Example 9-1

September 11, 2020

[ ]:
```
# install the following packages and libraries
install.packages("pder")
install.packages("plm")
install.packages("pglm")
install.packages("texreg")

library("pglm")
library("plm")
library("maxLik")
library("texreg")
```

[2]:
```
# code to create a table for our results
extract.maxLik <- function (model, include.nobs = TRUE, ...){
    s <- summary(model, ...)
    names <- rownames(s$estimate)
    class(names) <- "character"
    co <- s$estimate[, 1]
    se <- s$estimate[, 2]
    pval <- s$estimate[, 4]
    class(co) <- class(se) <- class(pval) <- "numeric"
    n <- nrow(model$gradientObs)
    lik <- logLik(model)
    gof <- numeric()
    gof.names <- character()
    gof.decimal <- logical()
    gof <- c(gof, n, lik)
    gof.names <- c(gof.names, "Num. obs.", "Log Likelihood")
    gof.decimal <- c(gof.decimal, FALSE, TRUE)
    tr <- createTexreg(coef.names = names, coef = co, se = se, pvalues = pval,
                       gof.names = gof.names, gof = gof, gof.decimal = gof.
 ↪decimal)
    return(tr)
}
setMethod("extract", signature = className("maxLik", "maxLik"), definition =␣
 ↪extract.maxLik)
```

[3]:
```
##---------------------------------Block 1--------------------------------

#### Example 9-1 ####

## ------------------------------------------------------------------------
data("Reelection", package = "pder")

## ------------------------------------------------------------------------

# all 4 options are the same logit model. the family option in glm allows for
# different distributions, in this case the binomial distribution
elect.l <- glm(reelect ~ ddefterm + ddefey + gdppc + dev + nd + maj,
          data = Reelection, family = "binomial", subset = narrow)
l2 <- update(elect.l, family = binomial)
l3 <- update(elect.l, family = binomial())
l4 <- update(elect.l, family = binomial(link = 'logit'))

## ------------------------------------------------------------------------

# estimation of the probit model
elect.p <- update(elect.l, family = binomial(link = 'probit'))

## ------------------------------------------------------------------------

# estimation of the logit and probit random effects model
elect.pl <- pglm(reelect ~ ddefterm + ddefey + gdppc + dev + nd + maj,
                Reelection, family = binomial(link = 'logit'),
                subset = narrow)
elect.pp <- update(elect.pl, family = binomial(link = 'probit'))

## ------------------------------------------------------------------------

# puts results into a table
screenreg(list(logit = elect.l, probit = elect.p,
               plogit = elect.pl, pprobit = elect.pp),
          digits = 3)
```

'\n==
logit probit plogit pprobit \n--\n(Intercept) -1.328 ** -0.822 *** -1.537 ** -0.942 **\n (0.410) (0.248) (0.489) (0.294) \nddefterm 14.413 8.381 14.086 8.223 \n (7.746) (4.685) (8.211) (4.853) \nddefey 14.171 * 8.555 * 13.793 * 8.339 \n (6.660) (4.039) (6.998) (4.257) \ngdppc 17.017 * 10.652 * 19.380 * 12.076 **\n (6.911) (4.198) (7.618) (4.602) \ndev 0.822 * 0.504 * 0.893 * 0.541 * \n (0.358) (0.218) (0.430) (0.258) \nnd 0.683 0.425 0.810 0.495 \n (0.380) (0.232) (0.439) (0.264) \nmaj 0.768 * 0.472 * 0.847 * 0.515 * \n (0.314) (0.192) (0.381) (0.230) \nsigma 0.841 * -0.518 * \n (0.346) (0.205) \n--\nAIC 343.708 343.851 \nBIC 368.497 368.640 \nLog Likelihood -164.854 -164.926 -163.435 -163.434 \nDeviance 329.708 329.851 \nNum. obs. 255 255 255 255

\n==
p < 0.001; ** p < 0.01; * p < 0.05\n'