

# MNL in Stata

---

We have data on the type of health insurance available to 616 psychologically depressed subjects in the United States (Tarlov et al. 1989, JAMA; Wells et al. 1989, JAMA). The insurance is categorized as either an indemnity plan (i.e., regular fee-for-service insurance, which may have a deductible or coinsurance rate) or a prepaid plan (a fixed up-front payment allowing subsequent unlimited use as provided, for instance, by an HMO). The third possibility is that the subject has no insurance whatsoever. We wish to explore the demographic factors associated with each subject's insurance choice. One of the demographic factors in our data is the race of the participant, coded as white or nonwhite.

Load the data using **webuse**

```
webuse sysdsn1, clear
```

Now, use **describe** to take a look at the kinds of variables that are in the data set.

```
. describe
Contains data from http://www.stata-press.com/data/r11/sysdsn1.dta
  obs:          644          Health insurance data
  vars:          13          28 Mar 2009 13:10
  size:        16,744 (99.9% of memory free)
```

variable name	storage type	display format	value label	variable label
patid	float	%9.0g		
noinsur0	byte	%8.0g		no insurance at baseline
noinsur1	byte	%8.0g		no insurance at year 1
noinsur2	byte	%8.0g		no insurance at year 2
age	float	%10.0g		NEMC (ISCNRD-IBIRTHD)/365.25
male	byte	%8.0g		NEMC PATIENT MALE
ppd0	byte	%8.0g		prepaid at baseline
ppd1	byte	%8.0g		prepaid at year 1
ppd2	byte	%8.0g		prepaid at year 2
nonwhite	float	%9.0g		
ppd	byte	%8.0g		
insure	byte	%9.0g	insure	
site	byte	%9.0g		

Sorted by: patid

From this it is not clear what sort of variable that insure is so list the first 10 observations.

```
. list insure in 1/10
```

	insure
1.	Indemnity
2.	Prepaid
3.	Indemnity
4.	Prepaid
5.	.
6.	Prepaid
7.	Prepaid
8.	.
9.	Uninsure
10.	Prepaid

This shows that insure is a string. The **mlogit** command in Stata will handle this for us, but it is a good thing to know. You can also see that there are some missing values. You should always take a look at

the summary statistics, just to get an idea as to what is in the data set. You can often identify qualitative variables and indicators based on these.

```
. summarize
```

Variable	Obs	Mean	Std. Dev.	Min	Max
patid	644	592838.1	315023.2	3292	997539
noinsur0	338	.0710059	.2572155	0	1
noinsur1	339	.0707965	.2568637	0	1
noinsur2	336	.0535714	.2255058	0	1
age	643	44.41415	14.22441	18.11087	86.07254
male	644	.2593168	.4386004	0	1
ppd0	644	.4751553	.4997705	0	1
ppd1	644	.4736025	.4996908	0	1
ppd2	616	.4545455	.4983343	0	1
nonwhite	644	.1956522	.3970103	0	1
ppd	644	.4736025	.4996908	0	1
insure	616	1.595779	.6225427	1	3
site	644	1.987578	.7964742	1	3

Patient age is between 18 and 86, most in the sample are female and white. There appear to be three insurance categories, which is what we expect.

Next, we'll tabulate the data by race. In this table, nothing is held constant. This is simply the numbers and percentages of whites and nonwhites in each insurance category. We'll use mlogit to control for other variables below.

```
webuse sysdsn1, clear
tabulate insure nonwhite, chi2 col
```

insure	nonwhite		Total
	0	1	
Indemnity	251 50.71	43 35.54	294 47.73
Prepaid	208 42.02	69 57.02	277 44.97
Uninsure	36 7.27	9 7.44	45 7.31
Total	495 100.00	121 100.00	616 100.00

Pearson chi2(2) = 9.5599 Pr = 0.008

The simplest way to get the hang of using mlogit is to start with a very simple model that has only 1 independent variable, nonwhite.

```
mlogit insure nonwhite
```

```

. mlogit insure nonwhite

Iteration 0:  log likelihood = -556.59502
Iteration 1:  log likelihood = -551.78935
Iteration 2:  log likelihood = -551.78348
Iteration 3:  log likelihood = -551.78348

Multinomial logistic regression          Number of obs   =      616
                                          LR chi2(2)      =      9.62
                                          Prob > chi2     =     0.0081
                                          Pseudo R2      =     0.0086

Log likelihood = -551.78348

```

insure	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
Indemnity (base outcome)						
Prepaid						
nonwhite	.6608212	.2157321	3.06	0.002	.2379942	1.083648
_cons	-.1879149	.0937644	-2.00	0.045	-.3716896	-.0041401
Uninsure						
nonwhite	.3779586	.407589	0.93	0.354	-.4209011	1.176818
_cons	-1.941934	.1782185	-10.90	0.000	-2.291236	-1.592632

The signs suggest that non-whites are more likely to have prepaid insurance or no insurance than indemnity. However, the coefficient in the uninsured probability is not significant so basically you'd conclude that there is no difference.

Predicting the probabilities from this simple model should give the same results as obtained using `tabulate`.

```

predict p1 if e(sample), outcome(1)
predict p2 if e(sample), outcome(2)
predict p3 if e(sample), outcome(3)
summarize p1 p2 p3

```

```

. summarize p1 p2 p3

```

variable	Obs	Mean	Std. Dev.	Min	Max
p1	616	.4772727	.0603184	.3553719	.5070707
p2	616	.4496753	.0596611	.420202	.5702479
p3	616	.0730519	.0006572	.0727273	.0743802

which it does.

Another way to express the results is in terms of **relative risk**. Suppose the base category is 1. Then

$$\Pr(y = 1) = \frac{1}{1 + e^{x\beta_2} + e^{x\beta_3}} \quad (1.1)$$

$$\Pr(y = 2) = \frac{e^{x\beta_2}}{1 + e^{x\beta_2} + e^{x\beta_3}} \quad (1.2)$$

So that the relative risk is the ratio

$$\frac{\Pr(y = 2)}{\Pr(y = 1)} = e^{x\beta_2} \quad (1.3)$$

It basically measures the relative probability of **y=2** compared to the base outcome. Numbers larger than 1 indicate that it is more probable. It can't be negative.

To obtain the relative risk ratio after estimations, simply recall the results using `mlogit` and use the `rrr` option.

```

mlogit, rrr

```

which produces:

```

Multinomial logistic regression      Number of obs =      616
LR chi2(2) =      9.62
Log likelihood = -551.78348         Prob > chi2 =      0.0081
Pseudo R2 =      0.0086

```

insure	RRR	Std. Err.	z	P> z	[95% Conf. Interval]	
Indemnity (base outcome)						
Prepaid						
1.nonwhite	1.936382	.4177397	3.06	0.002	1.268702	2.955442
Uninsure						
1.nonwhite	1.459302	.5947956	0.93	0.354	.656455	3.244036

You can see that both are greater than 1, though **Uninsure** is not significant. The standard errors are unchanged in this format.

## Adding Variables to the model

Now, add **age**, **male**, and the **site** to the model.

```
mlogit insure age i.male i.nonwhite i.site, base(1)
```

The variables **male** (0, 1) and **site** (1, 2, 3) are categorical and I've identified them as indicator variables using the **i.** factor variable prefix.

```

Iteration 0: log likelihood = -555.85446
Iteration 1: log likelihood = -534.67443
Iteration 2: log likelihood = -534.36284
Iteration 3: log likelihood = -534.36165
Iteration 4: log likelihood = -534.36165

```

```

Multinomial logistic regression      Number of obs =      615
LR chi2(10) =      42.99
Log likelihood = -534.36165         Prob > chi2 =      0.0000
Pseudo R2 =      0.0387

```

insure	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
Indemnity (base outcome)						
Prepaid						
age	-.011745	.0061946	-1.90	0.058	-.0238862	.0003962
1.male	.5616934	.2027465	2.77	0.006	.1643175	.9590693
1.nonwhite	.9747768	.2363213	4.12	0.000	.5115955	1.437958
site						
2	.1130359	.2101903	0.54	0.591	-.2989296	.5250013
3	-.5879879	.2279351	-2.58	0.010	-1.034733	-.1412433
_cons	.2697127	.3284422	0.82	0.412	-.3740222	.9134476
Uninsure						
age	-.0077961	.0114418	-0.68	0.496	-.0302217	.0146294
1.male	.4518496	.3674867	1.23	0.219	-.268411	1.17211
1.nonwhite	.2170589	.4256361	0.51	0.610	-.6171725	1.05129
site						
2	-1.211563	.4705127	-2.57	0.010	-2.133751	-.2893747
3	-.2078123	.3662926	-0.57	0.570	-.9257327	.510108
_cons	-1.286943	.5923219	-2.17	0.030	-2.447872	-.1260134

Checking out the relative risks

. mlogit, rrr

Multinomial logistic regression

Number of obs = 615  
 LR chi2(10) = 42.99  
 Prob > chi2 = 0.0000  
 Pseudo R2 = 0.0387

Log likelihood = -534.36165

insure	RRR	Std. Err.	z	P> z	[95% Conf. Interval]	
Indemnity (base outcome)						
Prepaid						
age	.9883237	.0061223	-1.90	0.058	.9763968	1.000396
1.male	1.75364	.3555444	2.77	0.006	1.178588	2.609267
1.nonwhite	2.650576	.6263875	4.12	0.000	1.66795	4.212086
site						
2	1.119672	.2353442	0.54	0.591	.7416116	1.690461
3	.5554437	.1266051	-2.58	0.010	.3553214	.868278
Uninsure						
age	.9922342	.011353	-0.68	0.496	.9702304	1.014737
1.male	1.571216	.5774008	1.23	0.219	.7645935	3.228799
1.nonwhite	1.242417	.5288177	0.51	0.610	.5394676	2.861341
site						
2	.2977316	.1400865	-2.57	0.010	.1183924	.7487316
3	.8123595	.2975613	-0.57	0.570	.396241	1.665471

Nonwhites are now 2.65 times as likely to use prepaid insurance relative to an indemnity.

To see how race affects the probabilities of insuring, take a look at the marginal effects. There are two ways to consider them. First, you can calculate them using the entire sample for which you have complete set of observations on the regressors (643). Or you can limit the sample to the ones used to estimate the model (615). The latter omits the observations for which the insurance variable is not observed.

. margins nonwhite if e(sample), predict(outcome(Prepaid))

Predictive margins  
 Model VCE : OIM

Number of obs = 615

Expression : Pr(insure==Prepaid), predict(outcome(Prepaid))

	Margin	Delta-method		z	P> z	[95% Conf. Interval]	
		Std. Err.					
nonwhite							
0	.4069474	.0217241	18.73	0.000	.364369	.4495258	
1	.6289396	.0436779	14.40	0.000	.5433325	.7145466	

. margins nonwhite, predict(outcome(Prepaid)) noesample

Predictive margins  
 Model VCE : OIM

Number of obs = 643

Expression : Pr(insure==Prepaid), predict(outcome(Prepaid))

	Margin	Delta-method		z	P> z	[95% Conf. Interval]	
		Std. Err.					
nonwhite							
0	.4082052	.021745	18.77	0.000	.3655858	.4508246	
1	.630078	.0436442	14.44	0.000	.544537	.715619	

In the first instance, `margins` is used with the `if e(sample)` qualifier. This throws out all observations that were not used in the estimation of the model. The `noesample` option uses all available observations. Mercifully, there is not much difference in the two sets of results.

The complete set of marginal effects (using `noesample`) is

```
. margins nonwhite, predict(outcome(Prepaid)) noesample
Predictive margins                                Number of obs =      643
Model VCE    : OIM
Expression   : Pr(insure==Prepaid), predict(outcome(Prepaid))
```

	Margin	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
nonwhite						
0	.4082052	.021745	18.77	0.000	.3655858	.4508246
1	.630078	.0436442	14.44	0.000	.544537	.715619

```
. margins nonwhite, predict(outcome(Uninsure)) noesample
Predictive margins                                Number of obs =      643
Model VCE    : OIM
Expression   : Pr(insure==Uninsure), predict(outcome(Uninsure))
```

	Margin	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
nonwhite						
0	.0776275	.0123981	6.26	0.000	.0533276	.1019274
1	.0586411	.019357	3.03	0.002	.0207021	.0965802

```
. margins nonwhite, predict(outcome(Indemnity)) noesample
Predictive margins                                Number of obs =      643
Model VCE    : OIM
Expression   : Pr(insure==Indemnity), predict(outcome(Indemnity))
```

	Margin	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
nonwhite						
0	.5141673	.0223485	23.01	0.000	.470365	.5579695
1	.3112809	.0418049	7.45	0.000	.2293448	.393217

The `margins` command has different syntax when you want to get marginal effects for indicator variables. In this case you can use the simple syntax `margins varname`. We want the marginal effect on the predictions for the `outcome` listed. For continuous variables the syntax requires the `dydx` option and is `margins, dydx(varname)`. Here is an example for computing the average marginal effect of being another year older (`age`) on the probability of being insured under an indemnity using all available observations.

```
margins, dydx(age) predict(outcome(Indemnity)) noesample
```

```
. margins, dydx(age) predict(outcome(Indemnity)) noesample
Average marginal effects          Number of obs =      643
Model VCE      : OIM
Expression    : Pr(insure==Indemnity), predict(outcome(Indemnity))
dy/dx w.r.t.  : age
```

	dy/dx	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
age	.0026614	.0013966	1.91	0.057	-.0000759	.0053987

I find the syntax somewhat confusing, but that is how it is. Here is a bit of code that replicates how these marginal effects are computed.

```
gen byte nonwhold = nonwhite // save real race
replace nonwhite = 0 // make everyone white
predict wpind, outcome(Indemnity) // predict probabilities
predict wpp, outcome(Prepaid)
predict wpnoi, outcome(Uninsure)
replace nonwhite=1 // make everyone nonwhite
predict nwpind, outcome(Indemnity)
predict nwpp, outcome(Prepaid)
predict nwpoi, outcome(Uninsure)
replace nonwhite=nonwhold // restore real race

summarize wp* nwp*, sep(3)
```

Basically, you generate a variable that is 0 for everyone (white and nonwhite). Get the predictions. Now replace this with 1 for everyone and compute the predictions. These reproduce the results obtained using the margins command.

```
. summarize wp* nwp*, sep(3)
```

Variable	Obs	Mean	Std. Dev.	Min	Max
wpind	643	.5141673	.0872679	.3092903	.71939
wpp	643	.4082052	.0993286	.1964103	.6502247
wpnoi	643	.0776275	.0360283	.0273596	.1302816
nwpind	643	.3112809	.0817693	.1511329	.535021
nwpp	643	.630078	.0979976	.3871782	.8278881
nwpoi	643	.0586411	.0287185	.0209648	.0933874

Comparing these to the marginal effects above reveal that they are indeed, the same!

Finally, here is a bit of code to produce a classification table similar in spirit to the one we used in probit.

```
predict indem, outcome(Indemnity) index // obtain indexes
predict prepaid, outcome(Prepaid) index
gen diff = prepaid-indem // obtain difference
predict sediff, outcome(Indemnity,Prepaid) stddp // & its standard error
gen type = 1 if diff/sediff < -1.645 // definitely indemnity
replace type = 3 if diff/sediff > 1.645 // definitely prepaid
replace type = 2 if type>=. & diff/sediff < . // ambiguous
```

```

label def type 1 "Def Ind" 2 "Ambiguous" 3 "Def Prep"
label values type type // label results
tabulate insure type

```

insure	type			Total
	Def Ind	Ambiguous	Def Prep	
Indemnity	98	156	40	294
Prepaid	57	150	70	277
Uninsure	18	19	8	45
<b>Total</b>	<b>173</b>	<b>325</b>	<b>118</b>	<b>616</b>

To be classified as *definitely indemnity*, the difference in the index between the two choices needs to be statistically significant. So, predict the indices, find the difference, get the standard error of the difference using the built-in option `stdp`, and compute the *t*-ratio. If it is smaller than -1.645, then you definitely choose an indemnity (def indemnity) and if greater than 1.645, you definitely use prepaid (definitely prepaid).

## do-file

```

webuse sysdsn1, clear
tabulate insure nonwhite, chi2 col
mlogit insure i.nonwhite

predict p1 if e(sample), outcome(1)
predict p2 if e(sample), outcome(2)
predict p3 if e(sample), outcome(3)
summarize p1 p2 p3
drop p1 p2 p3
mlogit, rrr

mlogit insure age i.male i.nonwhite i.site, base(1)

margins nonwhite if e(sample), predict(outcome(Prepaid))
margins nonwhite, predict(outcome(Prepaid)) noesample
margins nonwhite, predict(outcome(Uninsure)) noesample
margins nonwhite, predict(outcome(Indemnity)) noesample

margins, dydx(age) predict(outcome(Indemnity)) noesample

mlogit
mlogit, rrr

predict p1 if e(sample), outcome(1)
predict p2 if e(sample), outcome(2)
summarize p1 p2
histogram p1

```

```
mlogit, rrr
```

```
gen byte nonwhold = nonwhite // save real race  
replace nonwhite = 0 // make everyone white  
predict wpind, outcome(Indemnity) // predict probabilities  
predict wpp, outcome(Prepaid)  
predict wpoi, outcome(Uninsure)  
replace nonwhite=1 // make everyone nonwhite  
predict nwpind, outcome(Indemnity)  
predict nwpp, outcome(Prepaid)  
predict nwpoi, outcome(Uninsure)  
replace nonwhite=nonwhold // restore real race
```

```
summarize wp* nwp*, sep(3)
```

```
margins nonwhite, predict(outcome(Prepaid)) noesample  
margins nonwhite, predict(outcome(Indemnity)) noesample
```

```
predict indem, outcome(Indemnity) index // obtain indexes  
predict prepaid, outcome(Prepaid) index  
gen diff = prepaid-indem // obtain difference  
predict sediff, outcome(Indemnity,Prepaid) stddp // & its standard error  
gen type = 1 if diff/sediff < -1.645 // definitely indemnity  
replace type = 3 if diff/sediff > 1.645 // definitely prepaid  
replace type = 2 if type>=. & diff/sediff < . // ambiguous  
label def type 1 "Def Ind" 2 "Ambiguous" 3 "Def Prep"  
label values type type // label results  
tabulate insure type
```