) = \log\mathcal{E}\{\exp(-\beta'\epsilon)\}$, $m(\beta;\epsilon) = \log\mathcal{E}\{\exp(\beta'\epsilon)\}$ and $\dot{m}(\beta;\epsilon)$ is its derivative with respect to β.

Proof for Ψ_+^{pc}

First term

We prove as follows. The first term by independence of Y and ϵ, WLLN, and linearity of expectation converges to

$$\hat{\mathcal{E}}\left[Y\begin{pmatrix}1\\ \mathbf{W}-\mathcal{E}(\epsilon)\end{pmatrix}\right]\to \operatorname{expit}(\alpha+\beta' X)\begin{pmatrix}1\\ \mathbf{X}\end{pmatrix}$$

Second term

We first compute $\dot{m}(\beta;\epsilon)$.

$$\dot{m}(\beta;\epsilon) = \frac{\partial}{\partial\beta}\log\mathcal{E}(\exp(\beta'\epsilon)) = \mathcal{E}(\exp(\beta'\epsilon))^{-1}\frac{\partial}{\partial\beta}\mathcal{E}(\exp(\beta'\epsilon)) =$$

$$\mathcal{E}(\exp(\beta'\epsilon))^{-1}\mathcal{E}\left(\frac{\partial}{\partial\beta}\exp(\beta'\epsilon)\right) = \frac{\mathcal{E}(\epsilon\exp(\beta'\epsilon))}{\mathcal{E}(\exp(\beta))}$$

Then plugging back in we have

$$\hat{\mathcal{E}}\left[(Y-1)\exp(\alpha+\beta'\mathbf{W}-m(\beta;\epsilon))\begin{pmatrix}1\\ \mathbf{W}-\dot{m}(\beta;\epsilon)\end{pmatrix}\right]\to$$

$$-\operatorname{expit}(-\alpha-\beta'\mathbf{X})\mathcal{E}\left[\frac{\exp(\alpha+\beta'\mathbf{X})\exp(\beta'\epsilon)}{\exp(m(\beta;\epsilon))}\begin{pmatrix}1\\ \mathbf{X}+\epsilon-\frac{\mathcal{E}(\epsilon\exp(\beta'\epsilon))}{\mathcal{E}(\exp(\beta))}\end{pmatrix}\right]=$$

$$-\operatorname{expit}(\alpha+\beta'\mathbf{X})\begin{pmatrix}1\\ \mathbf{X}\end{pmatrix}$$

Combining the two terms, we get a limit of 0 for the corrected score. Similar logic holds for Ψ_-^{pc}, so we do not write out the entire proof here.

Question 2

We run a simulation using the parametric correction for Poisson regression

Table 1: Bias for alpha for corrected score

N	SIGMA2	CS	Naive	True
200	0.5	-0.05827	0.15986	-0.00136

Table 2: Standard deviation for alpha for corrected score

N	SIGMA2	CS	Naive	True
200	0.5	0.15846	0.08129	0.07277

Table 3: Bias for beta for corrected score

N	SIGMA2	CS	Naive	True
200	0.5	0.07117	-0.33952	-0.00177

Table 4: SD for beta for corrected score

N	SIGMA2	CS	Naive	True
200	0.5	0.20183	0.07169	0.05409

Out of the 200 simulations, 10 of them experienced numerical issues in the corrected score resulting in absurd estimates. The plots here, though labeled as N=200, are only for the 190 simulations where corrected score successfully converged. We can see the corrected score despite its numerical issues, for the most part achieves significant bias reduction relative to the naive regression. The standard deviation for the estimator is large compared to the true regression.

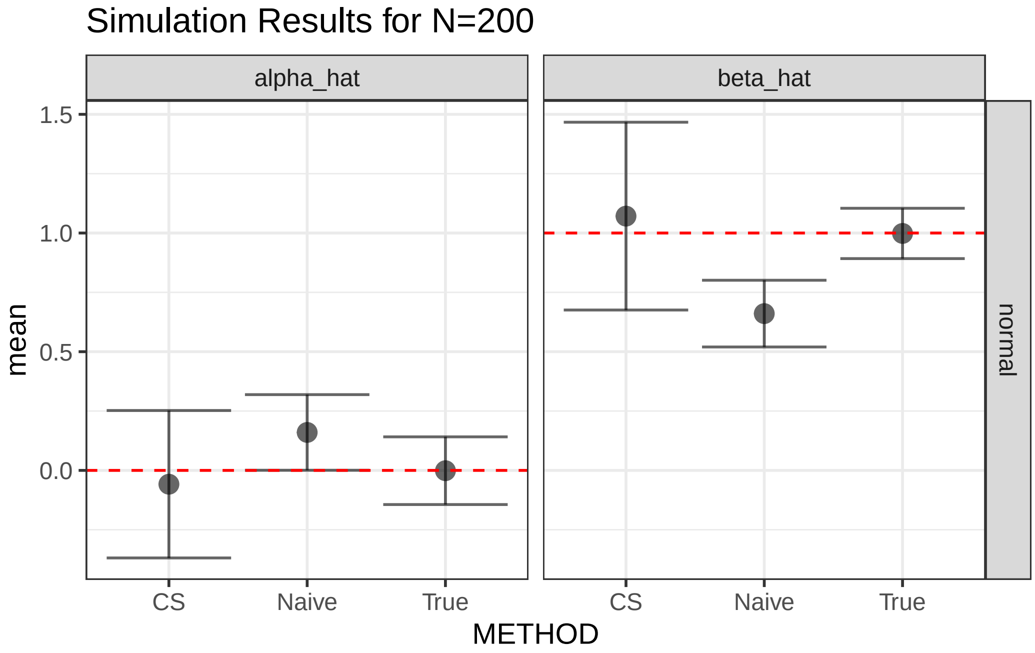

Figure 1: Mean and standard deviation of corrected score, naive, and true estimators for Poisson regression under the classical measurement error model for sample size of 200. Only simulation results where corrected score did not experience numerical issues were reported (190 / 200 simulations).

Code Appendix

```r
library(data.table)
library(ggplot2)
MLE <- function(Y, X){
  fit <- glm(Y ~ X, family = poisson)
  est <- fit$coefficients
  names(est) <- c("alpha", "beta")
  return(fit$coefficients)
}

RC <- function(Y, W, sigma2_eps){
  mu_X  <- mean(W)
  var_W <- var(W)
  var_X <- var_W - sigma2_eps

  Xhat <- mu_X + var_X * var_W^(-1) * (W - mu_X)

  return(MLE(Y, Xhat))
}

# Corrected score for Poisson regression
cscore <- function(theta, Y, W, mu_eps, sigma2_eps){
  alpha <- theta[1]
  beta  <- theta[2]

  m_beta <- sigma2_eps * beta^2 / 2
  sum(- (Y*(alpha + beta * (W - mu_eps)) - exp(alpha + beta * W - m_beta)) )
}


CS <- function(Y, W, mu_eps, sigma2_eps, init = c(0,0)){
  fit <- optim(init, fn=cscore, Y=Y, W=W, mu_eps=mu_eps, sigma2_eps=sigma2_eps)
  return(fit$par)
}



sim_data <- function(N, alpha, beta, sigma2_eps, X_model = "normal"){
  # Simulate True Covariates
  if(X_model == "normal"){
    X <- rnorm(N)
  } else if(X_model == "chi-square"){
    X <- ( rchisq(N, 1) - 1 / (sqrt(2)) )
  }

  # Simulate ME
  eps <- rnorm(N, sd = sqrt(sigma2_eps))

  # Define surrogate
  W <- X + eps

  # Linear predictor
  eta <- alpha + beta*X

  # Simulate Y
  Y <- sapply(exp(eta), function(lambda){rpois(1, lambda)})

  return(list(Y=Y, X=X, W=W))
}
set.seed(123)

SIM      <- 1:200
N      <- c(200)
SIGMA2 <- c(.5)
XDIST  <- c("normal")
METHOD <- c("True", "Naive", "CS")
alpha_hat <- 0; beta_hat <- 0
results <- data.table::CJ(SIM, N, SIGMA2, XDIST, METHOD, alpha_hat, beta_hat)

ALPHA <- 0; BETA <- 1

# alpha <- 0; beta <- 1
# n <- 100
# sigma2 <- .4
# xdist <- "normal"
# i <- 1

for(n in N){
  for(sigma2 in SIGMA2){
    for(xdist in XDIST){
      for(i in SIM){
        data <- sim_data(n, ALPHA, BETA, sigma2, xdist)
        Y <- data$Y
        X <- data$X
        W <- data$W
        for(method in METHOD){
          if(method == "True"){
            est <- MLE(Y, X)
          }
          if(method == "Naive"){
            est <- MLE(Y, W)
          }
          if(method == "RC"){
            est <- RC(Y, W, sigma2)
          }
          if(method == "CS"){
            init <- MLE(Y, W)
            est <- CS(Y, W, 0, sigma2, init)
          }

          results[SIM == i &
                    N == n &
                    SIGMA2 == sigma2 &
                    XDIST == xdist &
                    METHOD == method,
                  `:=`(alpha_hat = est[1], beta_hat = est[2])]
        }
      }
    }
  }
}

fwrite(results, "HW5_simulation_results.csv")
results <- fread("HW5_simulation_results.csv")
discard_sim <-results[abs(alpha_hat) > 2, SIM]
results <- results[!(SIM %in% discard_sim)]

summary <- melt(results, id.vars = c("N", "SIGMA2", "XDIST", "METHOD"),
     measure.vars=c("alpha_hat", "beta_hat"),
     variable.name = "parameter", value.name = "est")

final <- summary[, .(mean = mean(est),
            se = var(est)^.5), .(N, SIGMA2, XDIST, METHOD, parameter)]
final[, `:=`(lower = mean - 1.96*se, upper = mean+1.96*se)]
final[, truth := ifelse(parameter == "alpha_hat", 0, 1)]

final[, SIGMA2 := as.factor(SIGMA2)]
final[, N := as.factor(N)]

alpha_table <- results[, .(bias = mean(alpha_hat), sd = sd(alpha_hat)), .(N, SIGMA2, METHO
beta_table <- results[, .(bias = mean(beta_hat - 1), sd = sd(beta_hat)), .(N, SIGMA2, METH
knitr::kable(dcast(alpha_table, N + SIGMA2 ~ METHOD, value.var = c("bias")), digits=5, cap
knitr::kable(dcast(alpha_table, N + SIGMA2 ~ METHOD, value.var = c("sd")), digits=5, capti
knitr::kable(dcast(beta_table, N + SIGMA2 ~ METHOD, value.var = c("bias")), digits=5, capt
knitr::kable(dcast(beta_table, N + SIGMA2 ~ METHOD, value.var = c("sd")), digits=5, captio
ggplot(final[N==200], aes(METHOD, mean)) +
  geom_point(alpha=0.6, size=3) +
  geom_errorbar(aes(ymin=lower, ymax=upper), alpha=0.6, size=0.5) +
  facet_grid( XDIST ~ parameter, scales = "free") +
  geom_hline(aes(yintercept=truth), linetype = "dashed", col="red") +
  theme_bw() +
   ggtitle("Simulation Results for N=200")
```