

Advanced Programming in Stata

Kevin Sweeney
PRISM Senior Methods Fellow

Brandon Bartels
PRISM Junior Methods Fellow



Advanced Programming in Stata

- Programming your own maximum likelihood estimator.
 - Basic syntax
 - Likelihood functions
- Examples:
 - Normal regression (easy)
 - Logit and probit (easy)
 - Heteroskedastic regression (harder)
 - Split population duration model (harder)

Programming Likelihood Functions: The Basics

- As you will see, *programming your own ML estimator is incredibly easy to do in Stata.*
- From last session, we learned how to write a program in Stata using .do files, macros, looping, etc.
- In writing our own likelihood function, we need the following information:
 - An understanding of some of Stata’s “ml” family of commands.
 - *Note:* The help menus provide very useful information on MLE programming; *help ml* and/or *help mlmethod*
 - Log-likelihood function
 - Syntax for how to maximize the function
 - *THAT’S IT!* It’s so easy, it’s hard to believe!

Programming Likelihood Functions: Brief MLE Review

- In ML, we first need to specify the data generating process for the dependent variable under examination.
- In other words, we need to specify the probability distribution that generated the dependent variable; e.g., the normal for continuous variable, logit or probit for dichotomous, poisson for count data, etc.

Programming Likelihood Functions: Brief MLE Review

- Then, we specify the likelihood for case i :

$$L_i = L(\mathbf{q} \mid y_i)$$
$$L(\mathbf{q} \mid y_i) \propto p(y_i \mid \mathbf{q})$$

- The likelihood for the entire sample is simply the product of individual likelihoods:

$$L = \prod_{i=1}^N L_i$$

- MLEs are the values of the parameters for which the likelihood of observing the sample is maximized.

Programming Likelihood Functions: ML Normal Regression

- $Y \sim N(\mu, s^2)$

- pdf:
$$f(y_i | \mathbf{m}_i, \mathbf{s}^2) = \frac{1}{\sqrt{2p\mathbf{s}^2}} e^{-\frac{1}{2} \frac{(y_i - \mathbf{m}_i)^2}{\mathbf{s}^2}}$$

- Reparameterize $\mu_i = x_i \beta$

- Likelihood for case i :
$$L_i = \frac{1}{\sqrt{2p\mathbf{s}^2}} e^{-\frac{1}{2} \frac{(y_i - x_i \mathbf{b})^2}{\mathbf{s}^2}}$$

- Log-likelihood for case i (**this is what Stata wants**):

$$\ln L_i = -\frac{1}{2} \ln(2p) - \frac{1}{2} \ln(\mathbf{s}^2) - \frac{1}{2} \left[\frac{(y_i - x_i \mathbf{b})^2}{\mathbf{s}^2} \right]$$

Programming Likelihood Functions: ML Normal Regression

- The likelihood for the entire sample is simply the product of the individual likelihoods:

$$L = \prod_{i=1}^N \frac{1}{\sqrt{2\pi s^2}} e^{-\frac{1}{2} \frac{(y_i - x_i \mathbf{b})^2}{s^2}}$$

- And the log-likelihood for the entire sample is simply:

$$\ln L = -\frac{N}{2} \ln(2\pi) - \frac{N}{2} \ln(s^2) - \frac{1}{2} \sum_{i=1}^N \left[\frac{(y_i - x_i \mathbf{b})^2}{s^2} \right]$$

- Again, however, Stata only needs the log-likelihood for case i .

Programming Likelihood Functions: Syntax

- Goal: Write a program that Stata can use to maximize a log-likelihood function.
- First, Stata has 4 ML “evaluators”: lf, d0, d1, d2.
- “lf” is the most basic evaluator; the “d” evaluators are for more advanced programs. We’re only going to use “lf” in this session.

Programming Likelihood Functions: Syntax

```
program define proname  
  args lnf theta1 theta2 ...  
  tempvar tmp1 tmp2 ...  
  quietly gen double `tmp1' = ...  
  quietly replace `lnf' = ...  
end
```

- `lnf' is a variable to be filled in with values of the log-likelihood for case i (i.e., $\ln L_i$).
- `theta1' is associated with the first parameter, containing evaluation of the 1st equation: $\text{theta1}_i = x_{1i} b$
- `theta2' is associated with the second parameter, containing evaluation of the 2nd equation: $\text{theta2}_i = x_{2i} b$

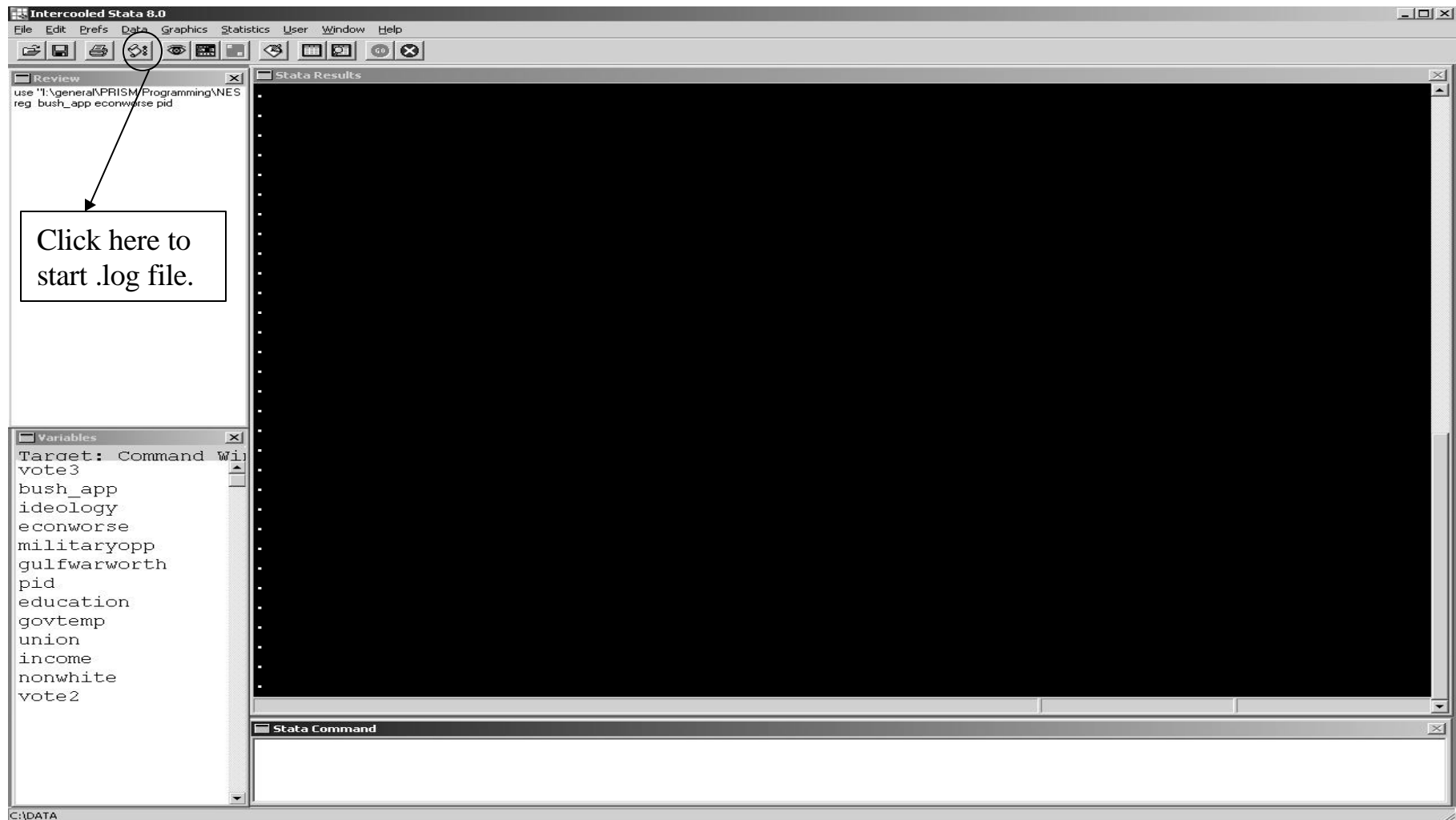
Programming Likelihood Functions: Syntax

- Global macros:

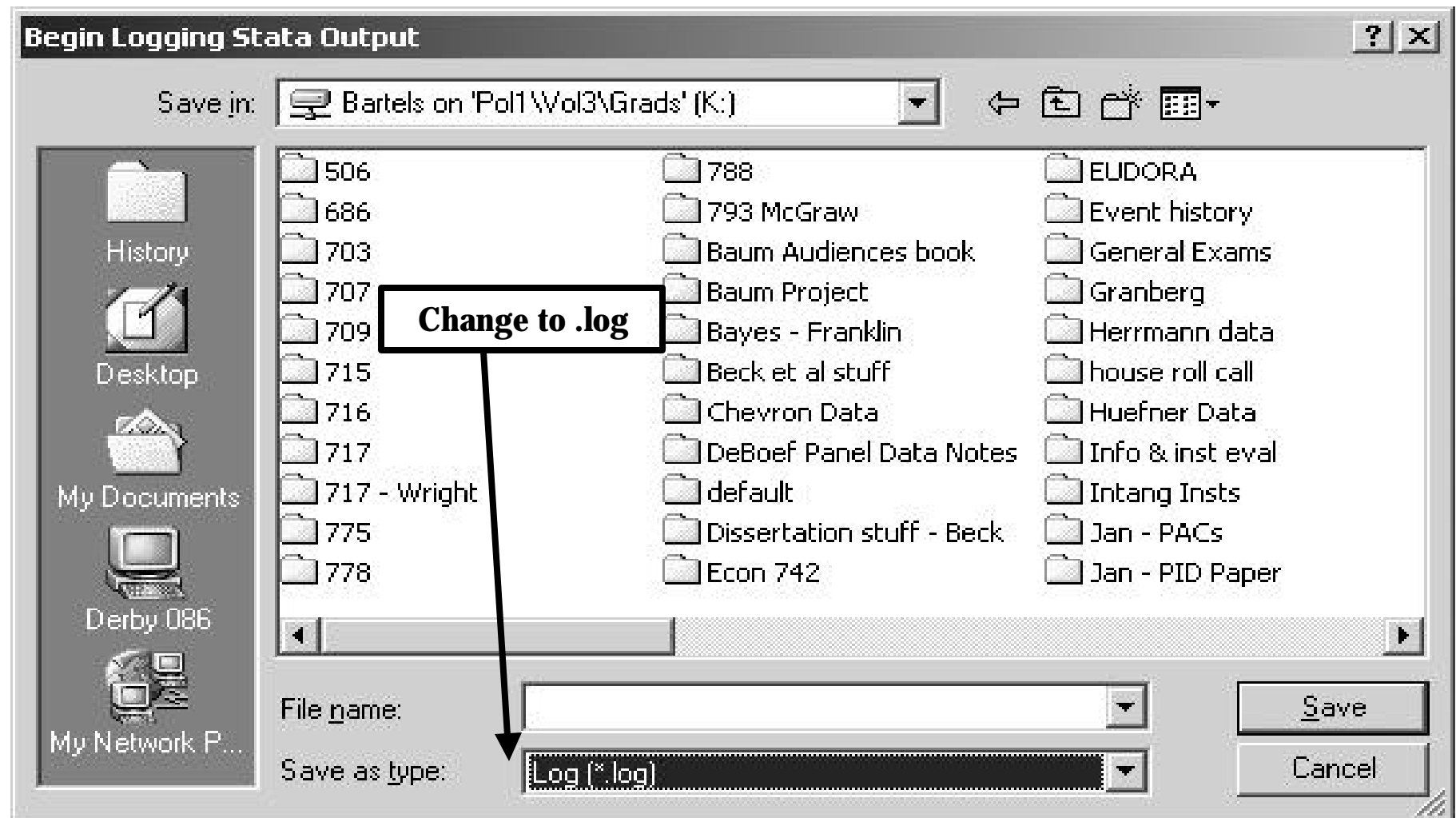
`$ML_y1` is a global macro for the name of the first dependent variable.

`$ML_y2` is a global macro for the name of the second dependent variable.

Onto the Machines: Start a .log File



Onto the Machines: Start a .log File



Programming Likelihood Functions: ML Normal Regression Program

- Let's open some data: 1992 NES
- File, Open

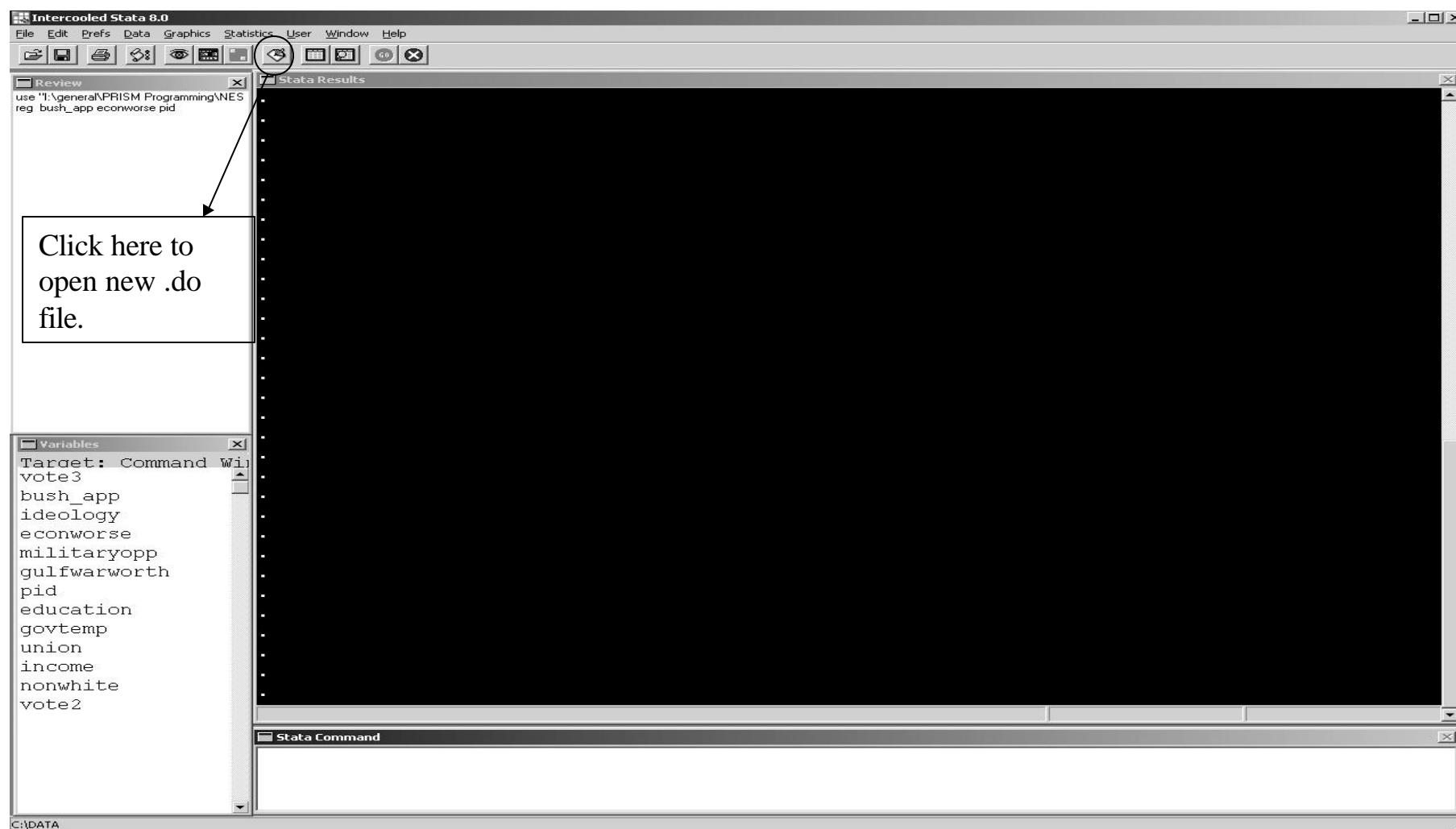
Go to the I: drive

Double-click on “general”

Double-click on “PRISM Programming”

Double-click on “NES 1992.dta”

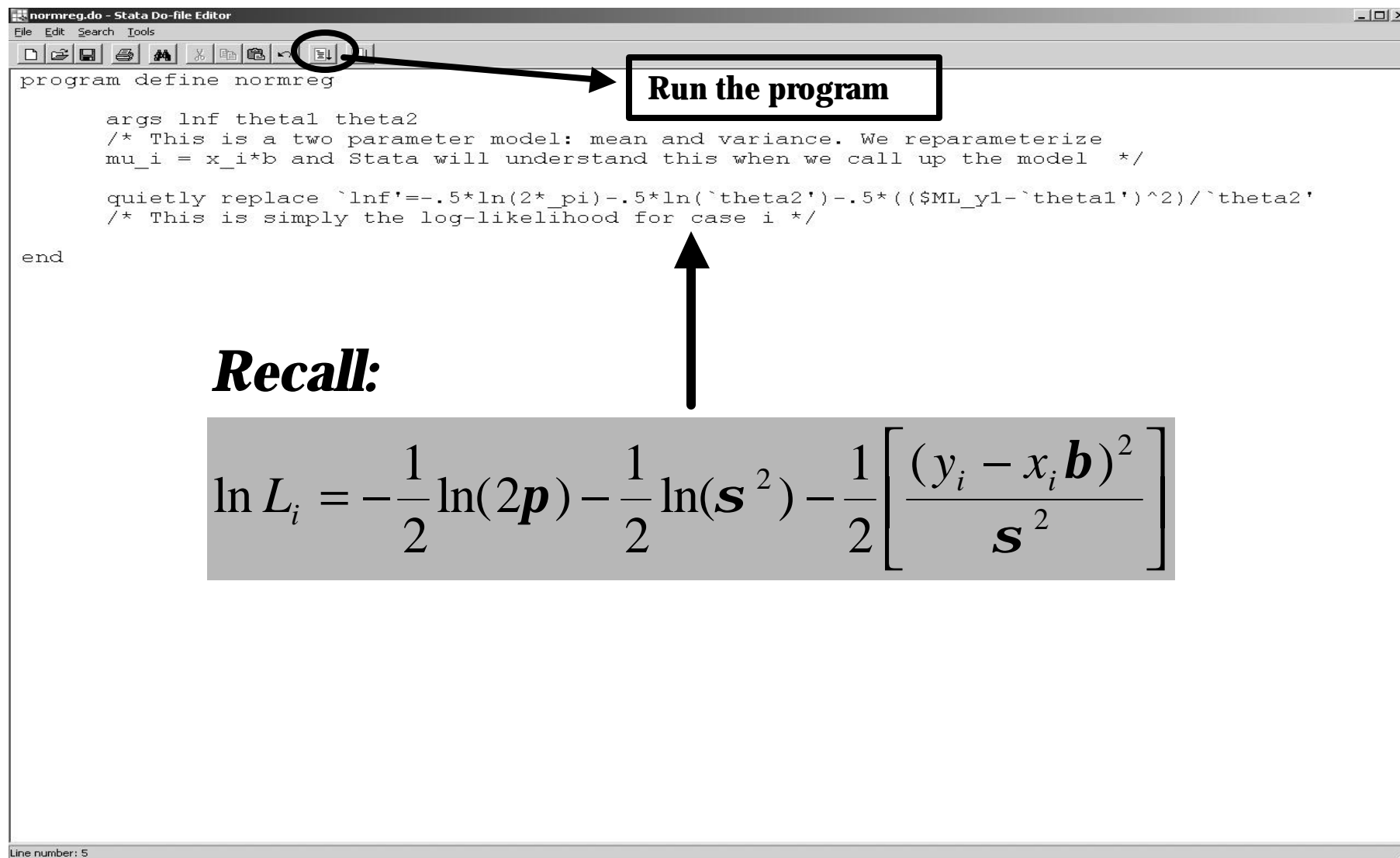
Programming Likelihood Functions: ML Normal Regression Program



Programming Likelihood Functions: ML Normal Regression Program

- Open “normreg” program from the .do file editor:
 - Go to the I: drive
 - Double-click on “general”
 - Double-click on “PRISM Programming”
 - Double-click on “normreg.do”

Programming Likelihood Functions: ML Normal Regression Program



```
normreg.do - Stata Do-file Editor
File Edit Search Tools
[Icons]
program define normreg

    args lnf thetal1 theta2
    /* This is a two parameter model: mean and variance. We reparameterize
    mu_i = x_i*b and Stata will understand this when we call up the model */

    quietly replace `lnf'=-.5*ln(2*_pi)-.5*ln(`theta2')-.5* (($ML_y1-`thetal1')^2)/`theta2'
    /* This is simply the log-likelihood for case i */

end
```

Run the program

Recall:

$$\ln L_i = -\frac{1}{2} \ln(2p) - \frac{1}{2} \ln(s^2) - \frac{1}{2} \left[\frac{(y_i - x_i \mathbf{b})^2}{s^2} \right]$$

Line number: 5

Programming Likelihood Functions: Maximizing the Likelihood Function

- Once we've written the program, we need to tell Stata to estimate it. This takes two steps:

(1) *ml model lf progname (eq1: y=x1 x2 x3)*

- or -

ml model lf progname (eq1: y=x1 x2 x3) (eq2: y=x1 x2 x3)

- or -

ml model lf progname (eq1: y=x1 x2 x3) /parameter

[If the second parameter is not reparameterized as a function of covariates, e.g., s^2 in ML normal regression.]

(2) *ml max*

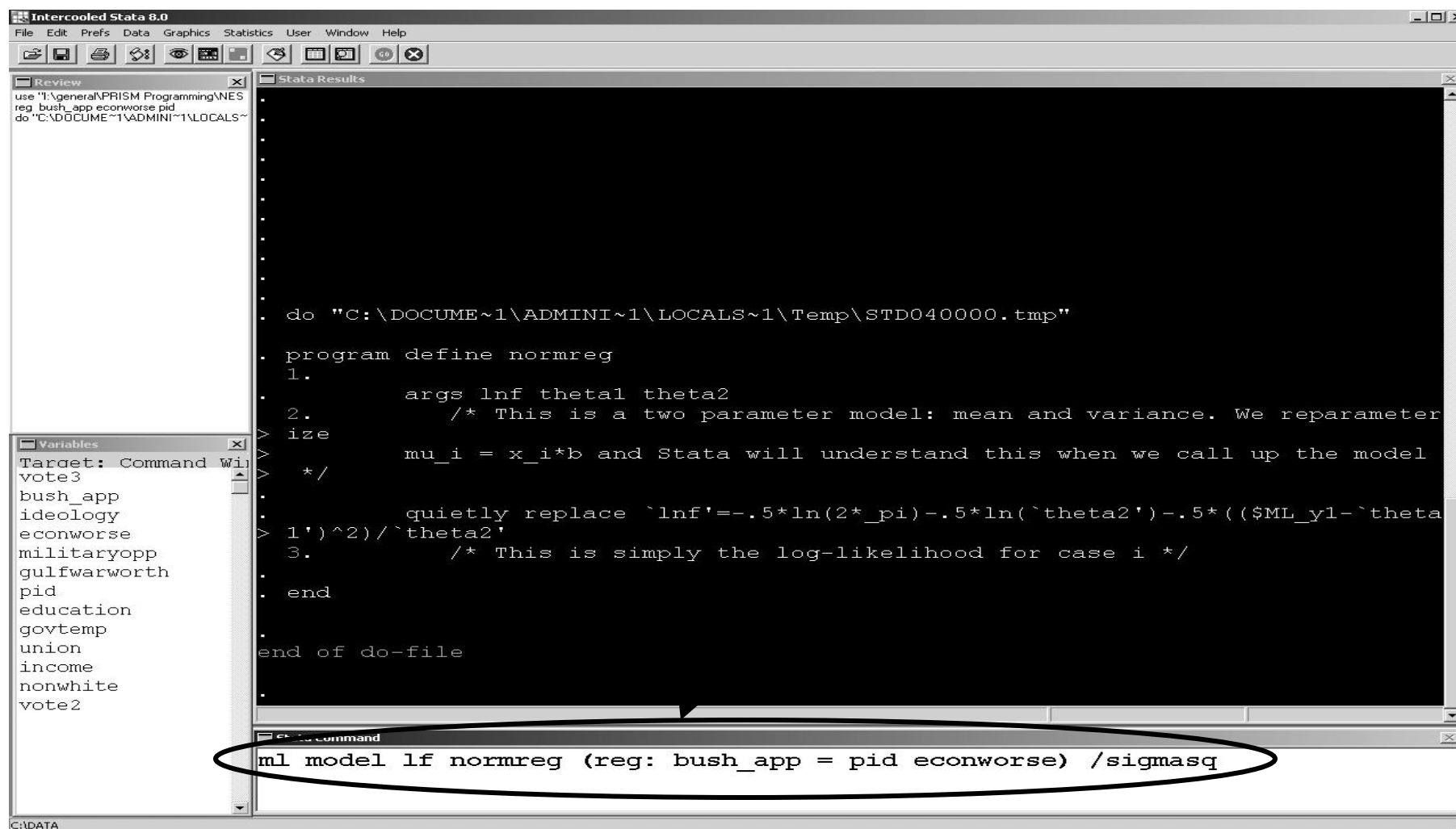
Programming Likelihood Functions: Maximizing the Likelihood Function

- Other useful commands to run after *ml model*:
 - ml check* verifies that the program you wrote works
 - ml search* searches for better starting values
 - lft0(#k LL0)* reports a likelihood ratio test (included after the “ml model” command), comparing fully specified model to an intercept only (i.e., null) model. The Wald test is produced by default. For the LR test, you need to specify the LL and the number of parameters for the intercept only model.

Programming Likelihood Functions: Estimating the ML Normal Model

- Let's estimate a simple model; we'll regress George H.W. Bush's approval on PID and economic perceptions.

Programming Likelihood Functions: Estimating the ML Normal Model



The screenshot shows the Intercooled Stata 8.0 interface. The main window displays a do-file with the following content:

```
. use "I:\general\PRISM Programming\NES  
. reg bush_app econworse pid  
. do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"  
  
. do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"  
  
. program define normreg  
1.  
    args lnf theta1 theta2  
2.  
    /* This is a two parameter model: mean and variance. We reparameter  
ize  
    mu_i = x_i*b and Stata will understand this when we call up the model  
    */  
3.  
    quietly replace `lnf' = -.5*ln(2*_pi) - .5*ln(`theta2') - .5*((`$ML_y1' - `theta  
1')^2)/`theta2'  
4.  
    /* This is simply the log-likelihood for case i */  
5.  
end  
  
end of do-file
```

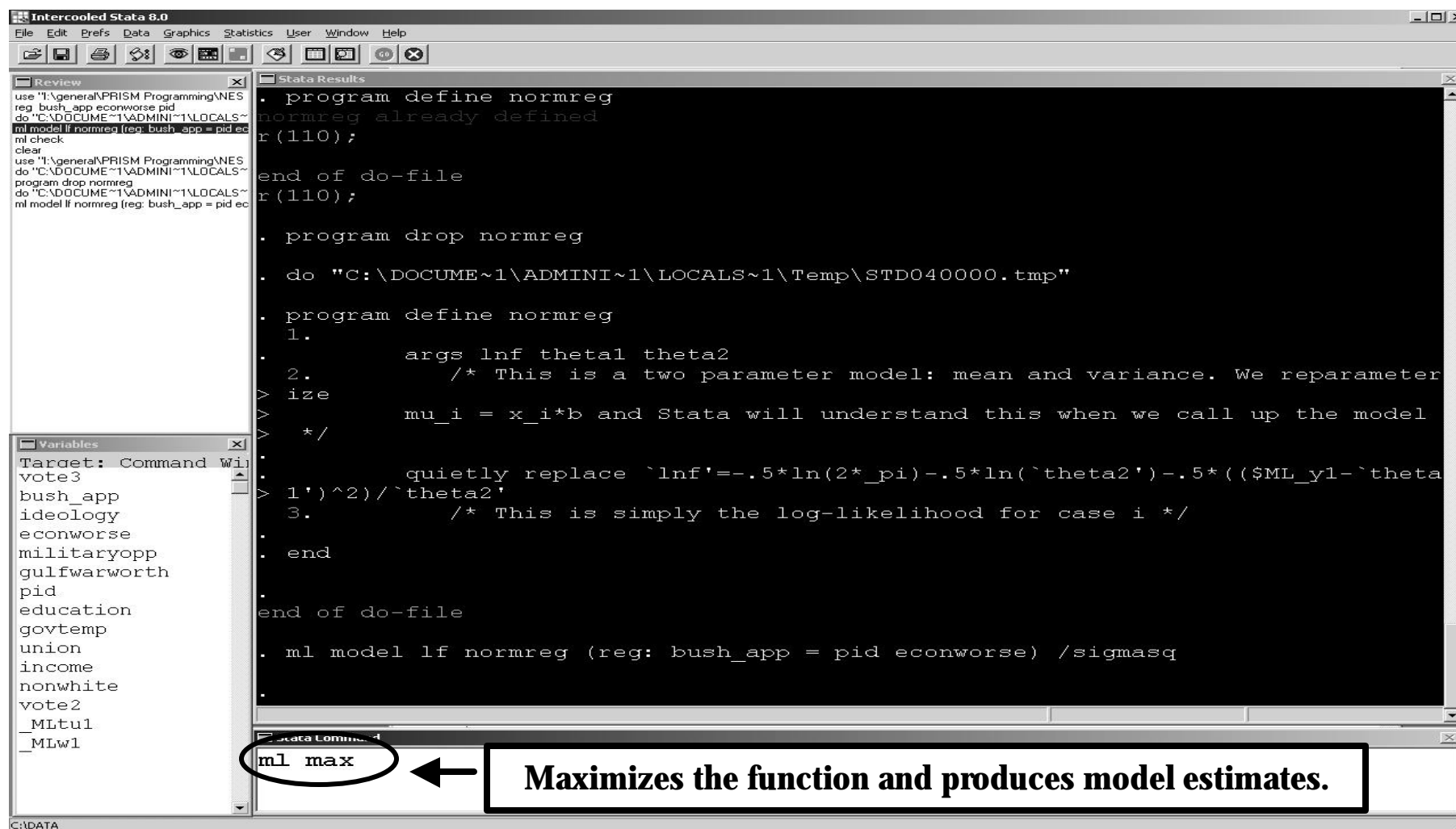
The left sidebar shows the Variables window with the following list:

- Target: Command Window
- vote3
- bush_app
- ideology
- econworse
- militaryopp
- gulfwarworth
- pid
- education
- govtemp
- union
- income
- nonwhite
- vote2

The bottom status bar shows the command:

```
ml model lf normreg (reg: bush_app = pid econworse) /sigmasq
```

Programming Likelihood Functions: Estimating the ML Normal Model



The screenshot displays the Stata 8.0 interface with three main windows: Review, Variables, and Stata Results. The Review window shows the command history, including the definition of the `normreg` program. The Variables window lists the data variables. The Stata Results window shows the execution of the `ml max` command, which maximizes the likelihood function and produces model estimates.

```
use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
```

Target: Command Wi
vote3
bush_app
ideology
econworse
militaryopp
gulfworworth
pid
education
govtemp
union
income
nonwhite
vote2
_MLt1
_MLw1

```
. program define normreg
normreg already defined
r(110);

end of do-file
r(110);

. program drop normreg

. do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"

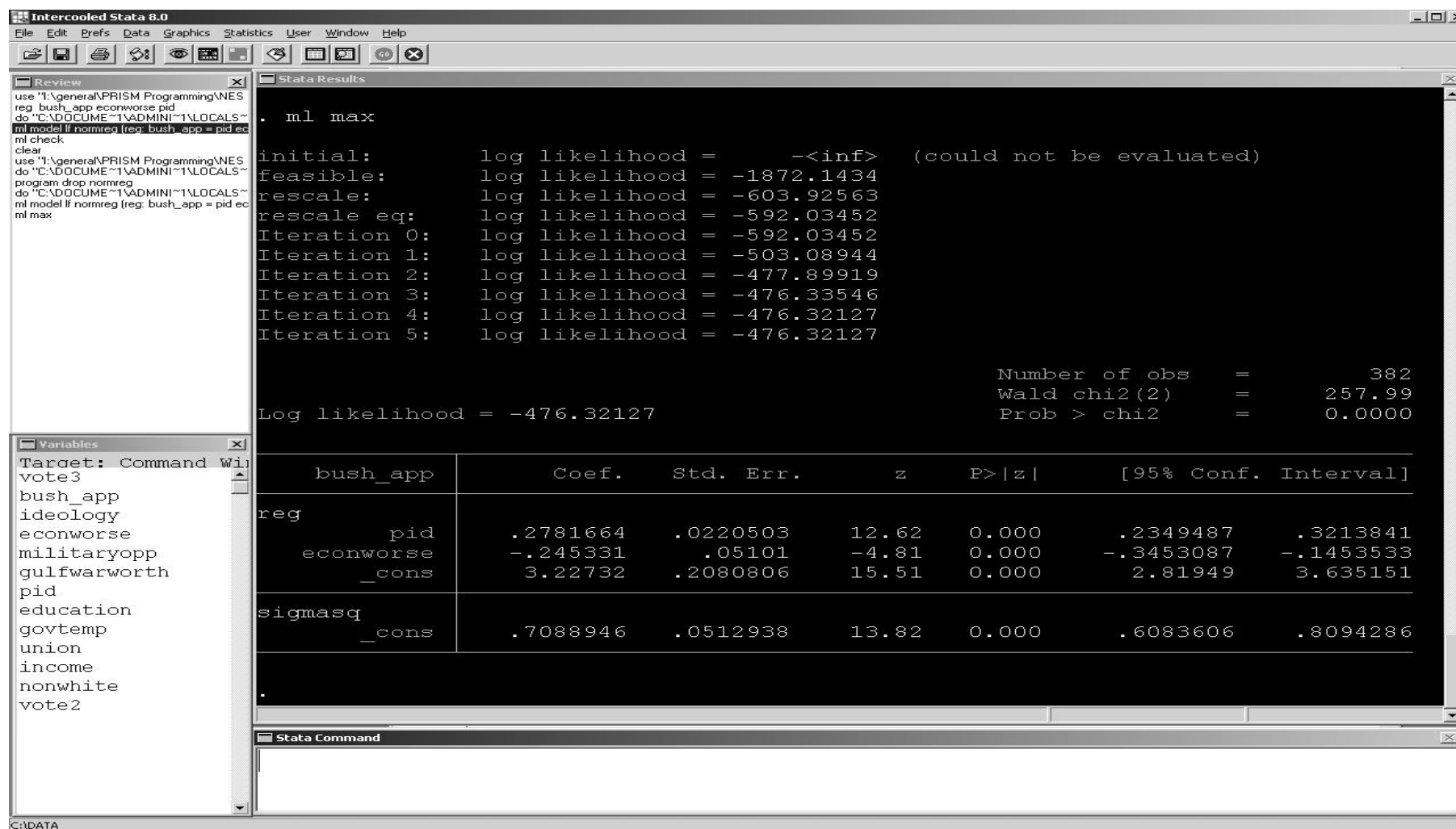
. program define normreg
1.
.     args lnf theta1 theta2
2.     /* This is a two parameter model: mean and variance. We reparameter
> ize
>     mu_i = x_i*b and Stata will understand this when we call up the model
> */
.
.     quietly replace `lnf'=-.5*ln(2*_pi)-.5*ln(`theta2')-.5*(($ML_y1-`theta
> 1')^2)/`theta2'
3.     /* This is simply the log-likelihood for case i */
. end
. end of do-file

. ml model lf normreg (reg: bush_app = pid econworse) /sigmasq

. ml max
```

Maximizes the function and produces model estimates.

Programming Likelihood Functions: Estimating the ML Normal Model



```
use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
```

```
. ml max

initial:      log likelihood =      -<inf>      (could not be evaluated)
feasible:     log likelihood = -1872.1434
rescale:      log likelihood = -603.92563
rescale eq:   log likelihood = -592.03452
Iteration 0:   log likelihood = -592.03452
Iteration 1:   log likelihood = -503.08944
Iteration 2:   log likelihood = -477.89919
Iteration 3:   log likelihood = -476.33546
Iteration 4:   log likelihood = -476.32127
Iteration 5:   log likelihood = -476.32127

Log likelihood = -476.32127

Number of obs      =          382
Wald chi2(2)       =        257.99
Prob > chi2        =          0.0000
```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
bush_app						
reg						
pid	.2781664	.0220503	12.62	0.000	.2349487	.3213841
econworse	-.245331	.05101	-4.81	0.000	-.3453087	-.1453533
_cons	3.22732	.2080806	15.51	0.000	2.81949	3.635151
sigmasq						
_cons	.7088946	.0512938	13.82	0.000	.6083606	.8094286

```
. 
```

Programming Likelihood Functions: Comparing ML Reg to OLS

- OLS and ML Normal Regression produce identical parameter estimates. It can be shown that the analytical solution for ML Normal Regression is:

$$\beta = (X'X)^{-1}X'Y$$

which is identical to the well-known formula for the OLS estimator.

- Standard errors will be different, though, because:

- In ML:
$$s^2 = \frac{\sum_{i=1}^N e_i^2}{N}$$

- In OLS:
$$s^2 = \frac{\sum_{i=1}^N e_i^2}{N - k}$$

Programming Likelihood Functions: Comparing ML Reg to OLS

The screenshot shows the Stata 8.0 interface with the following components:

- Review window:** Contains the command sequence:

```
use "I:\general\PRISM Programming\NES"
reg bush_app econworse pid
do "C:\ADOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\reg.bush_app = pid ec
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES"
do "C:\ADOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\reg.bush_app = pid ec
program drop normreg
do "C:\ADOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\reg.bush_app = pid ec
ml model lf normreg (reg: bush_app = pid ec
ml max
```
- Stata Results window:** Displays the output of the `ml max` command.

```
. ml max

initial:      log likelihood =      -<inf>      (could not be evaluated)
feasible:     log likelihood = -1872.1434
rescale:      log likelihood = -603.92563
rescale eq:   log likelihood = -592.03452
Iteration 0:   log likelihood = -592.03452
Iteration 1:   log likelihood = -503.08944
Iteration 2:   log likelihood = -477.89919
Iteration 3:   log likelihood = -476.33546
Iteration 4:   log likelihood = -476.32127
Iteration 5:   log likelihood = -476.32127

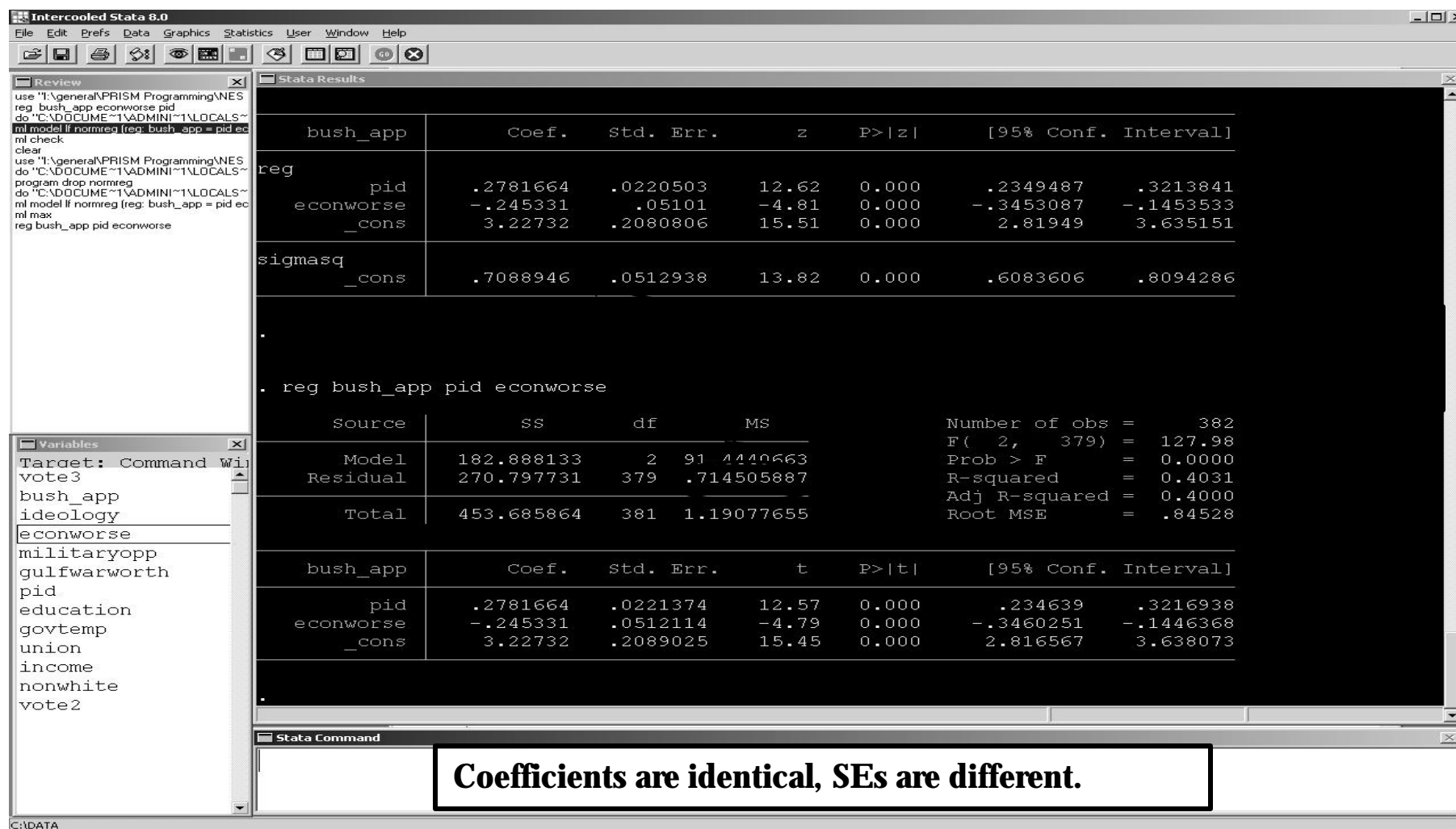
Log likelihood = -476.32127

                                Number of obs   =           382
                                Wald chi2(2)      =           257.99
                                Prob > chi2       =           0.0000
```

	bush_app	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
reg						
	pid	.2781664	.0220503	12.62	0.000	.2349487 .3213841
	econworse	-.245331	.05101	-4.81	0.000	-.3453087 -.1453533
	_cons	3.22732	.2080806	15.51	0.000	2.81949 3.635151
sigmasq						
	_cons	.7088946	.0512938	13.82	0.000	.6083606 .8094286
- Variables window:** Lists available variables: Target: Command Wi, vote3, bush_app, ideology, econworse, militaryopp, gulfwarworth, pid, education, govtemp, union, income, nonwhite, vote2.
- Stata Command window:** Contains the command `reg bush_app pid econworse`, which is circled in red.

An arrow points from the circled command to a box containing the text: **Estimate this OLS.**

Programming Likelihood Functions: Comparing ML Reg to OLS



Programming Likelihood Functions: Logit and Probit

- In binary response models, we want to model the probability of “success” for case i , i.e., $\Pr(y_i=1) = \pi_i$
- We parameterize π_i as a cumulative distribution function (cdf) of a particular distribution, i.e., $F(x_i\beta)$
 - For logit, we use the logistic cdf:

$$\Pr(y_i = 1) = F(x_i\mathbf{b}) = \frac{\exp(x_i\mathbf{b})}{1 + \exp(x_i\mathbf{b})}$$

- For probit, we use the normal cdf:

$$\Pr(y_i = 1) = F(x_i\mathbf{b}) = \Phi(x_i\mathbf{b})$$

Programming Likelihood Functions: Logit and Probit

- The likelihood for case i is:

$$L_i = [F(x_i \mathbf{b})]^{y_i} [1 - F(x_i \mathbf{b})]^{1-y_i}$$

- The log-likelihood for case i is:

$$\ln L_i = y_i \ln[F(x_i \mathbf{b})] + (1 - y_i) \ln[1 - F(x_i \mathbf{b})]$$

****Again, this is what we're going to give Stata****

- For logit, we'll replace $F(x_i \beta)$ with the logistic cdf, and for probit, the normal cdf.

Programming Likelihood Functions: Logit and Probit

- Open “mylogit.do” from the .do file editor.
 - Go to the I: drive
 - Double-click on “general”
 - Double-click on “PRISM Programming”
 - Double-click on “mylogit.do”

Programming Likelihood Functions: Logit and Probit

```
mylogit.do - Stata Do-file Editor
File Edit Search Tools

program define mylogit

    args lnf theta1
    /* only one parameter here, pi, which we reparameterize as x_i*b */

    tempvar p
    /* creating this temp var makes organization a little easier. Here, "p" is going to
    stand for F(xb), i.e., the logistic cdf */

    quietly gen double `p'=exp(`theta1')/(1+exp(`theta1'))
    /* we define the F(xb) as the logistic cdf */

    quietly replace `lnf'=$ML_y1*ln(`p')+(1-$ML_y1)*ln(1-`p')
    /* now, we just put `p' in where the cdf is defined in the lnL */

end
```

Run the program

Recall:

$$\ln L_i = y_i \ln[F(x_i \mathbf{b})] + (1 - y_i) \ln[1 - F(x_i \mathbf{b})]$$

Line number: 8

Programming Likelihood Functions: Logit and Probit

The screenshot shows the Stata 8.0 interface with the following components:

- Review window:** Displays the command sequence used for estimation:

```
use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\DOCUME~1\ADMINI~1\LOCALS~
```
- Stata Results window:** Shows the results of the maximum likelihood estimation.

	Total				Root MSE	
	453.685864	381	1.19077655			= .84528
bush_app	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pid	.2781664	.0221374	12.57	0.000	.234639	.3216938
econworse	-.245331	.0512114	-4.79	0.000	-.3460251	-.1446368
_cons	3.22732	.2089025	15.45	0.000	2.816567	3.638073
- Variables window:** Lists the variables in the dataset: Target, Command, W, vote3, bush_app, ideology, econworse, militaryopp, gulfwarworth, pid, education, govttemp, union, income, nonwhite, vote2.
- Command window:** Contains the Stata command to estimate the logit model:

```
ml model lf mylogit (logit: vote2 = pid econworse)
```

This command is circled in the image.

Programming Likelihood Functions: Logit and Probit

Stata Results

	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
bush_app					
pid	.2781664	.0221374	12.57	0.000	.234639 .3216938
econworse	-.245331	.0512114	-4.79	0.000	-.3460251 -.1446368
_cons	3.22732	.2089025	15.45	0.000	2.816567 3.638073

Review

```
use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf mylogit (logit: vote2 = pid econw
```

Variables

Target: Command Win
vote3
bush_app
ideology
econworse
militaryopp
gulfwarworth
pid
education
govtemp
union
income
nonwhite
vote2
_MLtu1
_MLw1

Stata Command

```
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"
. program define mylogit
1.
    args lnf theta1
2.
    /* only one parameter here, pi, which we reparameterize as x_i*b */
.
    tempvar p
3.
    /* creating this temp var makes organization a little easier. Here, "p" is going
> to stand for F(xb), i.e., the logistic cdf */
.
    quietly gen double `p'=exp(`theta1')/(1+exp(`theta1'))
4.
    /* we define F(xb) as the logistic cdf */
.
    quietly replace `lnf'=$ML_y1*ln(`p')+(1-$ML_y1)*ln(1-`p')
5.
    /* now, we just put `p' in where the cdf is defined in the lnL */
.
. end
. end of do-file
. ml model lf mylogit (logit: vote2 = pid econworse)
```

ml max ← Maximizes the function and produces model estimates.

Programming Likelihood Functions: Logit and Probit

Review

```
use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\DOCCUME~1\ADMINI~1\LOCALS~
ml model lf mylogit (logit: vote2 = pid econw
ml max
```

Stata Results

```
. end
.
.
end of do-file
. ml model lf mylogit (logit: vote2 = pid econworse)
. ml max

initial:      log likelihood = -264.78222
alternative:  log likelihood = -254.09741
rescale:      log likelihood = -254.09741
Iteration 0:  log likelihood = -254.09741
Iteration 1:  log likelihood = -166.21063
Iteration 2:  log likelihood = -162.49286
Iteration 3:  log likelihood = -162.45214
Iteration 4:  log likelihood = -162.45212
Iteration 5:  log likelihood = -162.45212

Log likelihood = -162.45212

Number of obs   =      382
Wald chi2(2)    =    105.87
Prob > chi2     =     0.0000
```

	vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
	pid	.8437178	.0878516	9.60	0.000	.6715318 1.015904
	econworse	-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
	_cons	.2923819	.6554855	0.45	0.656	-.992346 1.57711

Variables

Target: Command Window

vote3

bush_app

ideology

econworse

militaryopp

gulfworworth

pid

education

govtemp

union

income

nonwhite

vote2

Stata Command

C:\DATA

Programming Likelihood Functions: Logit and Probit

Intercooled Stata 8.0

File Edit Prefs Data Graphics Statistics User Window Help

Review

```
use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\ADOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\ADOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\ADOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\ADOCUME~1\ADMINI~1\LOCALS~
ml model lf mylogit (logit: vote2 = pid econw
ml max
logit vote2 pid econworse
```

Stata Results

```
Log likelihood = -162.45212
Wald chi2(2) = 105.87
Prob > chi2 = 0.0000
```

	vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
pid		.8437178	.0878516	9.60	0.000	.6715318 1.015904
econworse		-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
_cons		.2923819	.6554855	0.45	0.656	-.992346 1.57711

```
. logit vote2 pid econworse

Iteration 0: log likelihood = -254.0798
Iteration 1: log likelihood = -169.39683
Iteration 2: log likelihood = -162.84667
Iteration 3: log likelihood = -162.45449
Iteration 4: log likelihood = -162.45212
Iteration 5: log likelihood = -162.45212

Logit estimates
Number of obs = 382
LR chi2(2) = 183.26
Prob > chi2 = 0.0000
Pseudo R2 = 0.3606

Log likelihood = -162.45212
```

	vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
pid		.8437178	.0878516	9.60	0.000	.6715318 1.015904
econworse		-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
_cons		.2923819	.6554855	0.45	0.656	-.992346 1.57711

Variables

Target: Command Window

vote3

bush_app

ideology

econworse

militaryopp

gulfwarworth

pid

education

govtemp

union

income

nonwhite

vote2

Stata Command

C:\DATA

Programming Likelihood Functions: The Likelihood Ratio Test

- By default, “ml model” produces a Wald test for overall goodness of fit test (which tests that the coefficients are jointly equal to zero).
- To get an LR test, we need to:
 - Estimate an intercept only model to get LL_0 , the initial LL.
 - We need to specify k for the intercept-only model, which in this case is 1.
 - After the “ml model” command, we enter $lf0(k\ LL_0)$.

Programming Likelihood Functions: The Likelihood Ratio Test

The screenshot displays the Stata 8.0 interface with the following components:

- Review Panel:** Shows the command sequence used for the logit model:


```
use "I:\general\PRISM Programming\NES"
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\do"
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES"
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\do"
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\do"
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\do"
ml model lf mylogit (logit: vote2 = pid econw
ml max
logit vote2 pid econworse
```
- Stata Results Panel:**
 - Log likelihood = -162.45212
 - Wald chi2(2) = 105.87
 - Prob > chi2 = 0.0000
 - Iteration log likelihood values from 0 to 5.
 - Logit estimates: Number of obs = 382, LR chi2(2) = 183.26, Prob > chi2 = 0.0000, Pseudo R2 = 0.3606.
 - Log likelihood = -162.45212
 - Table of coefficients for the logit model:
- Stata Command Panel:** Shows the command: `ml model lf mylogit (logit: vote2 =)`

vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
pid	.8437178	.0878516	9.60	0.000	.6715318 1.015904
econworse	-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
_cons	.2923819	.6554855	0.45	0.656	-.992346 1.57711

Estimates intercept only model

Programming Likelihood Functions: The Likelihood Ratio Test

The screenshot displays the Intercooled Stata 8.0 interface. The **Stata Results** window shows the output of a logit model. The **Command** window shows the commands entered to fit the model and perform a likelihood ratio test.

Stata Results Window:

```

Log likelihood = -162.45212
LR chi2(2)      =      183.26
Prob > chi2     =      0.0000
Pseudo R2      =      0.3606
  
```

	vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
	pid	.8437178	.0878516	9.60	0.000	.6715318 1.015904
	econworse	-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
	_cons	.2923819	.6554855	0.45	0.656	-.992346 1.57711

```

. ml model lf mylogit (logit: vote2 =)
. ml max

initial:      log likelihood = -264.78222
alternative:  log likelihood = -254.09741
rescale:      log likelihood = -254.09741
Iteration 0:  log likelihood = -254.09741
Iteration 1:  log likelihood = -254.07998
Iteration 2:  log likelihood = -254.07998

Log likelihood = -254.0798
Number of obs   =      382
Wald chi2(0)    =      .
Prob > chi2     =      .
  
```

Command Window:

```

use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf mylogit (logit: vote2 = pid econw
ml max
logit vote2 pid econworse
ml model lf mylogit (logit: vote2 =)
ml max
  
```

Variables Window:

Target: Command Window

vote3

bush_app

ideology

econworse

militaryopp

gulfwarworth

pid

education

govtemp

union

income

nonwhite

vote2

Programming Likelihood Functions: The Likelihood Ratio Test

The screenshot shows the Stata 8.0 interface with the following components:

- Review Window:** Contains the Stata command script. The relevant part for the first model is:


```
use "I:\general\PRISM Programming\NES"
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\do"
ml model lf normreg (reg: bush_app = pid econworse)
ml check
clear
use "I:\general\PRISM Programming\NES"
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\do"
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\do"
ml model lf normreg (reg: bush_app = pid econworse)
ml max
reg bush_app pid econworse
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\temp\1\stata\do"
ml model lf mylogit (logit: vote2 = pid econworse)
ml max
logit vote2 pid econworse
ml model lf mylogit (logit: vote2 =)
ml max
```
- Variables Window:** Lists the variables in the dataset: Target: Command Window, vote3, bush_app, ideology, econworse, militaryopp, gulfwarworth, pid, education, govttemp, union, income, nonwhite, and vote2.
- Stata Results Window:** Displays the results of the first model (normreg).

Log likelihood = -162.45212

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
pid	.8437178	.0878516	9.60	0.000	.6715318 1.015904
econworse	-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
_cons	.2923819	.6554855	0.45	0.656	-.992346 1.57711

LR chi2(2) = 183.26
Prob > chi2 = 0.0000
Pseudo R2 = 0.3606
- Stata Command Window:** Shows the command for the second model:


```
ml model lf mylogit (logit: vote2 =)
ml max
```
- Stata Results Window (continued):** Displays the results of the second model (mylogit).

initial: log likelihood = -264.78222
alternative: log likelihood = -254.09741
rescale: log likelihood = -254.09741
Iteration 0: log likelihood = -254.09741
Iteration 1: log likelihood = -254.0798
Iteration 2: log likelihood = -254.0798

Log likelihood = -254.0798

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
_cons	-.4802252	.1052929	-4.56	0.000	-.6865956 -.2738548

Number of obs = 382
Wald chi2(0) = .
Prob > chi2 = .

Programming Likelihood Functions: The Likelihood Ratio Test

The screenshot displays the Intercooled Stata 8.0 interface. The main window shows the results of a logit model estimation. The command window on the left contains the following code:

```

use "I:\general\PRISM Programming\NES
reg bush_app econworse pid
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml check
clear
use "I:\general\PRISM Programming\NES
do "C:\DOCUME~1\ADMINI~1\LOCALS~
program drop normreg
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf normreg (reg: bush_app = pid ec
ml max
reg bush_app pid econworse
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf mylogit (logit: vote2 = pid econw
ml max
logit vote2 pid econworse
ml model lf mylogit (logit: vote2 =)
ml max
ml model lf mylogit (logit: vote2 = pid econw
ml max

```

The main window displays the following results:

vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
_cons	-.4802252	.1052929	-4.56	0.000	-.6865956 -.2738548

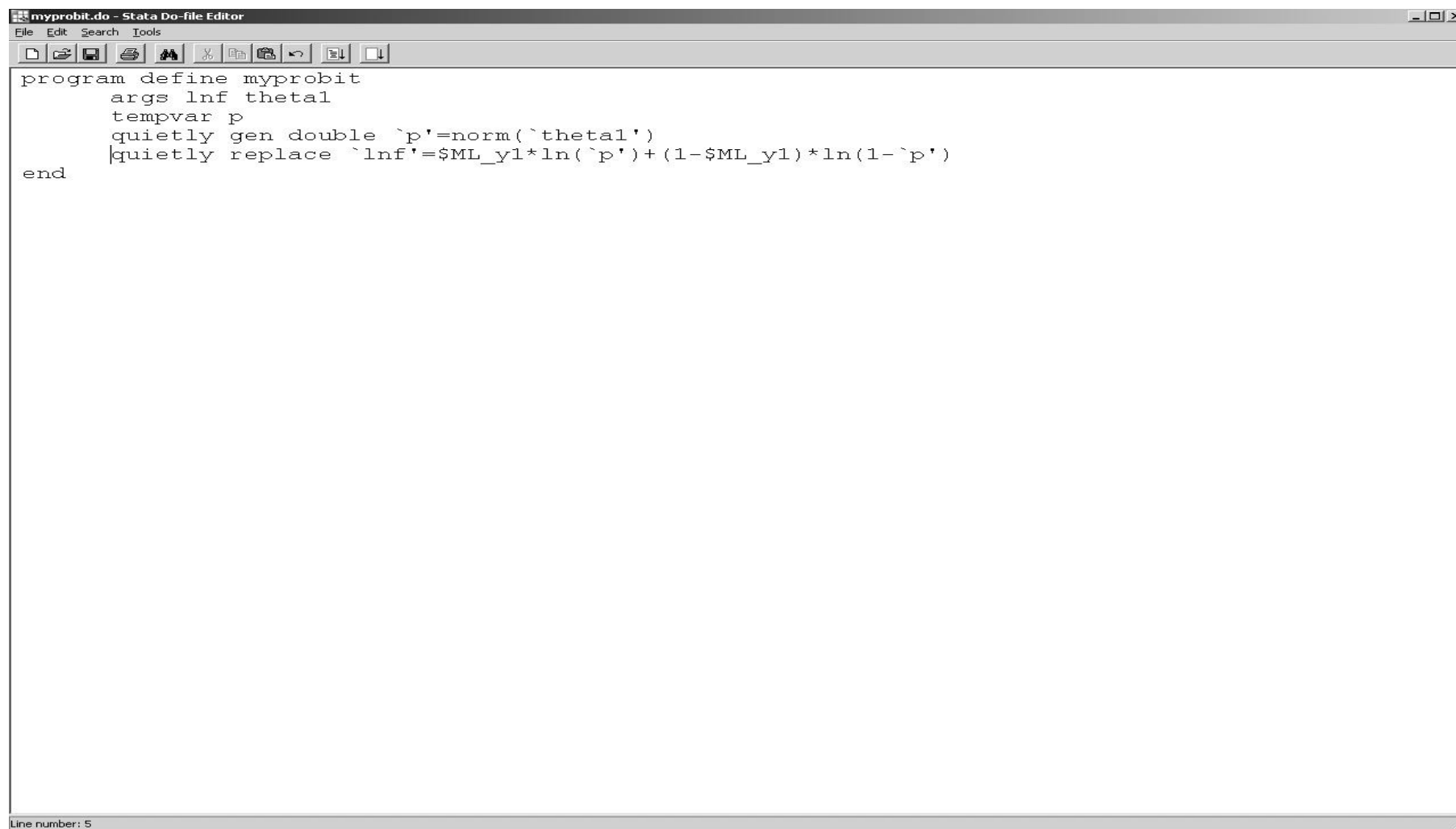
Log likelihood = -162.45212

Number of obs = 382
LR chi2(2) = 183.26
Prob > chi2 = 0.0000

vote2	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
pid	.8437178	.0878516	9.60	0.000	.6715318 1.015904
econworse	-.3076029	.1606638	-1.91	0.056	-.6224982 .0072923
_cons	.2923819	.6554855	0.45	0.656	-.992346 1.57711

The command window at the bottom is empty.

Programming Likelihood Functions: Probit Program



The screenshot shows the Stata Do-file Editor window titled "myprobit.do - Stata Do-file Editor". The window has a menu bar with "File", "Edit", "Search", and "Tools". Below the menu bar is a toolbar with icons for file operations (new, open, save, print, etc.) and editing (undo, redo, cut, paste, etc.). The main text area contains the following Stata code:

```
program define myprobit
  args lnf theta1
  tempvar p
  quietly gen double `p' = norm(`theta1')
  quietly replace `lnf' = $ML_y1 * ln(`p') + (1 - $ML_y1) * ln(1 - `p')
end
```

At the bottom of the window, the status bar indicates "Line number: 5".

Programming Likelihood Functions: Heteroskedastic Regression

- Heteroskedastic regression allows us to model the factors that influence both the expected value of Y *and* the factors that affect the *variability* around that expected value (see Franklin 1991; Alvarez and Brehm 1995, 1997, 1998).
- In regression, we always assume homoskedasticity: s^2
- With het. reg., we're explicitly interested in modeling the factors that influence s_i^2 .
- Good pedagogical example: it's more complicated, it generates two sets of simultaneously generated coefficients. *But*, bottom line: all you have to do is know the likelihood function, and you can program it in Stata.

Programming Likelihood Functions: Heteroskedastic Regression

- We parameterize σ_i^2 as:

$$\sigma_i^2 = e^{z_i \mathbf{g}}$$

- Where the z_i 's exogenous variables that influence the variability around the expected value, and gamma is a vector of parameters.
- Log-likelihood:

$$\ln L_i = -\frac{1}{2} \ln(2\mathbf{p}) - \frac{1}{2} z_i \mathbf{g} - \frac{1}{2} \left[\frac{(y_i - x_i \mathbf{b})^2}{e^{z_i \mathbf{g}}} \right]$$

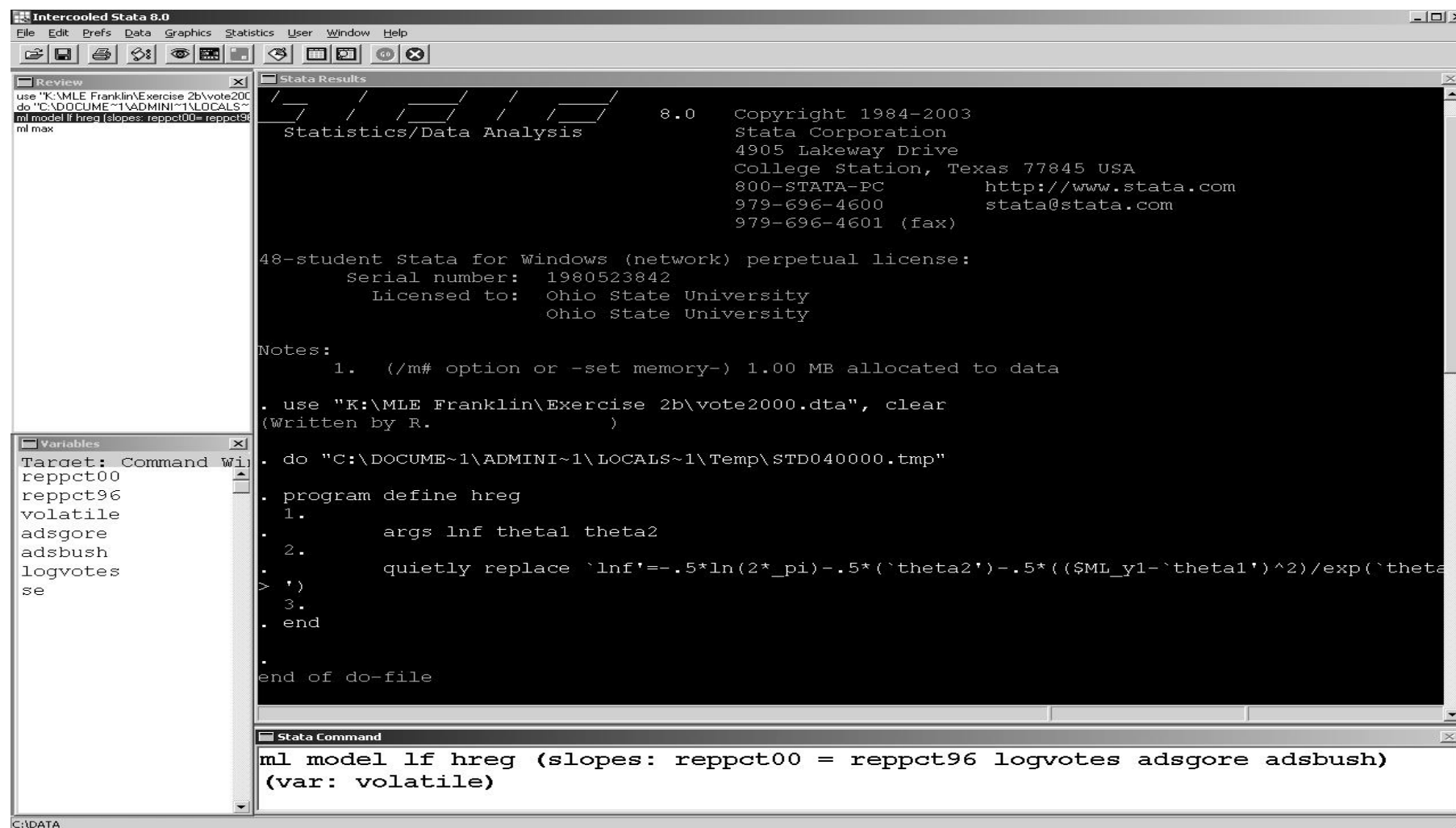
Programming Likelihood Functions: Heteroskedastic Regression

```
hetreg.do - Stata Do-file Editor
File Edit Search Tools
[Icons]
program define hreg
    args lnf theta1 theta2
    quietly replace `lnf'=-.5*ln(2*_pi)-.5*(`theta2')-.5* (($ML_y1-`theta1')^2)/exp(`theta2')
end
```

$$\ln L_i = -\frac{1}{2} \ln(2p) - \frac{1}{2} z_i \mathbf{g} - \frac{1}{2} \left[\frac{(y_i - x_i \mathbf{b})^2}{e^{z_i \mathbf{g}}} \right]$$

Line number: 8

Programming Likelihood Functions: Heteroskedastic Regression



The screenshot shows the Stata 8.0 software interface. The main window displays the Stata startup screen with the text "Statistics/Data Analysis 8.0" and copyright information. The "Review" window on the left shows the do-file content, and the "Variables" window below it lists the variables used in the model. The "Stata Command" window at the bottom shows the command being executed.

```
use "K:\MLE Franklin\Exercise 2b\vote2000.dta", clear
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"
ml model lf hreg (slopes: reppct00 = reppct96 logvotes adsgore adsbush)
ml max
```

Variables:

- Target: Command Window
- reppct00
- reppct96
- volatile
- adsgore
- adsbush
- logvotes
- se

Stata Command:

```
ml model lf hreg (slopes: reppct00 = reppct96 logvotes adsgore adsbush)
(var: volatile)
```

Programming Likelihood Functions: Heteroskedastic Regression

The screenshot displays the Stata 8.0 interface with the following components:

- Review window:** Shows the command `ml model lf hreg (slopes: reppct00= reppct96) ml max`.
- Stata Results window:** Displays the iterative log-likelihood values and the final model fit statistics.
- Variables window:** Lists the variables used in the model: `reppct00`, `reppct96`, `volatile`, `adsgore`, `adsbush`, `logvotes`, and `se`.
- Stata Command window:** Is empty.

Stata Results Output:

```
Iteration 2: log likelihood = -4458.1617 (not concave)
Iteration 3: log likelihood = -4123.6179 (not concave)
Iteration 4: log likelihood = -3878.4709 (not concave)
Iteration 5: log likelihood = -3779.2134 (not concave)
Iteration 6: log likelihood = -3713.0274 (not concave)
Iteration 7: log likelihood = -3667.38 (not concave)
Iteration 8: log likelihood = -3627.8335 (not concave)
Iteration 9: log likelihood = -3590.76 (not concave)
Iteration 10: log likelihood = -3439.2212 (backed up)
Iteration 11: log likelihood = -3188.4392
Iteration 12: log likelihood = -3184.6605
Iteration 13: log likelihood = -3184.6537
Iteration 14: log likelihood = -3184.6537

Log likelihood = -3184.6537

Number of obs = 1184
Wald chi2(4) = 10562.78
Prob > chi2 = 0.0000
```

Model Coefficients:

	reppct00	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
slopes						
reppct96		.9604168	.0103549	92.75	0.000	.9401216 .9807121
logvotes		-4.795379	.1706207	-28.11	0.000	-5.129789 -4.460968
adsgore		-.6220649	.1298877	-4.79	0.000	-.8766402 -.3674896
adsbush		1.057464	.165257	6.40	0.000	.7335665 1.381362
_cons		29.13514	.9552502	30.50	0.000	27.26288 31.00739
var						
volatile		.0818051	.0174115	4.70	0.000	.0476792 .1159309
_cons		2.278007	.0695479	32.75	0.000	2.141696 2.414319

Programming Likelihood Functions: Split-Population Duration Model

- Standard duration models, which model the hazard of an event occurring, assume that all cases will eventually experience the event of interest.
- This assumption may not hold for the process under examination; if not, will produce incorrect inferences.
 - The timing of congressional overrides of Supreme Court decisions (Hettinger and Zorn 2004).
 - Corporate and labor PAC contributions to congressional candidates (Box-Steffensmeier et al. 2004).

Programming Likelihood Functions: Split-Population Duration Model

- The split population duration (SPD) model relaxes the assumption that all cases will eventually experience the event of interest (Schmidt and Witte 1988, 1989; Forster and A. Jones 2001; Box-Steffensmeier and B. Jones 2004).
- Simultaneously estimates two sets of coefficients:
 1. Explaining the likelihood of the event occurring (i.e., the censoring indicator is the DV).
 2. Explaining the timing of the event occurring, conditional on the event having occurred in the first place.

Programming Likelihood Functions: Split-Population Duration Model

- LIMDEP is the only package that has a canned routine for the SPD. Great example of an advanced model that hasn't made its way into a lot of stat packages. But you can program it yourself!
- Acknowledgements to Forster and Jones...

Programming Likelihood Functions: Split-Population Duration Model

```
splitpop.do - Stata Do-file Editor
File Edit Search Tools

program define splitpop

    args lnf theta1 theta2 theta3
    /* theta1: XB for duration equation; theta2: XB for logit equation;
    theta3: shape parameter (delta) */

    tempvar p s d l

    quietly gen double `l'=exp(-`theta1')
    /* lambda of log-logistic distribution */

    quietly gen double `d'=exp(`theta2')/(1+exp(`theta2'))
    /* logistic cdf for probability of failure */

    quietly gen double `s'=1-`d'+`d'*(1/(1+(`l'*$ML_y1)^(1/`theta3'))))
    /* survival function */

    quietly gen double `p'=ln(`d')-ln(`theta3')+((1/`theta3')-1)*ln($ML_y1)+(1/`theta3')*
    /* ln(`l')-2*ln(1+(`l'*$ML_y1)^(1/`theta3')) */
    /* pdf */

    quietly replace `lnf'=$ML_y2*(`p')+(1-$ML_y2)*ln(`s')

end
```

$$\ln L_i = R_i [\ln d_i + \ln g(t_i, q)] + (1 - R_i) \ln [1 - d_i + d_i G(t_i, q)]$$

censoring indicator

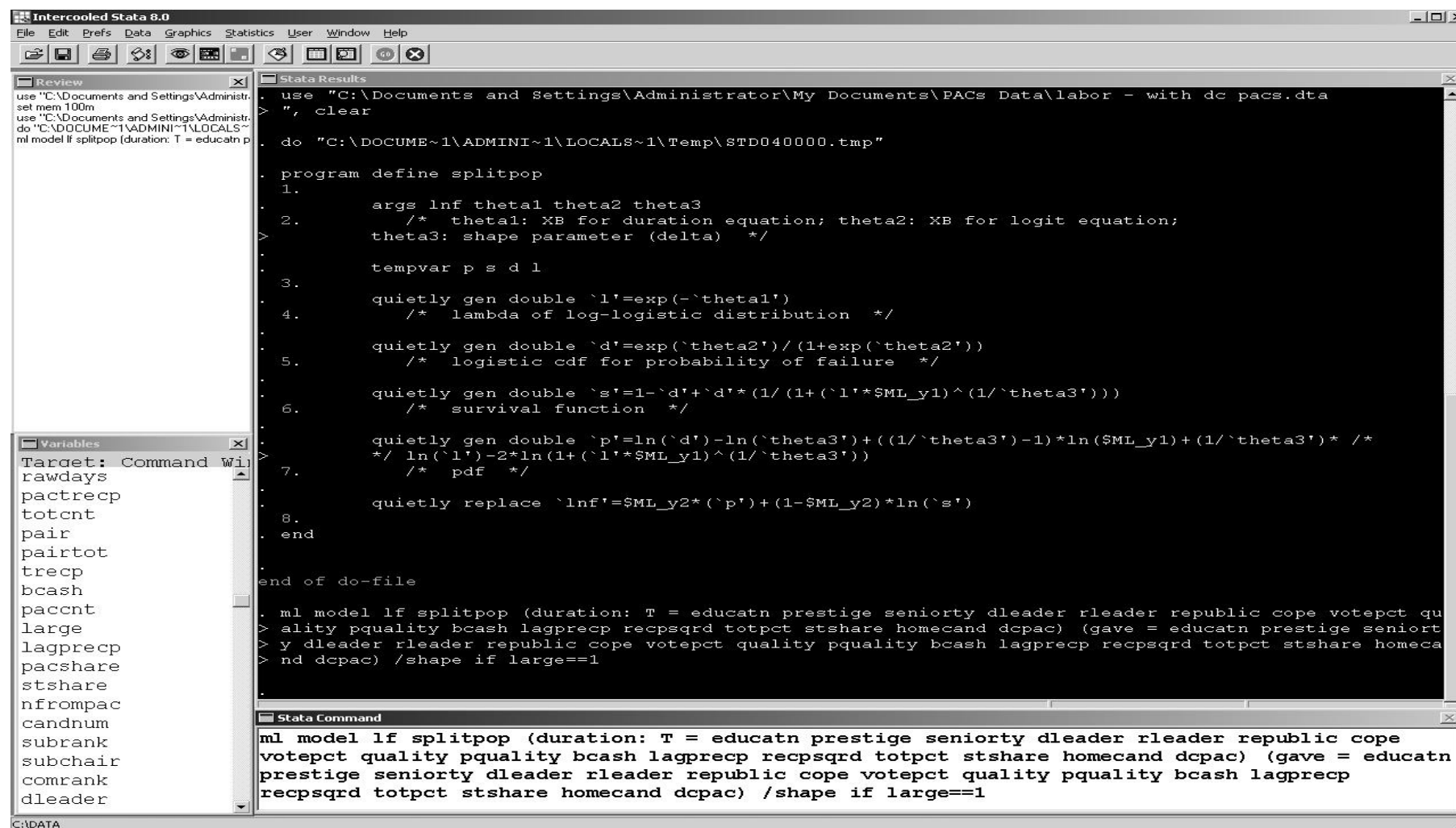
pdf

survivor function

Programming Likelihood Functions: Split-Population Duration Model

- Example: Explaining the incidence and timing of labor PAC contributions to incumbent House members, 1993-1994 (Box-Steffensmeier et al. 2004).
- We're interested in the timing of contributions in an election cycle. Early money is “seed money” for a campaign effort, and it helps candidates raise more down the line (Jacobson 1992).
- We don't expect labor PACs to contribute to every House incumbent, though. E.g., people trying to reform OSHA, or investigating the Teamsters.

Programming Likelihood Functions: Split-Population Duration Model



```
use "C:\Documents and Settings\Administr...
set mem 100m
use "C:\Documents and Settings\Administr...
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"

. program define splitpop
1.
    args lnf theta1 theta2 theta3
    /* theta1: XB for duration equation; theta2: XB for logit equation;
    theta3: shape parameter (delta) */
2.
    tempvar p s d l
3.
    quietly gen double `l'=exp(-`theta1')
    /* lambda of log-logistic distribution */
4.
    quietly gen double `d'=exp(`theta2')/(1+exp(`theta2'))
    /* logistic cdf for probability of failure */
5.
    quietly gen double `s'=1-`d'+`d'*(1/(1+(`l'*$ML_y1)^(1/`theta3'))))
    /* survival function */
6.
    quietly gen double `p'=ln(`d')-ln(`theta3')+((1/`theta3')-1)*ln($ML_y1)+(1/`theta3')* /*
    /* ln(`l')-2*ln(1+(`l'*$ML_y1)^(1/`theta3')) */
7.
    /* pdf */
8.
    quietly replace `lnf'=$ML_y2*(`p')+(1-$ML_y2)*ln(`s')
9.
end

. end of do-file

. ml model lf splitpop (duration: T = educatn prestige seniorty dleader rleader republic cope voteptct qu
> ality pquality bcash lagprecp recpsqrd totptct stshare homecand dcpac) (gave = educatn prestige seniort
> y dleader rleader republic cope voteptct quality pquality bcash lagprecp recpsqrd totptct stshare homeca
> nd dcpac) /shape if large==1

. ml model lf splitpop (duration: T = educatn prestige seniorty dleader rleader republic cope
voteptct quality pquality bcash lagprecp recpsqrd totptct stshare homecand dcpac) (gave = educatn
prestige seniorty dleader rleader republic cope voteptct quality pquality bcash lagprecp
recpsqrd totptct stshare homecand dcpac) /shape if large==1
```

Programming Likelihood Functions: Split-Population Duration Model

```
Intercooled Stata 8.0
File Edit Refs Data Graphics Statistics User Window Help

Review
use "C:\Documents and Settings\Administr...
set mem 100m
use "C:\Documents and Settings\Administr...
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"
ml model lf splitpop (duration: T = educatn p

Stata Results
. use "C:\Documents and Settings\Administrator\My Documents\PACs Data\labor - with dc pacs.dta
> ", clear

. do "C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\STD040000.tmp"

. program define splitpop
1.
    args lnf theta1 theta2 theta3
    /* theta1: XB for duration equation; theta2: XB for logit equation;
    theta3: shape parameter (delta) */

    tempvar p s d l

    quietly gen double `l'=exp(-`theta1')
    /* lambda of log-logistic distribution */

    quietly gen double `d'=exp(`theta2')/(1+exp(`theta2'))
    /* logistic cdf for probability of failure */

    quietly gen double `s'=1-`d'+`d'*(1/(1+(`l'*$ML_y1)^(1/`theta3'))))
    /* survival function */

    quietly gen double `p'=ln(`d')-ln(`theta3')+((1/`theta3')-1)*ln($ML_y1)+(1/`theta3')* /*
    /* ln(`l')-2*ln(1+(`l'*$ML_y1)^(1/`theta3')) */
    /* pdf */

    quietly replace `lnf'=$ML_y2*(`p')+(1-$ML_y2)*ln(`s')

2.
. end

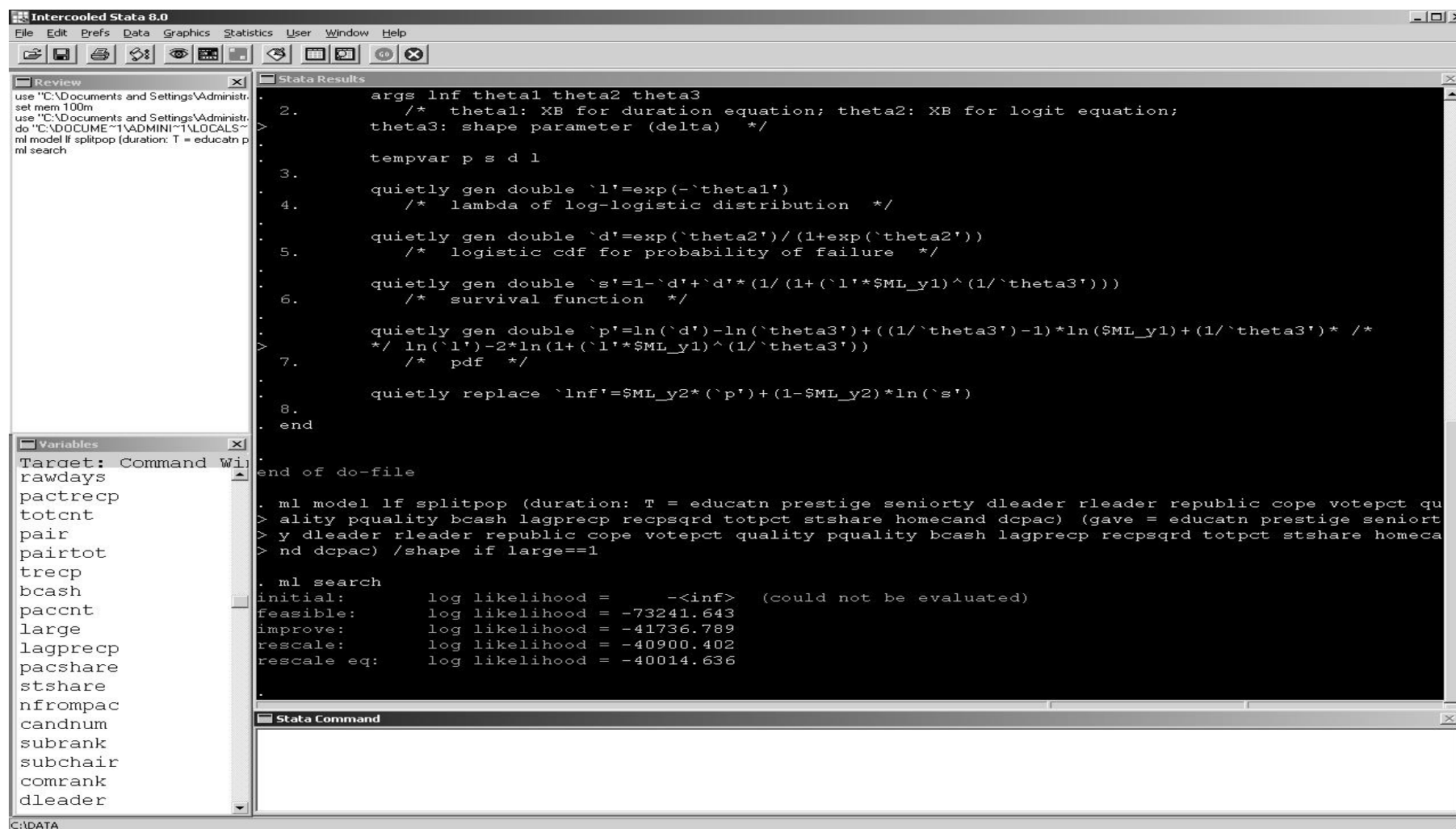
. end of do-file

. ml model lf splitpop (duration: T = educatn prestige seniorty dleader rleader republic cope votept qu
> ality pquality bcash lagprecp recpsqrd totptct stshare homecand dcpac) (gave = educatn prestige seniort
> y dleader rleader republic cope votept quality pquality bcash lagprecp recpsqrd totptct stshare homeca
> nd dcpac) /shape if large==1

Stata Command
ml search

Searches for better starting values
```

Programming Likelihood Functions: Split-Population Duration Model



```
use "C:\Documents and Settings\Administr...
set mem 100m
use "C:\Documents and Settings\Administr...
do "C:\DOCU...
ml model lf splitpop (duration: T = educatn p
ml search

args lnf theta1 theta2 theta3
/* theta1: XB for duration equation; theta2: XB for logit equation;
theta3: shape parameter (delta) */

tempvar p s d l

quietly gen double `l'=exp(-`theta1')
/* lambda of log-logistic distribution */

quietly gen double `d'=exp(`theta2')/(1+exp(`theta2'))
/* logistic cdf for probability of failure */

quietly gen double `s'=1-`d'+`d'*(1/(1+(`l'*$ML_y1)^(1/`theta3'))))
/* survival function */

quietly gen double `p'=ln(`d')-ln(`theta3')+(1/`theta3')-1)*ln($ML_y1)+(1/`theta3')* /*
*/ ln(`l')-2*ln(1+(`l'*$ML_y1)^(1/`theta3'))
/* pdf */

quietly replace `lnf'=$ML_y2*(`p')+(1-$ML_y2)*ln(`s')

end

end of do-file

. ml model lf splitpop (duration: T = educatn prestige seniorty dleader rleader republic cope voteptct qu
> ality pquality bcash lagprecp recpsqrd totptct stshare homecand dcpac) (gave = educatn prestige seniort
> y dleader rleader republic cope voteptct quality pquality bcash lagprecp recpsqrd totptct stshare homeca
> nd dcpac) /shape if large==1

. ml search
initial:      log likelihood =      -<inf>      (could not be evaluated)
feasible:     log likelihood = -73241.643
improve:      log likelihood = -41736.789
rescale:      log likelihood = -40900.402
rescale eq:   log likelihood = -40014.636
```

Programming Likelihood Functions: Split-Population Duration Model

Intercooled Stata 8.0

File Edit Refs Data Graphics Statistics User Window Help

Review

use "C:\Documents and Settings\Administr...
set mem 100m
use "C:\Documents and Settings\Administr...
do "C:\DOCUME~1\ADMINI~1\LOCALS~1\...
ml model if splitpop (duration: T = educatn T
ml search
ml max
ml model if splitpop (duration: T = educatn T
ml search
ml max

Stata Results

Number of obs = 10168
Wald chi2(17) = 534.57
Prob > chi2 = 0.0000

Log likelihood = -33857.726

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
duration					
educatn	.0657934	.0465943	1.41	0.158	-.0255297 .1571166
prestige	.0375843	.0296507	1.27	0.205	-.0205299 .0956986
seniorty	.2655538	.1936916	1.37	0.170	-.1140749 .6451824
dleader	-.2008474	.0426538	-4.71	0.000	-.2844473 -.1172475
rleader	-.2369234	.1366631	-1.73	0.083	-.5047783 .0309314
republic	-.0596892	.060448	-0.99	0.323	-.178165 .0587866
cope	-.2930568	.108651	-2.70	0.007	-.5060088 -.0801048
votept	.3198829	.1227685	2.61	0.009	.0792612 .5605047
quality	.0164541	.043222	0.38	0.703	-.0682594 .1011677
pquality	.0376763	.0765442	0.49	0.623	-.1123476 .1877002
bcash	.351434	.0716984	4.90	0.000	.2109078 .4919603
lagprecp	-.2789671	.0200107	-13.94	0.000	-.3181872 -.2397469
recpsqrd	.0259331	.0023862	10.87	0.000	.0212563 .03061
totpct	-.5041463	.2263586	-2.23	0.026	-.9478011 -.0604915
stshare	.4819196	.1913304	2.52	0.012	.106919 .8569203
homecand	-.2420607	.1037068	-2.33	0.020	-.4453224 -.0387991
dcpac	.4026476	.0298946	13.47	0.000	.3440552 .4612399
_cons	5.717728	.1143268	50.01	0.000	5.493652 5.941804
logit					
educatn	.1092988	.1184172	0.92	0.356	-.1227946 .3413922
prestige	.2806098	.0635753	4.41	0.000	.1560045 .4052151
seniorty	-.7225828	.3885568	-1.86	0.063	-1.48414 .0389745
dleader	.4483005	.117503	3.82	0.000	.2179989 .6786021
rleader	.2813031	.1753403	1.60	0.109	-.0623576 .6249638
republic	-1.798423	.1000995	-17.97	0.000	-1.994614 -1.602231
cope	3.250226	.2013426	16.14	0.000	2.855602 3.64485
votept	-2.374621	.2654199	-8.95	0.000	-2.894835 -1.854408
quality	.2355944	.0915296	2.57	0.010	.0561997 .414989
pquality	-.4067091	.149639	-2.72	0.007	-.6999962 -.113422
bcash	-.4111929	.1515213	-2.71	0.007	-.7081691 -.1142167
lagprecp	.4141835	.0447867	9.25	0.000	.3264032 .5019639
recpsqrd	-.0333042	.0054234	-6.14	0.000	-.0439338 -.0226746
totpct	3.878042	.4972922	7.80	0.000	2.903368 4.852717
stshare	.8182422	.4648726	1.76	0.078	-.0928913 1.729376
homecand	.298608	.2603486	1.15	0.251	-.2116658 .8088819
dcpac	.0935252	.0631602	1.48	0.139	-.0302664 .2173169
_cons	-1.331899	.2248491	-5.92	0.000	-1.772595 -.8912029
shape					
_cons	.480325	.0074244	64.70	0.000	.4657735 .4948765

C:\DATA

Programming Likelihood Functions: The Predict Command and Est. Split

Intercooled Stata 8.0

File Edit Refs Data Graphics Statistics User Window Help

Review

```
set mem 100m
use "K:\Jan - PACs\DATA\large labor PACs"
do "C:\ADOCUME~1\ADMINI~1\LOCALS~1\APPLIC~1\STATA\BIN\DO"
ml model lf splitpop (duration: T = educatn p
gen T=rawdays
ml model lf splitpop (duration: T = educatn p
ml search
ml max
```

Variables

Target: Command Wi

ignored

signal

first

copecase

rrawdays

previous

homecand

lndays

recpsqrd

republic

senior2

lndays2

sample

dcpac

lagrecp2

recsqrd2

bcash2

T

Stata Results

seniorty	.2655538	.1936917	1.37	0.170	-.114075	.6451826	
dleader	-.2008474	.0426538	-4.71	0.000	-.2844473	-.1172475	
rleader	-.2369235	.1366632	-1.73	0.083	-.5047784	.0309314	
republic	-.0596892	.060448	-0.99	0.323	-.1781651	.0587866	
cope	-.2930568	.108651	-2.70	0.007	-.5060089	-.0801047	
votept	.3198831	.1227685	2.61	0.009	.0792613	.560505	
quality	.0164541	.043222	0.38	0.703	-.0682594	.1011677	
pquality	.0376763	.0765443	0.49	0.623	-.1123477	.1877002	
bcash	.3514341	.0716984	4.90	0.000	.2109078	.4919604	
lagprecp	-.2789671	.0200107	-13.94	0.000	-.3181873	-.2397469	
recpsqrd	.0259331	.0023862	10.87	0.000	.0212563	.03061	
totpct	-.5041463	.2263587	-2.23	0.026	-.9478013	-.0604914	
stshare	.4819198	.1913305	2.52	0.012	.106919	.8569207	
homecand	-.2420607	.1037069	-2.33	0.020	-.4453224	-.038799	
dcpac	.4026476	.0298946	13.47	0.000	.3440552	.4612399	
_cons	5.717728	.1143268	50.01	0.000	5.493652	5.941804	
logit							
educatn	.1092988	.1184172	0.92	0.356	-.1227947	.3413923	
prestige	.2806099	.0635753	4.41	0.000	.1560045	.4052152	
seniorty	-.7225828	.3885569	-1.86	0.063	-1.48414	.0389747	
dleader	.4483005	.117503	3.82	0.000	.2179988	.6786022	
rleader	.2813031	.1753403	1.60	0.109	-.0623576	.6249638	
republic	-1.798423	.1000995	-17.97	0.000	-1.994614	-1.602231	
cope	3.250226	.2013426	16.14	0.000	2.855602	3.64485	
votept	-2.374621	.26542	-8.95	0.000	-2.894835	-1.854408	
quality	.2355944	.0915296	2.57	0.010	.0561997	.414989	
pquality	-.4067092	.1496391	-2.72	0.007	-.6999964	-.113422	
bcash	-.4111928	.1515213	-2.71	0.007	-.7081691	-.1142164	
lagprecp	.4141835	.0447867	9.25	0.000	.3264031	.5019638	
recpsqrd	-.0333042	.0054234	-6.14	0.000	-.0439338	-.0226746	
totpct	3.878043	.4972923	7.80	0.000	2.903368	4.852718	
stshare	.818243	.4648729	1.76	0.078	-.0928911	1.729377	
homecand	.298608	.2603486	1.15	0.251	-.211666	.8088819	
dcpac	.0935253	.0631602	1.48	0.139	-.0302663	.217317	
_cons	-1.331899	.2248491	-5.92	0.000	-1.772595	-.8912029	
shape							
_cons	.4803252	.0074244	64.70	0.000	.4657737	.4948767	

Stata Command

```
predict xb, eq(logit)
```

C:\DATA

Programming Likelihood Functions: The Predict Command and Est. Split

The screenshot displays the Stata 8.0 interface with the following components:

- Review Window:** Contains the Stata command script:


```
set mem 100m
use "K:\an - PACs\Data\large labor PACs.
do "C:\DOCUME~1\ADMINI~1\LOCALS~
do "C:\DOCUME~1\ADMINI~1\LOCALS~
ml model lf splitpop (duration: T = educatn p
gen T=rawdays
ml model lf splitpop (duration: T = educatn p
ml search
ml max
predict xb, eq(logit)
```
- Variables Window:** Lists the variables in the dataset, including Target, Command, Wil, copecase, rrawdays, previous, homecand, lndays, recpsqrd, republic, senior2, lndays2, sample, dcpac, lagrecp2, recsqrd2, bcash2, T, and xb.
- Stata Results Window:** Displays the results of the logit model. The results are organized into two main sections: **logit** and **shape**. Each section contains a table of coefficients, standard errors, z-scores, p-values, and marginal effects.

Variable	Coefficient	Std. Err.	z	P > z	Marginal Effect	95% CI
educatn	.1092988	.1184172	0.92	0.356	-.1227947	.3413923
prestige	.2806099	.0635753	4.41	0.000	.1560045	.4052152
seniorty	-.7225828	.3885569	-1.86	0.063	-1.48414	.0389747
dleader	.4483005	.117503	3.82	0.000	.2179988	.6786022
rleader	.2813031	.1753403	1.60	0.109	-.0623576	.6249638
republic	-1.798423	.1000995	-17.97	0.000	-1.994614	-1.602231
copec	3.250226	.2013426	16.14	0.000	2.855602	3.64485
votept	-2.374621	.26542	-8.95	0.000	-2.894835	-1.854408
quality	.2355944	.0915296	2.57	0.010	.0561997	.414989
pquality	-.4067092	.1496391	-2.72	0.007	-.6999964	-.113422
bcash	-.4111928	.1515213	-2.71	0.007	-.7081691	-.1142164
lagrecp	.4141835	.0447867	9.25	0.000	.3264031	.5019638
recpsqrd	-.0333042	.0054234	-6.14	0.000	-.0439338	-.0226746
totpct	3.878043	.4972923	7.80	0.000	2.903368	4.852718
stshare	.818243	.4648729	1.76	0.078	-.0928911	1.729377
homecand	.298608	.2603486	1.15	0.251	-.211666	.8088819
dcpac	.0935253	.0631602	1.48	0.139	-.0302663	.217317
_cons	-1.331899	.2248491	-5.92	0.000	-1.772595	-.8912029
- Stata Command Window:** Shows the command `gen prob=exp (xb) / (1+exp (xb))`.

Calculates probability of the exchange of a contribution.



Programming Likelihood Functions: Conclusion

- ***Bottom line:***
 - If you need to estimate a model that is not canned in a popular software package, you can probably program it in Stata.
 - All you need to know is the likelihood function!