

Abstract

Data separation is a well-studied phenomenon that can cause problems in the estimation and inference from binary response models. Complete or quasi-complete separation occurs when there is a combination of regressors in the model whose value can perfectly predict one or both outcomes. In such cases, and such cases only, the maximum likelihood estimates and the corresponding standard errors are infinite. It is less widely known that the same can happen in further microeconomic models. One of the few works in the area is Santos Silva and Tenreiro (2010) who note that the finiteness of the maximum likelihood estimates in Poisson regression depends on the data configuration and propose a strategy to detect and overcome the consequences of data separation. However, their approach can lead to notable bias on the parameter estimates when the regressors are correlated. We illustrate how bias-reducing adjustments to the maximum likelihood score equations can overcome the consequences of separation in Poisson and Tobit regression models.

Software

R package `brglm2` from CRAN: <https://CRAN.R-project.org/package=brglm2>

R package `brtobit` from R-Forge: https://R-Forge.R-project.org/R/?group_id=2305

Illustration

The simplest but arguably often-encountered occurrence of data separation in practice is when there is a binary regressor such that the response $y = 0$ (or another boundary value) whenever the regressor is 1. If $P(y = 0)$ is monotonically increasing in the linear predictor of the model, then the coefficient of the binary regressor will diverge to minus infinity in order to push $P(y = 0)$ in this subgroup as close to 1 as possible.

To illustrate this phenomenon in R for both Poisson and Tobit regression we employ a simple data-generating process: In addition to the intercept we generate a continuous regressor x_2 uniformly distributed on $[-1, 1]$ and a binary regressor x_3 . The latter comes from a Bernoulli distribution with probability 0.25 if x_2 is positive and with probability 0.75 otherwise. Thus, x_2 and x_3 are correlated.

The linear predictor employed for both Poisson and Tobit is: $1 + x_2 - 10 x_3$, where the extreme coefficient of -10 assures that there is almost certainly data separation. In the full paper linked above we also consider less extreme scenarios where separation may or may not occur. The Poisson response is then drawn from a Poisson distribution using a log link between mean and linear predictor. The Tobit response is drawn from a normal distribution censored at zero with identity link and constant variance of 2. Here, we draw two samples with 100 observations from both models:

```
dgp <- function(n = 100, coef = c(1, 1, -10, 2), prob = 0.25, dist =
"poisson") {
  x2 <- runif(n, -1, 1)
  x3 <- rbinom(n, size = 1, prob = ifelse(x2 > 0, prob, 1 - prob))
  y <- switch(match.arg(tolower(dist), c("poisson", "tobit")),
    "poisson" = rpois(n, exp(coef[1] + coef[2] * x2 + coef[3] * x3)),
    "tobit" = rnorm(n, mean = coef[1] + coef[2] * x2 + coef[3] * x3, sd
```

```

= sqrt(coef[4]))
)
y[y <= 0] <- 0
data.frame(y, x2, x3)
}

set.seed(2020-10-29)
d1 <- dgp(dist = "poisson")
set.seed(2020-10-29)
d2 <- dgp(dist = "tobit")

```

Both of these data sets exhibit quasi-complete separation of y with respect to x_3 , i.e., y is always 0 if x_3 is 1.

```

xtabs(~ x3 + factor(y == 0), data = d1)
##      factor(y == 0)
## x3  FALSE TRUE
##   0     47    8
##   1      0   45

```

We then compare four different modeling approaches in this situation:

- ML: Standard maximum likelihood estimation.
- BR: Bias-reduced estimation based on adjusted score equations, first suggested by David Firth and later refined by David Firth and Ioannis Kosmidis.
- ML/sub: ML estimation on the subset not affected by separation, i.e., omitting both the regressor and all observations affected.
- ML/SST: ML estimation omitting only the regressor affected by the separation but keeping all observations. This strategy is recommended more commonly (compared to ML/sub) in the literature, specifically for Poisson in a paper by Santos Silva and Tenreiro (2010, *Economics Letters*).

For Poisson regression, all these models can be fitted with the standard `glm()` function in R. To obtain the BR estimate `method = "brglmFit"` can be plugged in using the `brglm2` package (by Ioannis Kosmidis).

```

install.packages("brglm2")
library("brglm2")

m12_ml <- glm(y ~ x2 + x3, data = d1, family = poisson)
m12_br <- update(m12_ml, method = "brglmFit")
m1_all <- glm(y ~ x2, data = d1, family = poisson)
m1_sub <- update(m1_all, subset = x3 == 0)
m1 <- list("ML" = m12_ml, "BR" = m12_br, "ML/sub" = m1_sub, "ML/SST" =
m1_all)

```

This yields the following results (shown with the wonderful [modelsummary](#) package):

```

library("modelsummary")
msummary(m1)

```

	ML	BR	ML/sub	ML/SST
(Intercept)	0.951	0.958	0.951	0.350

	ML	BR	ML/sub	ML/SST
	(0.100)	(0.099)	(0.100)	(0.096)
x2	1.011	1.006	1.011	1.662
	(0.158)	(0.157)	(0.158)	(0.144)
x3	-20.907	-5.174		
	(2242.463)	(1.416)		
Num.Obs.	100	100	55	100
Log.Lik.	-107.364	-107.869	-107.364	-169.028

The following remarks can be made:

- Standard ML estimation using all observations leads to a large estimate of x_3 with even larger standard error. As a result, a standard Wald test results in no evidence against the hypothesis that x_3 should not be in the model, despite the fact that using $x_3 = -10$ when generating the data makes x_3 perhaps the most influential regressor.
- The ML/sub strategy, i.e., estimating the model without x_3 only for non-separated observations (with $x_3 = 0$), yields exactly the same estimates as ML.
- Compared to ML and ML/sub, BR has the advantage of returning a finite estimate and standard error for x_3 . Hence a Wald test can be directly used to examine the evidence against $x_3 = 0$. The other parameter estimates and the log-likelihood are close to ML. BR slightly shrinks the parameter estimates of x_2 and x_3 towards zero.
- Finally, the estimates from ML/SST, where regressor x_3 is omitted and all observations are used, appear to be far from the values we used to generate the data. This is due to the fact that x_3 is not only highly informative but also correlated with x_2 .

Moreover, in more extensive simulation experiments in the paper it is shown that the BR estimates are always finite, and result in Wald-type intervals with better coverage probabilities.

Analogous results can be obtained for Tobit regression with our `brtobit` package, currently available from R-Forge. This provides both ML and BR estimation for homoscedastic tobit models. (Some tools are re-used from our `crch` package that implements various estimation techniques, albeit not BR, for Tobit models with conditional heteroscedasticity.) Below we fit the same four models as in the Poisson case above.

```
install.packages("brtobit", repos = "http://R-Forge.R-project.org")
library("brtobit")

m22_ml <- brtobit(y ~ x2 + x3, data = d2, type = "ML", fsmxit = 28)
m22_br <- brtobit(y ~ x2 + x3, data = d2, type = "BR")
m2_all <- brtobit(y ~ x2, data = d2, type = "ML")
m2_sub <- update(m2_all, subset = x3 == 0)
m2 <- list("ML" = m22_ml, "BR" = m22_br, "ML/sub" = m2_sub, "ML/SST" =
m2_all)
```

Because `brtobit` does not yet provide a direct interface for `modelsummary` (via `broom`) we go through the `coeftest()` results as an intermediate step. These can then be rendered by `modelsummary`:

```
library("lmtest")
m2 <- lapply(m2, coeftest)
```

```
msummary(m2)
```

	ML	BR	ML/sub	ML/SST
(Intercept)	1.135	1.142	1.135	-0.125
	(0.208)	(0.210)	(0.208)	(0.251)
x2	0.719	0.705	0.719	2.074
	(0.364)	(0.359)	(0.364)	(0.404)
x3	-11.238	-4.218		
	(60452.270)	(0.891)		
(Variance)	1.912	1.970	1.912	3.440
	(0.422)	(0.434)	(0.422)	(0.795)
Num.Obs.	100	100	55	100
Log.Lik.	-87.633	-88.101	-87.633	-118.935

The results show exactly the same pattern as for the Poisson regression above: ML, BR, and ML/sub yield results close to the true coefficients for intercept, x_2 , and the variance while the ML/SST estimates are far from the true values. For x_3 only the BR estimates are finite while the ML estimates diverge towards minus infinity. Actually, the estimates would have diverged even more if we hadn't stopped the Fisher scoring early (via `fsmexit = 28` instead of the default 100).

Overall this clearly indicates that bias-reduced (BR) estimation is a convenient way to avoid infinite estimates and standard errors in these models and to enable standard inference even when data separation occurs. In contrast the common recommendation to omit the regressor associated with the separation should be avoided or applied to the non-separated subset of observations only. Otherwise it can give misleading results when regressors are correlated.