Mind the Margins: Stay Close to The Data

Stephanie Chan and Mandy Wu

May 2, 2018

1 Introduction

Separation, a common problem for binary outcome models, occurs when one covariate or combination of covariates perfectly predict the outcome variable. Complete separation occurs when all observations can be perfectly predicted by the covariates and quasi-complete separation occurs when only a subset of the observations can be perfectly predicted.

In this paper, we explore different options when dealing with quasi-complete separation across two programming software, namely Stata and R. We focus on marginal effects as they are easier to interpret than coefficients in probit regressions. Exploring three methods of calculating marginal effects, we argue that the best method to produce marginal effect estimates is to calculate average marginal effects (AME), which brings the analyst the closest to the data.

This paper was motivated by a replication exercise of Huang et al. (2017) published in American Economic Review. The authors performed the analysis in Stata, the standard program used in most economics research. As a course assignment, we replicated the paper results in R and found interesting differences between the two programs. This section will continue with an overview of Huang et al. (2017)'s paper and the issues we uncovered. The following sections will cover a descriptive analysis between Stata and R (Section 2), provide solutions when calculating coefficients (Section 3.1), and when calculating marginal effects (Section 3.2). The paper will end with a discussion of our preferred method and future extensions (Section 4).

1.1 Background

Huang et al. (2017) examined the causes of decentralization of state-owned enterprises (SOE) in China. Decentralization is defined as a shift from higher level of government to lower level of government, i.e., shifting from central to provincial governments, or from provincial to municipal governments. The authors tested Hayek's hypothesis, which posits that SOEs further away from an oversight government are more likely to decentralize than SOEs that are closer (Hayek, 1945). Using data from a Chinese Annual Survey of Industrial Firms (ASIF) from 1998-2007, the authors found support for Hayek's conjecture.

The main specification used is a probit model, as expected with a binary outcome of decentralization. Figure 1 shows a selection of the results from the original paper, which reports the marginal effects instead of the coefficients from the regressions; this distinction will play a key role in our paper. (The paper also included robustness checks, which we do not discuss here, as we focus on understanding the core issue of separation that underlies all analyses).

In addition to main variable of interest, logged-distance ($Distance_{lag}$) from the overseeing government, the authors added additional variables as controls, including return on sales (ROS_{lag}), firm asset ($log(firm\ asset_{lag})$) and importance ($Firm\ importance_{log}$), etc. Of particular interest to us are the government, year, and industry dummies included across all specifications. There are 331 municipal-level governments, 31 provincial governments, and the central government. Each of these 363 "governments" were converted into dummies to identify the oversight government. Likewise, dummies were created for each of the 9 years in the data set, and each of the 29 industries the SOEs belong to.

Figure 1: Original Results

	Probit			
	Whole sample	Central SOEs	Provincial SOEs	Municipal SOEs
	$Decentralized_{(t)}$			
	(1)	(2)	(3)	(4)
Panel A. Baseline results				
$Distance_{lag}$	0.0054	0.0047	0.0037	0.0044
	(0.0007)	(0.0017)	(0.0008)	(0.0005)
$\log(\text{firm asset}_{lag})$	−0.0029	−0.0049	−0.0030	−0.0017
	(0.0005)	(0.0009)	(0.0010)	(0.0009)
ROS_{lag}	−0.0102	−0.0171	−0.0129	−0.0044
	(0.0015)	(0.0065)	(0.0029)	(0.0017)
$\text{Firm importance}_{lag}$	−0.0466	−0.2243	−0.1461	−0.0248
	(0.0203)	(0.5033)	(0.1049)	(0.0152)
$\text{Fully state-owned}_{lag}$	−0.0069	−0.0129	−0.0147	0.0001
	(0.0013)	(0.0051)	(0.0023)	(0.0013)
$\text{GDP per capita}_{lag}$	0.0061	−0.0026	−0.0094	0.0620
	(0.0044)	(0.0050)	(0.0331)	(0.0224)
$\text{State sector share}_{lag}$	−0.0062	−0.0958	0.0765	0.0620
	(0.0309)	(0.0239)	(0.0625)	(0.0470)
$\text{Unemployment rate}_{lag}$	−0.1128	0.2042	−0.1744	−0.1104
	(0.1100)	(0.2195)	(0.4464)	(0.2980)
Government, year, and industry dummy	YES	YES	YES	YES
Observations	69,785	11,171	20,356	38,258
Pseudo R^2	0.115	0.085	0.111	0.189

Note: This is a truncated specification from Table 3 of Huang et al's paper.

1.2 Complete & Quasi-Complete Separation

1.2.1 Definition

Separation is the presence of one or more covariates that perfectly predict the outcome of interest (Zorn, 2005). Much of the literature discusses separation in the case of a two-level covariate (e.g., females and males, vaccinated and not vaccinated) perhaps for convenience of explanation. Some scholars (e.g. Albert and Anderson (1984), Lesaffre and Albert (1989)) differentiated between "complete" and "quasi-complete" separation, where quasi-complete separation describes a scenario

where perfect prediction occurs only for a subset of observations in the data.

Formally, assuming Y is the binary outcome variable, X is the explanatory covariates, β is the coefficient estimates and we have N observations.
Complete separation occurs when all N observations can be correctly allocated as either $Y_i = 0$ or $Y_i = 1$ by a subvector $X_s \subset X$ (X_s is usually referred to as separating variables). We can also express this in mathematical term:
There exists a vector b such that:

$$b^T X_i > 0 \quad \text{if } Y_i = 1$$
$$b^T X_i < 0 \quad \text{if } Y_i = 0$$

for $i = 1, 2, ..., N$.

Quasi-complete separation occurs when only a subset of N observations can be correctly allocated as either $Y_i = 0$ or $Y_i = 1$ by separating variables X_s. We can also express this in mathematical term:
There exists a vector b such that:

$$b^T X_i \geq 0 \quad \text{if } Y_i = 1$$
$$b^T X_i \leq 0 \quad \text{if } Y_i = 0$$

for $i = 1, 2, ..., N$ and there exists at least one $k \in \{1, 2, ..., N\}$ such that $b^T X_k = 0$.

1.2.2 Coefficient estimates

Separation causes problem when using maximum likelihood to get parameter estimates and their standard errors.[1]

In the case of complete separation, the likelihood function becomes monotonic and the absolute maximum of the likelihood function is attained when parameter estimates become infinite. Hence, we do not have a finite maximum likelihood estimate for β and the variance-covariance matrix also becomes unbounded.

Similarly for quasi-complete separation, the likelihood function again becomes monotonic and the absolute maximum of the likelihood function is attained when parameter estimates of the separating variable X_s become infinite. However, the model's other covariates may remain relatively unaffected. The variance-covariance matrix again becomes unbounded.

In response to this issue, different statistical programs have defaulted into a range of decisions rules to address this issue. In the next section, we cover such decision rules for Stata and R.

[1] A detailed mathematical proof can be found in Albert and Anderson (1984). We only provide a summary here.

2 Diagnostic: Stata vs. R

2.1 Decision Rules

When given a data set with complete or quasi-complete separation, commercial program softwares have a wide range of approaches. On the rather paternalistic side, Stata automatically drops rows of observations and omits variable when the variable exhibits separation.[2] As it calculates maximum likelihood, it alerts the analyst with the following: "note: 'variable != 0' predicts failure perfectly. 'variable' dropped and X observations not used". However, authors who subsequently use these results to publish papers frequently fail to mention the dropping of observations and variables. This is the case with Huang et al. (2017), where there was no mentioning of this issue anywhere in the paper or supplementary materials[3]. This is problematic because the issue will only be uncovered when replicators (such as us) re-run their Stata code that was submitted and available on the journal's website.

Compared to Stata's paternalistic approach, R takes the "hands-free" approach, entrusting analysts to deal with the issue of separation themselves. It will run the code as instructed by analysts, not dropping any observations nor alerting the analyst when there is an issue of separation. Even if the variable predicts perfectly the outcome, R will include it into the maximum likelihood calculations. As mentioned in Section 1.2.2, this could be problematic since complete/quasi-complete separation both yield infinite coefficients. Since it does not output alerts for this, R leaves it up to the analyst to detect and decide the best path forward in dealing with separation.

2.2 Solutions

Several theoretical suggestions haven been proposed by researchers in dealing with separation. Some of them are incorporated into R packages. We provide a brief summary below.

One such alternative is the Penalized Likelihood Approach (Zorn, 2005). Zorn suggests, instead of using usual likelihood function $L(\beta|y)$ for parameter estimates, one should use a penalized likelihood function $L(\beta|y)|I(\beta)|^{\frac{1}{2}}$ proposed by Firth (1993). In his original paper, Firth incorporates the penalty term $|I(\beta)|^{\frac{1}{2}}$ in the likelihood function to eliminates small-sample bias found in logistic regression models. Zorn (2005) argues that this penalized likelihood approach can also resolve separation problem as it produces finite estimates under separation cases. This method is incorporated into several R packages, including *brlr* and *logistf*.

Another approach is the Bayesian likelihood approach: placing prior information into maximum likelihood function through Bayes' rule:

$$p(\beta|y) = \frac{p(y|\beta)p(\beta)}{\int p(y|\beta)p(\beta)d\beta}$$

Interestingly, Poirier (1994) proves that for logistic model, Firth's penalized likelihood is the same as a Bayesian approach with Jeffreys' prior. Gelman (2008), however, argues that Jeffery's prior can not be easily interpreted as the actual prior information since it depends on the data in a complicated manner. He proposes using a Cauchy prior with scale 2.5 on the coefficients instead.

[2]We notice that Stata do not drop all observations associated with the separating variables. Since it is not an open-sourced program like R is, we are still locating the source code to understand this decision.

[3]They made one mention of this in their replication code, and then go on to state that even though Stata drops observations, they will report observations numbers prior to Stata's dropping.

This method is incorporated into R packages, including *arm*.

For this paper, we use Gelman's Cauchy prior and package *arm* instead of Jeffery's prior due to the above argument.

3 Calculating Coefficients and Marginal Effects

In order to understand the differences between Stata and R, we experimented with existing packages as well as ad-hoc calculations. First, we present regression coefficients of the key covariates[4] from probit analyses on the whole sample. We found that the results from the two programs are in fact relatively similar. Then, we present different methods of calculating marginal effects. While some existing packages and commands/functions hold covariates at their mean values, it is sometimes difficult to interpret (as it is the case in the Huang et al. (2017) data we use). A simple improvement is to hold covariates at their median values, which results in more interpretable results. However, we argue that the best method is to be close to the data by calculating Average Marginal Effects (AME).

3.1 But first, Coefficients

While we found the marginal results to be different in Huang et al. (2017)'s paper because of Stata and R's different decision rules, the regression coefficients of the key covariates are largely unaffected by separation. As argued before, in the case of quasi-complete separation, only the parameter estimates of the separating variables (in this case, a portion the dummies variables) will be infinite, while the estimates of other variables remain unaffected. On the other hand, in the case of complete separation, the coefficients of other covariates will be zero (and standard error, infinite) because there's no variance left to explain besides the separating variable (Zorn, 2005).

In Table 1, we show the coefficient estimates of the key covariates, those displayed in (Huang et al., 2017)'s paper[5]. Specification (1) shows the output from Stata after it dropped observations and omitted variables where it perfectly predicts decentralization. Specification (2) to (4) are outputs from R. We tried different variations to test the stability of our estimates. In Specification (2), we used *bayesglm* function in the *arm* package (as described in Section 2.2). In Specification (3), we used the canonical *glm* function with a probit link. There is a slight difference between the *bayesglm* and *glm* result. This is because *bayesglm* adds Cauchy(0,2.5) prior to the maximum likelihood estimates of the separating variables. In Specification (4), we identified the observations that Stata used to calculate maximum likelihood and used the *glm* function on the selected data.

We show that in line with Zorn (2005)'s observation, the regression coefficients of covariates that do not have separation are unaffected. Since separation only occurs in the government, year, and industry dummies in our data set, the values in Table 1 are largely aligned.

[4]Key covariates are those reported in the orginial Huang et al. (2017) paper.

[5]For the remainder of this paper, we focus on understanding results from the whole sample. The original authors ran separate regressions for different portions of the data (see Figure 1). Our exploration and conclusion for this paper will extend to all of their other regressions.

Table 1: Coefficients of probit regression under different software packages

	Stata	R		
		`Bayesglm`	`glm`	`glm` using Stata obs
	Decentralized$_{(t)}$	Decentralized$_{(t)}$		
	(1)	(2)	(3)	(4)
Distance_lag	0.1402	0.1439	0.1402	0.1402
	(0.0100)	(0.0098)	(0.0102)	(0.0102)
log(firm asset)_lag	−0.0744	−0.0738	−0.0744	−0.0744
	(0.0076)	(0.0075)	(0.0075)	(0.0075)
ROS_lag	−0.2657	−0.2639	−0.2657	−0.2657
	(0.0462)	(0.0455)	(0.0456)	(0.0456)
Firm importance_lag	−1.2118	−1.2340	−1.2118	−1.2117
	(0.3966)	(0.3950)	(0.4005)	(0.4004)
Fully state-owned_lag	−0.1789	−0.1766	−0.1789	−0.1789
	(0.0339)	(0.0338)	(0.0339)	(0.0339)
GDP per capita_lag	0.1589	0.1658	0.1589	0.1589
	(0.0581)	(0.0574)	(0.0579)	(0.0579)
State sector share_lag	−0.1611	−0.1390	−0.1611	−0.1611
	(0.2693)	(0.2664)	(0.2676)	(0.2676)
unemployment rate_lag	−2.9329	−3.0109	−2.9329	−2.9328
	(2.2718)	(2.2581)	(2.2636)	(2.2636)
Government, year, and industry dummy	YES	YES	YES	YES
Log Likelihood	-5968.9156	-5973.4708	-5968.9157	-5968.9156
Observations	55408	69785	69785	55408

Notes: In this table, we displayed the coefficient of probit regression on the whole sample. Column 1 reports coefficients produced by Stata. Columns $2-4$ reports coefficients produced by R: column 2 uses command `Bayesglm` in `arm` package, column 3 uses `glm` in `stats` package, column 4 also uses `glm` in `stats` package but reduced the sample size to those used by Stata.

3.2 Margins in quasi-complete separation

The calculation of coefficients seem to go unscathed insofar as the variables of interest are not variables with separation in a quasi-complete separation situation. We still, however, urge that authors mention about any form of separation if it occurs in the data to inform the readers of the situation, and how (if at all) it was addressed.

The situation becomes more important when calculating marginal effects because marginal effect calculations take into account all covariates, including those with separation. Here, existing program and default settings begin to diverge in calculations, therefore producing different results.

3.2.1 Holding other covariates at Mean

In Huang et al. (2017), the authors reported marginal effects in Stata. The particular command they used sets the covariates at its means by default.[6] In Table 2, we show the results from Stata in Specification (1), which align with the values from the original paper (see Figure 1, Specification (1)). We turn our discussion to calculating margins by setting covariates at mean in R.

In R, we calculated the marginal effects by first drawing 10000 simulations of β from our regression outputs, then calculating marginal effects by plugging each simulated βs into the marginal effect formula while holding other covariates at their mean, resulting in 10000 simulated margins. We reports the mean and standard error of the 10000 simulated margins. We repeat the same process for the different specifications. The results are reported in Table 2, Specification (2), (3) and (5).[7] The differences between the estimates in the three specification are due to separation. Rainey (2016) shows that by adding a prior, both the point estimates and standard errors of the separating variables will shrink towards zero, resulting in smaller point-estimates as well as standard errors than standard maximum likelihood estimates.

R also has a vanilla package for calculating marginal effects: *mfx*, with the *probitmfx* function for probit regressions. Although we report the results in specification (4) and (6) in Table 2, we would like to alert future users to be cautious about how the package calculates marginal effects. To get point estimates of marginal effect, *probitmfx* plugs the point estimates of β into the marginal effects formula. In doing so, the author fails to account for the estimation uncertainty of β and potentially biases the marginal effect estimation. We can see the result of this bias in specification (4), where the point estimates of marginal effects are much smaller those produced by simulations.

Recall that thus far, we set covariates at its mean values. These are sensible to interpret in the main covariates such as firm asset, GDP, and other continuous variables. However, interpretation becomes difficult in terms of dichotomous variables (i.e., government, year, and industry dummies). What does it mean to set dummy say, gov_304 = 0.0036637 when the actual values can only be 0 or 1? As an alternative descriptive statistic, the median can overcome the issue of interpretation. We explore this method next.

[6]The authors used command "dprobit" to report marginal effects in probit regressions. However, As of Stata Version 11, this is no longer supported by Stata, and command "margins" is recommended instead. As we will discuss later, "margins" by default will report Average Marginal Effects, but has the option to set covariates at their means. Both "dprobit" and "margins"(at mean) will return the same results.

[7]Specification (2) we used *bayesglm* function. Specification (3) we used the canonical *glm* function. Specification (5) we used *glm* function but limit our observation samples to those Stata kept after dropping observations.

Table 2: Marginal effects holding covariates at mean

	Stata	R				
		`Bayesglm` with Simulations	`glm` with Simulations	`glm` with `probitmfx`	`glm` with simulations Stata obs	`glm` with `probitmfx` Stata obs
	Decentralized$_{(t)}$	Decentralized$_{(t)}$				
	(1)	(2)	(3)	(4)	(5)	(6)
Distance_lag	0.0054	0.0027	0.0103	0.0006	0.0054	0.0054
	(0.0004)	(0.0003)	(0.0175)	(0.0078)	(0.0004)	(0.0004)
log(firm asset)_lag	-0.0029	-0.0014	-0.0055	-0.0003	-0.0029	-0.0029
	(0.0003)	(0.0002)	(0.0094)	(0.0041)	(0.0003)	(0.0003)
ROS_lag	-0.0102	-0.0049	-0.0196	-0.0012	-0.0102	-0.0102
	(0.0018)	(0.0009)	(0.034)	(0.0147)	(0.0017)	(0.0018)
Firm importance_lag	-0.0466	-0.0228	-0.0884	-0.0055	-0.0464	-0.0466
	(0.0152)	(0.0073)	(0.1602)	(0.0673)	(0.0154)	(0.0153)
Fully state-owned_lag	-0.0069	-0.0033	-0.0132	-0.0008	-0.0069	-0.0069
	(0.0013)	(0.0007)	(0.0229)	(0.0099)	(0.0013)	(0.0013)
GDP per capita_lag	0.0061	0.0031	0.0117	0.0007	0.0061	0.0061
	(0.0022)	(0.0011)	(0.0216)	(0.0088)	(0.0022)	(0.0022)
State sector share_lag	-0.0062	-0.0026	-0.012	-0.0007	-0.0061	-0.0062
	(0.0104)	(0.005)	(0.0442)	(0.009)	(0.0103)	(0.0103)
unemployment rate_lag	-0.1128	-0.0567	-0.2165	-0.0132	-0.1114	-0.1128
	(0.0873)	(0.0421)	(0.4942)	(0.1631)	(0.0877)	(0.087)
Government, year, and industry dummy	YES	YES	YES	YES	YES	YES
Observations	55408	69785	69785	69785	55408	55408

Notes: In this table, we displayed the marginal effect from probit regression when setting covariates at their mean. Column 1 reports coefficients produced by Stata. Columns $2-6$ reports coefficients produced by R: column 2 uses command `Bayesglm` to produce coefficients and simulation to produce marginal effects, column $3-4$ uses `glm` to produce coefficients and column 3 use simulation to produce marginal effects while column 4 use command `probitmfx` in package `mfx`, column $5-6$ uses the same settings as column $3-4$ but reduces sample size to those used by Stata.

3.2.2 Holding other covariates at Median

As mentioned above, using the median has the advantage of setting covariates at a level that makes sense for dummies, as well as continuous variables. To calculate marginal effects, we follow the same process as section 3.2.1. The only change is instead of holding other covariates at their mean, we switch to median. Table 3 shows the results of setting covariates at their median.

The results are similar across all the specifications. This is largely because our dummies are very right-skewed. In Huang et al. (2017), the median for all dummies are 0 (i.e., there are very few decentralized SOEs relative to non-decentralized SOEs). As a result, when we calculate the marginal effects, whether through simulation or *probitmfx*, the effect of separation almost vanishes. To provide a mathematical interpretation, we look at the marginal effects formula.

For continuous variable, marginal effect of variable X_j is:

$$\frac{\partial \Phi(X\beta)}{\partial X_j} = \varphi(X\beta) \cdot \beta_j$$

Since we set separating variables to be 0 (their median values), no matter what the point estimates are, $X_s\beta_s$[8] would be 0. This suggests that separation would not cause large changes in marginal effects when the value of separating variables are set to 0.[9]

Another alternative to setting covariates at median is to calculate the average marginal effect. This has the advantage of absorbing the variations in the data (instead of just representing variations with a specific value, e.g., at mean or median), which results in a more accurate representation of the data.

[8] X_s is the set of variables causing separation and β_s are their estimates

[9] Separation can still affect non-separating variables, but as we argued before, in a quasi-complete separation case, the point estimates and standard errors of these covariates remain relatively unchanged.

Table 3: Marginal effects holding covariates at median

	Stata	R				
		`Bayesglm` with Simulations	`glm` with Simulations	`glm` with `probitmfx`	`glm` with simulations Stata obs	`glm` with `probitmfx` Stata obs
	Decentralized$_{(t)}$	Decentralized$_{(t)}$				
	(1)	(2)	(3)	(4)	(5)	(6)
Distance_lag	0.011	0.0084	0.0113	0.0103	0.0121	0.011
	(0.0093)	(0.0044)	(0.0058)	(0.0056)	(0.0061)	(0.0059)
log(firm asset)_lag	-0.0058	-0.0043	-0.006	-0.0055	-0.0065	-0.0058
	(0.0049)	(0.0023)	(0.0032)	(0.0031)	(0.0034)	(0.0032)
ROS_lag	-0.0209	-0.0155	-0.0216	-0.0195	-0.0231	-0.0209
	(0.0178)	(0.0086)	(0.0119)	(0.0113)	(0.0125)	(0.0118)
Firm importance_lag	-0.0952	-0.0725	-0.0988	-0.089	-0.1052	-0.0951
	(0.0891)	(0.0464)	(0.0632)	(0.0582)	(0.0665)	(0.0612)
Fully state-owned_lag	-0.014	-0.0103	-0.0146	-0.0131	-0.0155	-0.014
	(0.0121)	(0.0058)	(0.0082)	(0.0077)	(0.0086)	(0.0081)
GDP per capita_lag	0.0125	0.0096	0.0128	0.0117	0.0138	0.0125
	(0.0112)	(0.0062)	(0.0083)	(0.0076)	(0.0088)	(0.008)
State sector share_lag	-0.0126	-0.0086	-0.0138	-0.0118	-0.0144	-0.0126
	(0.0240)	(0.0184)	(0.0265)	(0.021)	(0.0274)	(0.0224)
unemployment rate_lag	-0.2304	-0.1751	-0.2328	-0.2155	-0.2473	-0.2302
	(0.2717)	(0.1727)	(0.2349)	(0.1988)	(0.2489)	(0.2107)
Government, year, and industry dummy	YES	YES	YES	YES	YES	YES
Observations	55408	69785	69785	69785	55408	55408

Notes: In this table, we displayed the marginal effect from probit regression when setting covariates at their median. Column 1 reports coefficients produced by Stata. Columns $2-6$ reports coefficients produced by R: column 2 uses command `Bayesglm` to produce coefficients and simulation to produce marginal effects, column $3-4$ uses `glm` to produce coefficients and column 3 use simulation to produce marginal effects while column 4 use command `probitmfx` in package `mfx`, column $5-6$ uses the same settings as column $3-4$ but reduces sample size to those used by Stata.

3.2.3 Average Marginal Effects

By default, Stata reports average marginal effects when using command 'margins' with no specifications[10]. We do the same in R, incorporating estimation uncertainty of β into marginal effects through simulations of β[11]. To do this, we wrote a function that does the following: first, calculate the marginal effects for one simulated β. For each observation, it calculates the marginal effects for each covariate by holding the covariate of interest at its mean[12] and all other covariates at their original values. Repeat this for all observations and report the mean as marginal effect for this one β. Then, repeat the first step for all simulated βs and calculate mean and standard errors.

In this way, we make use of every observation in the data set, and only estimate the marginal effects by tweaking the covariate of interest at the time of calculation.

Table 4 shows the results of AMEs for Stata and R. The values are relatively similar across specifications. The results across all specifications are also quite similar, with a magnitude in between the mean specifications (Table 2) and median specifications (Table 3).

[10] As mentioned above, the default of 'dprobit' in Stata will set covariates at mean

[11] We only report the result by simulate β 1000 times to reduce computation time. However, simulate β 10000 times produce similar result.

[12] Given that the key covariates are continuous, mean values are interpretable. We also calculated average marginal effects by holding covariate values at their medians and found that the two results are almost identical.

Table 4: Average marginal effects - holding covariates at counterfactual

	Stata	R		
		`Bayesglm` with Simulations	`glm` with Simulations	`glm` with simulations Stata obs
	Decentralized$_{(t)}$	Decentralized$_{(t)}$		
	(1)	(2)	(3)	(4)
Distance_lag	0.0077	0.0064	0.0054	0.0074
	(0.0006)	(0.0005)	(0.0004)	(0.0005)
log(firm asset)_lag	-0.0041	-0.0037	-0.0034	-0.0042
	(0.0004)	(0.0004)	(0.0004)	(0.0004)
ROS_lag	-0.0146	-0.0134	-0.0122	-0.0152
	(0.0026)	(0.0023)	(0.0021)	(0.0026)
Firm importance_lag	-0.0667	-0.0613	-0.0541	-0.069
	(0.0219)	(0.0184)	(0.0171)	(0.023)
Fully state-owned_lag	-0.0098	-0.0091	-0.0083	-0.0102
	(0.0019)	(0.0017)	(0.0016)	(0.0019)
GDP per capita_lag	0.0087	0.0087	0.0073	0.0091
	(0.0032)	(0.003)	(0.0028)	(0.0035)
State sector share_lag	-0.0089	-0.0067	-0.0076	-0.0093
	(0.0148)	(0.0133)	(0.0126)	(0.0158)
unemployment rate_lag	-0.1614	-0.1581	-0.1354	-0.1719
	(0.1250)	(0.1118)	(0.1035)	(0.128)
Government, year, and industry dummy	YES	YES	YES	YES
Observations	55408	69785	69785	55408

Notes: In this table, we displayed the marginal effect from probit regression when holding covariates at their counterfactuals. Column 1 reports coefficients produced by Stata. Columns $2-4$ reports coefficients produced by R: column 2 uses command `Bayesglm` to produce coefficients and simulation to produce marginal effects, column 3 uses `glm` to produce coefficients and simulation to produce marginal effects, column 4 uses the same settings as column 3 but reduces sample size to those used by Stata. We didn't report marginal effects calculated by `probitmfx` because it reports error message `"Error: vector memory exhausted (limit reached?)"`.

4 Discussion

We presented three methods of calculating marginal effects for probit regressions: mean, median, and AME. AME is our preferred calculation because it gives estimates that are closest to the data. As political scientists and economists encounter observational data sets, which are fixed by the authors of the survey and not at the discretion of later analysts, it is important that scholars make the best use of the data set that they have. While mean and medians are often good representations of the data, they come with several downfalls. If using the mean, it could result in interpretation problems where dichotomous variables are represented in values that cannot exist. If using the median, it could be sensitive to skewness of the data. We recommend using AME with simulations in calculating marginal effects.

Moreover, our data set has the additional issue of quasi-complete separation. Using the Bayesian Likelihood Approach, we adjusted the maximum likelihood calculation by adding a Cauchy(0,2.5) prior based on Gelman (2008). Using the coefficient estimates from the bayesian model, we calculated marginal effects. We compared the result produced by *bayesglm* and *glm*, and notice that point estimates from the Bayesian model are consistently smaller than the regular maximum likelihood model, as argued in Rainey (2016). We also compared the result produced by simulation and *probitmfx*, and encourage researchers to use simulation when generating quantities of interest as *probitmfx* failed to account for the estimation uncertainty.

Naturally, researchers have preferences (personal or field specific) for which programming language to use when analyzing data. As this paper highlights, different programs can have quite different decision rules of when to keep and omit observations and variables, as well as methods of calculations (i.e., setting at mean or median or AME). These decisions by the program writers have important downstream consequences as it may affect how analysts interpret the results, and derive at conclusions thereafter. It is important then, for analysts to gain a good understanding of the different methods taken by the program, and adjust accordingly. For example, if the authors were using R for Huang et al. (2017) instead of Stata, they would have to acknowledge the issue of separation before the analysis stage, weigh the different options to analyze data with quasi-complete separation, and write a code that would accommodate their decision[13].

In future work, we hope to apply the AME method to other quasi-complete separation data sets. This will allow us to assess the stability of our code. As it stabilizes, we hope to write a R package, which will improve on the *probitmfx* function (from *mfx* package) that produces concerning estimates yet quite widely used.

[13]As mentioned before, the vanilla code for 'probitmfx' available in R is not preferred, and should the authors run the analysis in R, they should write their own code

References

Albert, A. and J. A. Anderson, "On the Existence of Maximum Likelihood Estimates in Logistic Regression Models," *Biometrika*, 1984, *71* (1), 1–10.

Firth, David, "Bias Reduction of Maximum Likelihood Estimates," *Biometrika*, 1993, *80* (1), 27–38.

Gelman, Andrew, "Scaling regression inputs by dividing by two standard deviations," *Statistics in medicine*, 2008, *27* (15), 2865–2873.

Hayek, F.A.von, "The Use of Knowledge in Society," *American Economic Review*, 1945, *35* (4), 519–530.

Huang, Zhangkai, Lixing Li, Guangrong Ma, and Lixin Colin Xu, "Hayek, Local Information, and Commanding Heights: Decentralizing State-Owned Enterprises in China," *American Economic Review*, 2017, *107* (8), 2455–2478.

Lesaffre, E. and A. Albert, "Partial Separation in Logistic Discrimination," *Journal of the Royal Statistical Society. Series B*, 1989, *51* (1), 109–116.

Poirier, Dale, "Jeffreys' prior for logit models," *Journal of Econometrics*, 1994, *63* (2), 327–339.

Rainey, Carlisle, "Dealing with separation in logistic regression models," *Political Analysis*, 2016, *24* (3), 339–355.

Zorn, Christopher, "A Solution to Separation in Binary Response Models," *Political Analysis*, 2005, *13*, 157–170.