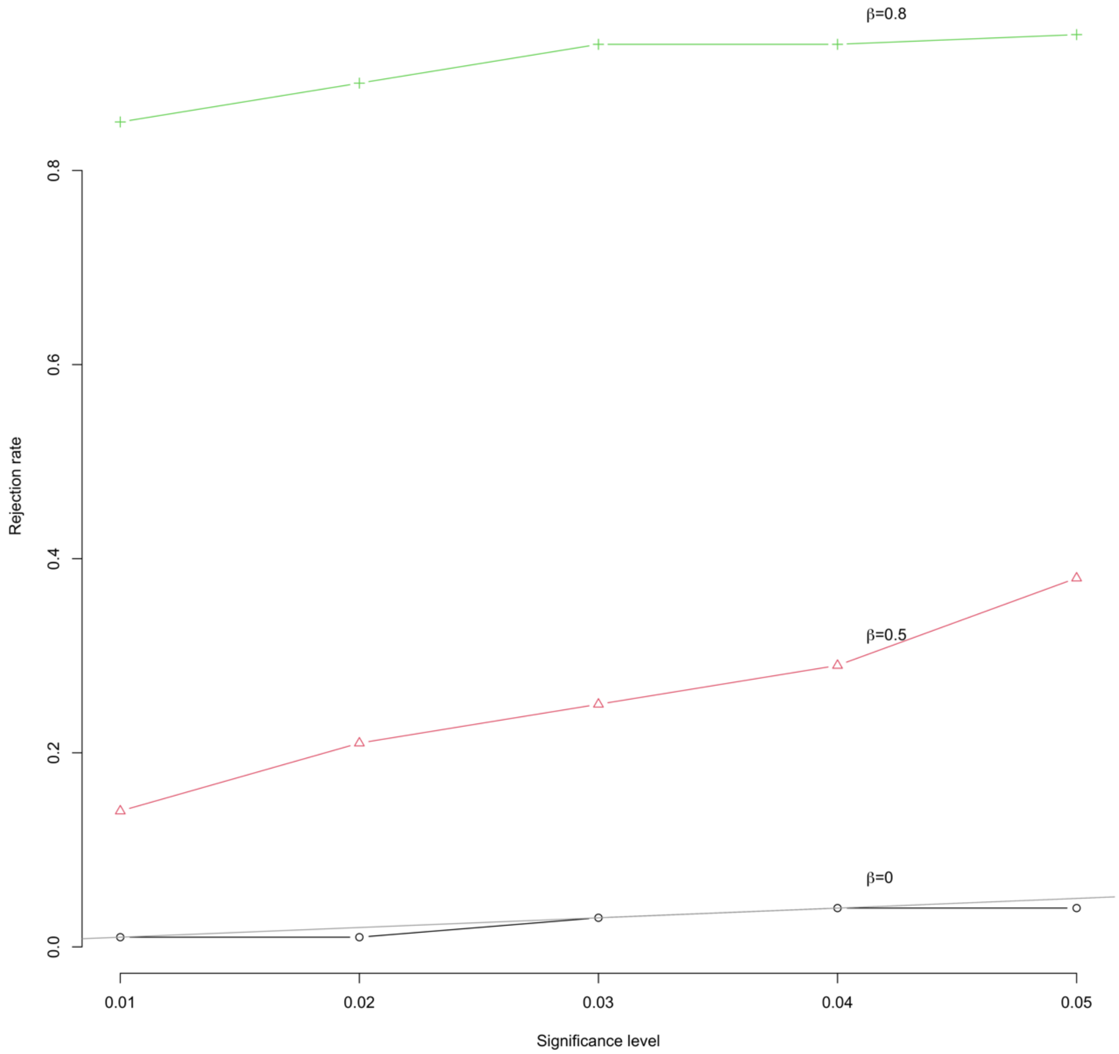
流行病學與生物統計計算 Homework 12
學號：b07401048 系級：醫學五 姓名：賴柏瑞

# Homework12

#EX 23-1

```r
beta <- c(0, 0.5, 0.8)
sig <- seq(0.01, 0.05, by = 0.01) #significant level
n <- 1000
norep <- 100
y <- c()
mle <- matrix(NA, norep, 3)
rejrate <- matrix(NA, length(beta), length(sig))
for (betaloop in 1 : length(beta)) {
   pvalue <- c()
   for (i in 1 : norep) {
      set.seed(i)
      gpa <- rnorm(n = n, mean = 3.1, sd = 0.3)
      gre <- rnorm(n = n, mean = 580, sd = 80)
      linear <- -6 + beta[betaloop] * gpa + 0.005 * gre
      pii <- exp(linear) / (1 + exp(linear))
      x <- cbind(rep(1, n), gpa, gre)

      for (k in 1 : n) {
         y[k] <- sample(c(0, 1), 1, c(1 - pii[k], pii[k]), replace = FALSE)
      }

      ftn <- function(betacoef) {
         pi1 <- exp(x %*% betacoef) / (1 + exp(x %*% betacoef))
         gradient <- t(x) %*% (y - pi1)
         hessian <- -t(x) %*% diag(c(pi1 * (1 - pi1)), n) %*% x
         return(list(gradient, hessian))
      }

      highnew <- function(ftn, x0, tol, maxiter) {
         x <- x0
         fx <- ftn(x)
```

```
            iter <- 0
            while ((max(abs(fx[[1]])) > tol) && (iter < maxiter)) {
                x <- x - (solve(fx[[2]]) %*% fx[[1]])
                fx <- ftn(x)
                iter <- iter + 1
            }
            if (max(abs(fx[[1]])) > tol) {
                cat("algorithm failed to converge\n")
                return(NA)
            } else {
                cat("algorithm converges\n")
                return(x)
            }
        }

        mle[i, ] <- highnew(ftn, x0 = c(0, 0, 0), tol = 1e-9, maxiter = 100)
        vcov <- solve(-ftn(mle[i, ])[[2]])
        semle <- sqrt(diag(vcov))
        ward <- mle[i, ] / semle
        pvalue[i] <- ((1 - pt(abs(ward), n - 3)) * 2)[2]
    }

    for (z in 1:length(sig)) {
        rejrate[betaloop, z] <- sum(pvalue < sig[z]) / norep #rejection rate
    }
}


setwd("/Users/raymond/Desktop/R")
png(filename = "hw12.png", width = 6000, height = 6000, res = 500)
matplot(sig, t(rejrate),
    col = c(1:length(beta)),
    pch = c(1:length(beta)),
    lty = c(rep(1, 3)),
    type = "b",
    frame = FALSE,
    xlab = "Significance level",
    ylab = "Rejection rate")
abline(a = 0, b = 1, col = 8)
legend(0.04, rejrate[1, 4] + 0.05, expression(paste(beta, "=0")), bty = "n")
legend(0.04, rejrate[2, 4] + 0.05, expression(paste(beta, "=0.5")), bty = "n")
legend(0.04, rejrate[3, 4] + 0.05, expression(paste(beta, "=0.8")), bty = "n")
dev.off()
```


β=0.8
β=0.5
β=0
Rejection rate
Significance level