

多重代入

代入とは、欠損したデータを尤もらしい値で補う方法であり、複数の値から代入を行うのが多重代入です。この「代入」では、欠損値のあるデータを複数の尤もらしい値で置き換えて、表現します。

ここでは、 M を代入した回数、 m を個々の代入で表します、つまり、 $m = 1$ は最初の代入、 $m = 2$ は2つ目の代入というように続きます。

多重代入を行う理由

- 架空の喫煙と心臓発作の関係を調査した症例対照研究データを例に考えます。

```
use https://www.stata-press.com/data/r16/mheart0, clear
describe
```

Contains data from <https://www.stata-press.com/data/r16/mheart0.dta>

```
obs:      154      Fictional heart attack data; bmi missing
vars:      9       19 Jun 2018 10:50
```

variable name	storage type	display format	value label	variable label
attack	byte	%9.0g		Outcome (heart attack)
smokes	byte	%9.0g		Current smoker
age	float	%9.0g		Age, in years
bmi	float	%9.0g		Body Mass Index, kg/m^2
female	byte	%9.0g		Gender
hsgrad	byte	%9.0g		High school graduate
marstatus	byte	%9.0g	mar	Marital status: single, married, divorced
alcohol	byte	%24.0g	alc	Alcohol consumption: none, <2 drinks/day, >=2 drinks/day
hightar	byte	%9.0g		Smokes high tar cigarettes

Sorted by:

- 変数 `attack` と `smokes` の関係をロジットモデルで確認します。

```
logit attack smokes age bmi hsgrad female
```

```
Iteration 0: log likelihood = -91.359017
Iteration 1: log likelihood = -79.374749
Iteration 2: log likelihood = -79.342218
Iteration 3: log likelihood = -79.34221
```

```
Logistic regression      Number of obs   =      132
                        LR chi2(5)                 =      24.03
                        Prob > chi2                =      0.0002
Log likelihood = -79.34221      Pseudo R2       =      0.1315
```

attack	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
smokes	1.544053	.3998329	3.86	0.000	.7603945	2.327711
age	.026112	.017042	1.53	0.125	-.0072898	.0595137
bmi	.1129938	.0500061	2.26	0.024	.0149837	.211004
hsgrad	.4048251	.4446019	0.91	0.363	-.4665786	1.276229
female	.2255301	.4527558	0.50	0.618	-.6618549	1.112915
_cons	-5.408398	1.810603	-2.99	0.003	-8.957115	-1.85968

- データセットには 154 の観測値がありますが、欠損値を含むため推定に使用されたのは 132 のみであることがわかります。misstable summarize で欠損値を確認します。

```
misstable summarize
```

Variable	Obs<.			Unique values	Obs<.	
	Obs=.	Obs>.	Obs<.		Min	Max
bmi	22		132	132	17.22643	38.24214

- bmi が欠損している観測値が 22 個あることがわかります。上記のロジットモデルではこれらを完全に排除して推定しましたが、多重代入を使えば、これらの情報を分析に取り込むことができます。

多重代入とは

- 多重代入はシミュレーションを使用してデータの欠損に対して柔軟に対応する、統計技術です。以下のステップで行われます；
 1. 代入: 選択した代入モデルを使用して、 M 個の完全な(欠損値の無い)データセットを作成します。
 2. 推定: 代入したデータセットごと ($m = 1, 2, \dots, M$) に、本来行いたい推定を行います。
 3. プーリング: M 個の分析から得られたそれぞれの結果を 1 つの結果に統合します。
- 多重代入の特徴は、1) 代入データを使用して既存の推定を行う、2) 代入と推定が独立している、の 2 点です。
- 欠損データを扱う手法で一般的なのが、完全ケース分析またはリストワイズ削除と単一代入ですが、これらには、多重代入と比較すると、いくつか点で問題があります。

リストワイズ削除では、欠損値を含む観測値を削除するため、それらの情報は失われます。欠損値が多数あれば、分析結果の有効性は多重代入のものに比べ低くなります。また、残された観測値が母集団とかけ離れていた場合、バイアスのかかったパラメータが導き出されてしまいます。

- 先のロジットモデルでは、リストワイズ削除を使用しています。観測値を削除した標本では、**age** は統計的に有意ではありませんが、多重代入によって全ての観測値を使用した分析では、**age** の有意性が明らかになります。
- リストワイズ削除とは異なり、単一代入は欠損値を削除せず、既知の値として扱います。これでは、分散が過小評価され、精度と信頼区間が過大になり、結果として有意性検定が楽観的になります。これに対して多重代入では、複数回代入を行うことで、サンプリングの分散を保証しています。
- 代入のステップが分析と独立している点において、データ収集・代入ではより正確な代入を行うことが、分析の段階では様々な分析を行うことができます。
- 上の心臓発作の例では、**bmi** の欠損値を、線形回帰を使用して代入し、ロジットモデルで推定を行います。推定では欠損を考慮して調整を行う必要がありません。

多重代入の理論的背景

- 多重代入はベイズ統計の技術を利用し、既知のサンプルから発生させたランダム化分布による点推定値と信頼区間の統計的な妥当性を保証するために Rubin (1996) の定義を使用します。
- 多重代入では、代入データを作成する代入モデルと分析モデルの 2 つを設定しなければなりません。分析モデルでは、興味のあるパラメータ Q の完全データ推定値 \hat{Q} 、および \hat{Q} に関するサンプリングの妥当性 U を推定します。プーリング段階では、完全データで推定された個々の (\hat{Q}, U) を (\hat{Q}_{MI}, T) に統合して反復代入を行います。
- 代入モデルと分析モデルがどちらもベイジアンである場合、ベイジアンモデルによる欠損データの事後分布の反復代入を行います。統合された推定値 \hat{Q}_{MI} とサンプリング分散 $T = W + B$ は、事後平均と Q の分散に近似します。 W は代入データセット内の分散(完全データの推定分散 U の平均)、 B は代入データセット間の分散(反復代入間の推定分散 \hat{Q}_{MI}) です。事後平均と分散が事後分散を正しく表していれば、統合された推測結果は、ベイズ統計の観点からも、頻度統計の観点からも、有効であると判断できます。
- 反復代入による推測の統計的妥当性は、1) 適切な代入によって多重代入が行われているか、2) 分析段階の (\hat{Q}, U) に基づく完全データ推論が適切であるか、に依ります。 (\hat{Q}, U) に基づく完全データ推論は $\hat{Q} \sim N\{Q, \text{Var}(\hat{Q})\}$ であり、 U がサンプルの分布 $\text{Var}(\hat{Q})$ の一致推定量である場合に有効です。
- 多重代入のランダム化妥当性は代入を無限に行うことで保障されますが、実際の代入

回数は、 M の値を決定しなければなりません。ここで、再度 Q を次のように定義します：

$$T_M^{-1/2}(Q - \hat{Q}_M) \sim t_{\nu_M}$$

- \hat{Q}_M は Q の完全データ推定値 M の平均、 $T_M = W + (1 + 1/M)B$ 、また t_{ν_M} は代入回数と欠損している情報の割合に依存する、自由度 ν_M の Student の t 分布です。

M はどの程度の大きさが良いか

- Rubin (1987, 114)では、無限回の M と比較して、欠損している情報の割合が 50%までであれば、代入回数が 2 回であっても有限な M との漸近的な相対有効性(RE)はおおよそ 90%であるとしています。
- 多くの文献では、 $M = 5$ (欠損割合が 50%で、RE が 95%)であれば有効な推測を行うのに十分であるとしています。一般的には、欠損している情報量だけではなく、分析モデルとデータ数を考量して、代入回数を決定する必要があります。 M が 50 以上であれば、安定した結果を得られるとしている研究もあります。
- 代入計算を行う際の実効性とサンプリング誤差の低減を考慮して、Stata では少なくとも 20 回の代入を行うことをお勧めします。

欠損データに関する仮定

- 多重代入では、データが欠損するメカニズムについて仮定を設けていませんが、ほとんどの代入方法では欠損のメカニズムが無視可能な(ignorable)ものであることを前提としています。
- データが欠損するメカニズムが他のデータに依らない、missing completely at random (MCAR)であれば、欠損値を削除しても推定結果の一致性と有効性は保たれます。
- データが欠損するメカニズムは他のデータに依存する missing at random(MAR)の場合、リストワイズ削除を行うと、推定結果にバイアスが生じます。
- 欠損のメカニズムが MAR で、モデルパラメータと欠損パターンのパラメータが異なる場合、欠損パターンを考慮する必要はなく、代入モデルを単純化できます。
- 実際の分析では、欠損メカニズムが無視可能であるかの判別は難しく、上記の仮定を用いるには慎重な判断が必要です。

データの欠損パターン

- $N \times p$ のデータ行列 $Y = (Y_1, Y_2, \dots, Y_p)'$ で、変数は p 個で、観測値は N 個を考えます。行列 R_1 が Y の欠損パターンを表しているとしします。

$$R_1 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

- Y_j が観測値*i*において観測されれば R_1^{ij} は1、観測出来なければ0です。 R_1 の1列目と3列目を入れ替えると、 Y の欠損パターンはモノトーン(単調)ということが出来ます。

$$R_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

- 行を上のように並べ替えると、欠損値の単調性がさらに明らかになります。これに対して、モノトーンではないパターンは次ようになります。 R_2 の最初の2列では、片方の列の欠損値がもう一方を説明するようなパターンが見られません。

$$R_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

- このモノトーンパターンは代入手順を大幅に単純化します。欠損パターンがモノトーンでの多変量代入は、独立した単変量代入の繰り返しに置き換えることができ、代入モデルを柔軟に作成できます。

適切な代入方法

- 多重代入は次の場合、妥当性を担保できます。
 1. 多重代入の推定量 \hat{Q}_{MI} が漸近的に平均 \hat{Q} の正規分布と分散共分散推定 B に従う。
 2. 代入内の分散推定 W は、 $\text{Var}(\hat{Q}_{MI})$ の分散共分散推定 U の一致推定である。
- 実際には、これらの基準で妥当性を判断することは難しく、Rubin (1987, 125-127)は代入モデルと欠損メカニズムのベイジアンモデルの事後分布を利用して代入することを提案しています。予測平均マッチングと連鎖方程式を除き、Stataで利用できる代入手法は、欠損データのベイズ統計による事後推定分布を使用します。
- 代入モデルは、欠損データのメカニズムに関連する予測因子を含む必要があります。例えば、分析モデルで2つの変数が相関していることがわかり、代入モデルでそれらのいずれかを削除してしまうと、相関の推定値に0に向かうバイアスが生じます。また、分析モデルのアウトカム変数を代入モデルで使用しないことも推定値にバイアスを生じさせます。サーベイデータでは、加重やストラータ、クラスタIDなどの構造変数も代入モデルに含めなければなりません。一般的には、分析モデルで使用する全ての変数は、代入モデルにも含めるべきです。
- 不適切な代入モデルがどの程度の影響を与えるかは、観測データに対応する代入データの比率によります。代入した値の割合が小さければ、推定結果に大きな影響を与える

ことはありません。

多重代入データの分析

- 代入データが準備できましたら、それぞれのデータセットで推定を行い、結果を Rubin の combination rule に基づいて統合します。
- 推定段階では、様々な推定方法を使用できますが、それぞれの推定方法は本来、結果を統合すること想定していません。このため、尤度比検定や適合度検定などの推定後機能は多重代入ではそのまま使用できません。

Stata での多重代入

- Stata の多重代入では、代入から結果のプーリングまで、複数の変数を扱う様々な代入方法を利用できます。分析とプーリング段階では、`mi estimate` コマンドで推定した結果を統合します。一般的な推定方法で係数を統合します。解析段階では、一般的な推定のみならず、代入データの管理や診断を行うことができます。
- ここでは、心臓発作のデータの例を使用して、推定を行います。目的は、1) bmi の欠損値を、線形回帰で代入、2) 代入したデータを、`mi estimate` を利用してロジスティック回帰で分析、の 2 つです。これらの 2 つのステップを行う前に、データを `mi` で使用できるように準備します。最初に、データを `mi` データであることを宣言します：

```
use https://www.stata-press.com/data/r16/mheart0
mi set mlong
```

- ここでは、メモリを効率的に利用できるように `mlong` 形式を使用します。
- 代入を行う `mi impute` のために、まず、代入変数を登録します。一般には、`mi register` に `imputed`、`passive` または `regular` を使用して、分析に関連するすべての変数を登録することをお勧めします。

```
mi register imputed bmi
mi register regular attack smokes age hsgrad female
```

- シミュレーションエラーを抑えるため、`add(20)` で 20 個の代入を作成します。また、再現性のため、`rseed()` オプションで乱数シード値を設定します。

```
mi impute regress bmi attack smokes age hsgrad female, add(20) rseed(2232)
```

```

Univariate imputation          Imputations =    20
Linear regression              added =       20
Imputed: m=1 through m=20     updated =        0

```

Variable	Observations per <i>m</i>			
	Complete	Incomplete	Imputed	Total
bmi	132	22	22	154

(complete + incomplete = total; imputed is the minimum across *m* of the number of filled-in observations.)

- 22個の欠損している bmi の値が代入されました。代入したデータと元データを比べて確認するために、元データ ($m = 0$) と最初の代入データ ($m = 1$) および最後の代入データ ($m = 20$) に対して、`mi xeq` で `summarize` を実行します。

```
mi xeq 0 1 20: summarize bmi
```

m=0 data:

-> summarize bmi

Variable	Obs	Mean	Std. Dev.	Min	Max
bmi	132	25.24136	4.027137	17.22643	38.24214

m=1 data:

-> summarize bmi

Variable	Obs	Mean	Std. Dev.	Min	Max
bmi	154	25.28134	3.969649	17.22643	38.24214

m=20 data:

-> summarize bmi

Variable	Obs	Mean	Std. Dev.	Min	Max
bmi	154	25.30992	4.05665	16.44644	38.24214

- 代入データは理想的な形に見えますので、`mi estimate` プリフィックスコマンドでロジットモデルにフィットします。

```
mi estimate, dots: logit attack smokes age bmi hsgrad female
```

Imputations (20):

.....10.....20 done

Multiple-imputation estimates		Imputations	=	20
Logistic regression		Number of obs	=	154
		Average RVI	=	0.0611
		Largest FMI	=	0.2518
DF adjustment: Large sample		DF: min	=	311.30
		avg	=	116,139.89
		max	=	252,553.06
Model F test: Equal FMI		F(5,19590.7)	=	3.52
Within VCE type: OIM		Prob > F	=	0.0035

attack	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
smokes	1.222431	.3608138	3.39	0.001	.5152409	1.92962
age	.0358403	.0154631	2.32	0.020	.0055329	.0661476
bmi	.1094125	.0518803	2.11	0.036	.0073322	.2114929
hsgrad	.1740094	.4055789	0.43	0.668	-.6209156	.9689344
female	-.0985455	.4191946	-0.24	0.814	-.9201594	.7230684
_cons	-5.625926	1.782136	-3.16	0.002	-9.124984	-2.126867

- リストワイズ削除を使用した、上記の **logit** モデルと比較し、欠損によって検出できなくなっていた **age** の有意性を確認できました。

まとめ

- 多重代入はシミュレーションベースの手法です。個々の欠損値を真の値に近づけるのではなく、妥当性のある統計推測を行うことを目的としています。
- 多重代入は、1) 欠損データのメカニズムに対して、代入手法が適切であるとき、2) 完全データの分析が適切であるときに、有効な推測が可能です。
- 欠損データの割合が小さい場合は、代入回数が小さくても(5 から 20)問題はありません。この割合が高かったり、またはデータ構造によっては、最大で 100 回(またはそれ以上)の代入を用意する必要があります。
- 代入回数が小さい場合、多重代入の分布は Student の t (または F)に従います。残差の自由度は M と欠損情報の比率に依存します。代入回数が大きい場合、分布は正規分布(または χ^2)に近づきます。
- 代入モデルで使用する変数が分析モデルより制限されている場合、多重代入の前提が満たされず、有効な推測ができません。一方、分析モデルが制限されていれば、推測は有効ではありますが、保守的になります。分析の前提が異なる場合、その結果はバイアスしてしまいます。
- 多重代入の枠組みでは、尤度や偏差など特定の指標を正しく評価できません。このような推定後機能は、多重代入の結果には直接適用できません。

代入データのスタイル

- **mi** データには、**flongsep**、**flong**、**mlong**、および **wide** の 4 種類があります。
 1. **flong**: $m = 0, m = 1, \dots, m = M$ の代入データが個別の **.dta** データセットとなります。 $m = 0$ が **pat.dta** という名前で保存されれば、 $m = 1$ は **_1_pat.dta**、 $m = 2$ は **_2_pat.dta** となります。**flongsep** は *full long and separate* の略です。
 2. **flong**: $m = 0, m = 1, \dots, m = M$ の代入データを、 $_N = N + M \times N$ の単独のデータセットにします。 N は $M = 0$ の観測数です。**flong** は *full long* の略です。
 3. **mlong**: $m = 0, m = 1, \dots, m = M$ の代入データを $N = N + M \times n$ の単独のデータセットにします。 n は $m = 0$ における欠損値を含む観測値の数です。
 4. **wide**: $m = 0, m = 1, \dots, m = M$ の代入データを $_N = N$ の単独のデータセットにします。多重代入に使用する全ての変数に M 個の変数が追加されます。変数 **bp** が $m = 0$ で欠損値を含んでいる場合、 $m = 1$ では変数 **_1_bp**、 $m = 2$ では変数 **_2_bp** を作成します。

まとめ

1. **mi** データは 4 つのフォーマットのいずれかで保存します。これらの詳細は、上記をご覧ください。
2. **mi** データには $m = 1, 2, \dots, M$ の番号を持つ M 個の代入が含まれ、 $m = 0$ は欠損値を含むオリジナルのデータです。
3. **mi** データの変数は、**imputed**、**passive** または **regular** にいずれかとして登録されます。これら以外は **unregistered** となります。
 - a. **unregister** の変数はほぼ **regular** の変数と同じように扱われます。
 - b. **regular** の変数は通常、欠損値を含みません。欠損値がある場合は、 $m > 0$ において代入が行われません。
 - c. **imputed** 変数は $m = 0$ において欠損値を含みます。 $m > 0$ において代入されます。
 - d. **passive** 変数は **imputed**、**regular**、および他の **passive** 変数の組み合わせです。
4. **imputed** 変数が $m = 0$ において、. より大きい値、.a、.b から .z を含む場合、これらの値は、 $m > 0$ においても欠損値のままです。

mi コマンドの一覧

- オリジナルデータから **mi** データを生成する

mi set	mi データを宣言する
mi register	変数を imputed 、 register 、 passive または unregister として

	登録する
<code>mi unregister</code>	登録されている変数を解除する
<code>mi unset</code>	データをオリジナルの状態に戻す (基本的に使用しません)
● 代入済みのデータをインポートする	
<code>mi import</code>	<code>mi</code> データをインポートする
<code>mi export</code>	<code>mi</code> データを Stata 以外の形式にエクスポートする
● <code>mi set</code> または <code>mi import</code> 後に使用するコマンド	
<code>mi query</code>	<code>mi set</code> の設定を確認する
<code>mi describe</code>	<code>mi</code> データを確認する
<code>mi varying</code>	m ごとの変数を変化を確認する
<code>mi misstable</code>	欠損値を表にする
<code>mi passive</code>	<code>passive</code> 変数を作成し、登録する
● <code>mi</code> データを使う推定	
<code>mi impute</code>	欠損値を代入する
<code>mi estimate</code>	$m > 0$ において推定し、結果を統合する
<code>mi ptrace</code>	MCMC の安定性をチェックする
<code>mi test</code>	係数を検定する
<code>mi testtransform</code>	変形した係数を検定する
<code>mi predict</code>	線形予測を行う
<code>mi predictnl</code>	非線形予測を行う
● <code>mi</code> データの <code>stset</code> 、 <code>svyset</code> 、 <code>tsset</code> および <code>xtset</code> を行う	
<code>mi fvset</code>	<code>mi</code> データの <code>fvset</code>
<code>mi svyset</code>	<code>mi</code> データの <code>svyset</code>
<code>mi xtset</code>	<code>mi</code> データの <code>xtset</code>
<code>mi tsset</code>	<code>mi</code> データの <code>tsset</code>
<code>mi stset</code>	<code>mi</code> データの <code>stset</code>
<code>mi streset</code>	<code>mi</code> データの <code>streset</code>
<code>mi st</code>	<code>mi</code> データの <code>st</code>
● <code>mi</code> データの管理	
<code>mi rename</code>	変数名を変更する
<code>mi append</code>	<code>mi</code> データの <code>append</code>
<code>mi merge</code>	<code>mi</code> データの <code>merge</code>
<code>mi reshape</code>	<code>mi</code> データの <code>reshape</code>
<code>mi stsplitt</code>	<code>mi</code> データの <code>stsplitt</code>
<code>mi stjoin</code>	<code>mi</code> データの <code>stjoin</code>
<code>mi add</code>	<code>mi</code> データセットを追加する

- mi プリフィックスコマンドの無い環境でのデータ管理

<code>mi extract</code>	$m = 0$ のデータを取り出す
...	通常通り、データを管理
<code>mi replace0</code>	mi データの $m = 0$ を取り換える

- $m = 0, m = 1$ に対して、コマンドを実行する

<code>mi xeq: ...</code>	$m = 0, m = 1, m = 2, \dots, m = M$ に対してコマンドを実行する
<code>mi xeq #: ...</code>	$m = \#$ に対して、コマンドを実行する
<code>mi xeq # #: ...</code>	m の値に対して、コマンドを実行する

- その他の便利なコマンド

<code>mi convert</code>	mi データのスタイルを変更する
<code>mi extract #</code>	mi データから $m = \#$ を取り出す
<code>mi select #</code>	mi extract と同様のプログラミング用コマンド
<code>mi copy</code>	mi データをコピーする
<code>mi erase</code>	mi データのファイルを削除する
<code>mi update</code>	mi データを更新する
<code>mi reset</code>	imputed と passive の変数を解除する

簡単な例題

- これから以下の 6 つのコマンドを使用します。

1. 例題データをダウンロードします。

```
use https://www.stata-press.com/data/r16/mheart5
```

2. mi コマンドでデータの設定を行います。

```
mi set mlong
```

3. 欠損値を持ち、代入される変数を mi で Stata に知らせます。

```
mi register imputed age bmi
```

4. 代入の再現性を確保するため、乱数シード値を設定します。この手順はオプションです。

```
set seed 29390
```

5. 3. で設定した変数の欠損値ごとに、 $M = 10$ の代入データを作成します。

```
mi impute mvn age bmi = attack smokes hsgrad female, add(10)
```

6. 代入した 10 個のデータセットごとにモデルにフィットして、推定結果を統合します。

```
mi estimate: logistic attack smokes age bmi hsgrad female
```

- 実行した結果は次のようになります。

```
. use https://www.stata-press.com/data/r16/mheart5, clear
(Fictional heart attack data)
```

```
. webuse mheart5
(Fictional heart attack data)
```

```
. mi set mlong
```

```
. mi register imputed age bmi
(28 m=0 obs. now marked as incomplete)
```

```
. set seed 29390
```

```
. mi impute mvn age bmi = attack smokes hsgrad female, add(10)
```

Performing EM optimization:

note: 12 observations omitted from EM estimation because of all imputation variables missing
observed log likelihood = -651.75868 at iteration 7

Performing MCMC data augmentation ...

```
Multivariate imputation          Imputations =      10
Multivariate normal regression    added =      10
Imputed: m=1 through m=10        updated =       0

Prior: uniform                    Iterations =    1000
                                   burn-in =      100
                                   between =      100
```

Variable	Observations per <i>m</i>			
	Complete	Incomplete	Imputed	Total
age	142	12	12	154
bmi	126	28	28	154

(complete + incomplete = total; imputed is the minimum across *m* of the number of filled-in observations.)

```
. mi estimate: logistic attack smokes age bmi hsgrad female
```

```
Multiple-imputation estimates      Imputations =      10
Logistic regression                Number of obs =     154
                                   Average RVI   =     0.0835
                                   Largest FMI    =     0.2642
DF adjustment: Large sample        DF:   min     =    139.75
                                   avg           =  19,591.87
                                   max           =  67,578.07
Model F test:   Equal FMI          F(   5, 4836.6) =     3.32
Within VCE type: OIM              Prob > F       =     0.0054
```

attack	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
smokes	1.187152	.3623514	3.28	0.001	.4768502	1.897453
age	.0315179	.0163884	1.92	0.055	-.0006696	.0637055
bmi	.1090419	.0516554	2.11	0.037	.0069434	.2111404
hsgrad	.1712372	.4054594	0.42	0.673	-.623472	.9659464
female	-.065744	.4156809	-0.16	0.874	-.8804781	.7489901
_cons	-5.369962	1.863821	-2.88	0.005	-9.054895	-1.685029

- 最後のコマンドで表示される結果は、オッズ比ではなく、係数を報告します。これは、推定コマンドが `logistic` ではなく、`mi estimate` であり、このコマンドは、`logistic` を `mi estimate` 独自の方法で統合して、デフォルトでは係数で結果を報告します。オッズ比で報告したい場合は、次のように入力します。

```
mi estimate, or
```

推定

- 多重代入したデータを用いた推定は、Stata の通常の推定とほぼ同じです。しかし、以下の点に注意が必要です。
 1. `mi` データであることを宣言しなければなりません。
既に多重代入したデータ (Stata フォーマット) があれば、`mi import` でインポートします。
代入したデータがない場合、`mi set` でオリジナルのデータを `mi` データとして宣言して、`mi impute` で欠損値を代入します。
 2. `mi` データの宣言後は、`svyset` や `stset`、`xtset` などのデータ構造を設定するコマンドは使用できません。これらの代わりに、サーベイデータでは `mi svyset`、生存データでは `mi stset`、パネルデータでは `mi xtset` を使用します。
 3. 推定には `mi estimate:` を使用します。
- `mi estimate` をサポートしている推定コマンドは下記の通りです。

コマンド	記述
線形回帰モデル	
<code>regress</code>	線形回帰
<code>cnsreg</code>	制約付き線形回帰
<code>mvreg</code>	多変量回帰
二値応答の回帰モデル	
<code>logistic</code>	ロジスティック回帰、オッズ比を報告
<code>logit</code>	ロジスティック回帰、係数を報告
<code>probit</code>	プロビット回帰
<code>cloglog</code>	補二重対数回帰
<code>binreg</code>	二項分布の拡張機能を伴う GLM
カウント応答の回帰モデル	
<code>poisson</code>	ポアソン回帰
<code>nbreg</code>	負の二項回帰
<code>gnbreg</code>	一般化負の二項回帰
順序応答の回帰モデル	

<code>ologit</code>	順序ロジット回帰
<code>oprobit</code>	順序プロビット回帰
カテゴリカル応答の回帰モデル	
<code>mlogit</code>	多項ロジット回帰
<code>mprobit</code>	多項プロビット回帰
<code>clogit</code>	制約付き(固定効果)ロジスティック回帰
フラクショナル応答の回帰モデル	
<code>fracreg</code>	フラクショナル応答回帰
分位点回帰モデル	
<code>qreg</code>	四分位回帰
<code>iqreg</code>	四分位範囲回帰
<code>sqreg</code>	同時分位点回帰
<code>bsqreg</code>	ブートストラップ分位点回帰
生存回帰モデル	
<code>stcox</code>	Cox 比例ハザードモデル
<code>streg</code>	パラメトリック生存モデル
<code>stcrreg</code>	競合リスク回帰
その他の回帰モデル	
<code>glm</code>	一般化線形モデル
<code>areg</code>	大きなダミーデータセットによる線形回帰
<code>rreg</code>	ロバスト回帰
<code>truncreg</code>	切断回帰
記述統計	
<code>mean</code>	平均の推定
<code>proportion</code>	比率の推定
<code>ratio</code>	比の推定
<code>total</code>	合計の推定
パネルデータモデル	
<code>xtreg</code>	固定効果、変量効果、母集団平均の線形モデル
<code>xtrc</code>	ランダム係数回帰
<code>xtlogit</code>	固定効果、変量効果、母集団平均のロジットモデル
<code>xtprobit</code>	変量効果と母集団平均のプロビットモデル
<code>xtcloglog</code>	変量効果と母集団平均の cloglog モデル
<code>xtpoisson</code>	固定効果、変量効果と母集団平均のポアソンモデル
<code>xtnbreg</code>	固定効果、変量効果と母集団平均の負の二値モデル
<code>xtgee</code>	母集団平均パネルデータモデルを GEE にフィット

マルチレベル混合効果モデル

mixed

マルチレベル混合効果線形回帰

サーベイ回帰モデル

svy:

サーベイデータの推定コマンド(上記に記載のある
ものに限る)