

Description

In this example, we show how to fit FMMS with covariates, and we illustrate how you might determine the number of latent classes. For an example without covariates and for a conceptual overview of FMMS, see [\[FMM\] fmm intro](#).

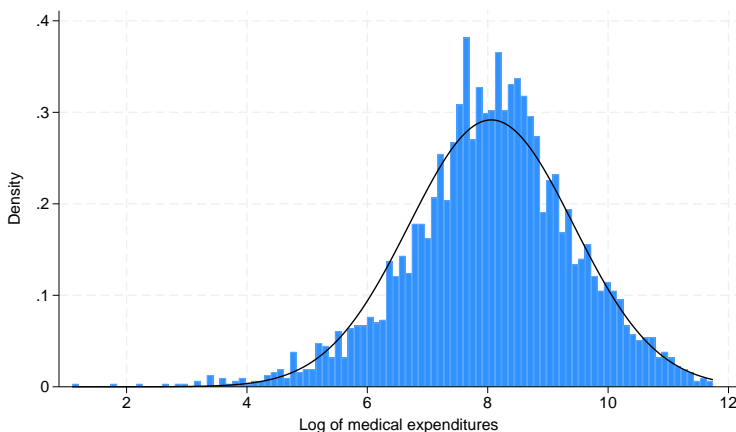
Remarks and examples

Medical expenditures vary greatly from person to person. We believe that some of the variation may be due to having different types of medical care users. We might think of these types as low spenders, average spenders, and high spenders. Because we cannot necessarily tell which group a person belongs to, an FMM may be appropriate for these data.

We use an abbreviated version of `mus03data.dta` from [Cameron and Trivedi \(2022, chap. 3\)](#). `mus03sub.dta` contains information on the log of medical expenditures, `lmedexp`. For brevity, we use only the variables `female`, `age`, `income`, and `totchr`, the last variable recording the number of chronic health problems.

First, let us look at the distribution of medical expenditures.

```
. use https://www.stata-press.com/data/r19/mus03sub  
(Abbreviated dataset mus203mepsmedexp from Cameron and Trivedi (2022))  
. histogram lmedexp, bins(100) normal  
(bin=100, start=1.0986123, width=.10642325)
```



The variable `lmedexp` looks approximately normally distributed. Indeed, it looks as if it may come from a single normal distribution. However, our model includes covariates, and this histogram does not give us an indication of how the regression models may differ across groups. We start by fitting the three-group model, but we will certainly want to check whether a model with a single distribution or with two distributions is a better fit for these data.

```
. fmm 3: regress lmedexp income c.age##c.age totchr i.sex
```

Fitting class model:

Iteration 0: (class) log likelihood = -3246.3993

Iteration 1: (class) log likelihood = -3246.3993

Fitting outcome model:

Iteration 0: (outcome) log likelihood = -4700.2736

Iteration 1: (outcome) log likelihood = -4700.2736

Refining starting values:

Iteration 0: (EM) log likelihood = -7482.765

Iteration 1: (EM) log likelihood = -7327.5583

Iteration 2: (EM) log likelihood = -7271.2407

Iteration 3: (EM) log likelihood = -7254.4109

Iteration 4: (EM) log likelihood = -7246.0793

Iteration 5: (EM) log likelihood = -7238.679

Iteration 6: (EM) log likelihood = -7231.9742

Iteration 7: (EM) log likelihood = -7226.4046

Iteration 8: (EM) log likelihood = -7222.1152

Iteration 9: (EM) log likelihood = -7219.0098

Iteration 10: (EM) log likelihood = -7216.9001

Iteration 11: (EM) log likelihood = -7215.5809

Iteration 12: (EM) log likelihood = -7214.8641

Iteration 13: (EM) log likelihood = -7214.5912

Iteration 14: (EM) log likelihood = -7214.6342

Iteration 15: (EM) log likelihood = -7214.8937

Iteration 16: (EM) log likelihood = -7215.2936

Iteration 17: (EM) log likelihood = -7215.7769

Iteration 18: (EM) log likelihood = -7216.3017

Iteration 19: (EM) log likelihood = -7216.8377

Iteration 20: (EM) log likelihood = -7217.3632

note: EM algorithm reached maximum iterations.

Fitting full model:

Iteration 0: Log likelihood = -4734.6429

Iteration 1: Log likelihood = -4733.3724

Iteration 2: Log likelihood = -4732.1323

Iteration 3: Log likelihood = -4731.0186

Iteration 4: Log likelihood = -4729.3225

Iteration 5: Log likelihood = -4727.7218

Iteration 6: Log likelihood = -4727.6741

Iteration 7: Log likelihood = -4727.6738

Finite mixture model

Number of obs = 2,955

Log likelihood = -4727.6738

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
1.Class	(base outcome)					
2.Class _cons	1.162296	.292186	3.98	0.000	.5896216	1.73497
3.Class _cons	-1.153202	.3188697	-3.62	0.000	-1.778175	-.5282289

Class: 1
Response: lmedexp
Model: regress

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
lmedexp						
income	.0059804	.002604	2.30	0.022	.0008768	.0110841
age	.1201823	.2926979	0.41	0.681	-.4534951	.6938597
c.age#c.age	-.0007572	.0019417	-0.39	0.697	-.0045628	.0030483
totchr	.9223744	.0810612	11.38	0.000	.7634974	1.081251
sex						
Female	.0576508	.1453985	0.40	0.692	-.227325	.3426266
_cons	.6300965	10.96433	0.06	0.954	-20.8596	22.11979
var(e.lmed~p)	1.43183	.1533984			1.160642	1.766382

Class: 2
Response: lmedexp
Model: regress

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
lmedexp						
income	.0023725	.0012209	1.94	0.052	-.0000205	.0047655
age	.2136658	.1075408	1.99	0.047	.0028897	.424442
c.age#c.age	-.0013195	.0007152	-1.84	0.065	-.0027213	.0000823
totchr	.3106586	.0292864	10.61	0.000	.2532583	.3680589
sex						
Female	-.0918924	.0543976	-1.69	0.091	-.1985097	.0147249
_cons	-.9546721	4.017561	-0.24	0.812	-8.828947	6.919602
var(e.lmed~p)	.7966127	.0805009			.6534764	.9711013

Class: 3
Response: lmedexp
Model: regress

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
lmedexp						
income	.0009315	.0048146	0.19	0.847	-.0085049	.0103679
age	-.2645947	.2637125	-1.00	0.316	-.7814618	.2522724
c.age#c.age	.0015761	.001754	0.90	0.369	-.0018616	.0050138
totchr	.186475	.0647115	2.88	0.004	.0596427	.3133072
sex						
Female	-.1761484	.1371471	-1.28	0.199	-.4449517	.0926549
_cons	20.79524	9.853989	2.11	0.035	1.481775	40.1087
var(e.lmed~p)	.3846891	.0983236			.2331038	.634849

That is a lot of output! Let's start with the part of the output that is probably familiar if you have used `regress`. We have one regression table for each class. The coefficient estimates here are interpreted just as you do the coefficients from a linear regression model. Because the dependent variable is log transformed, we can interpret the coefficients in terms of a percentage change. For example, a one-unit increase in `totchr` results in an 18.6% increase in medical expenditures for class 3, all else held constant. The estimates for each class also include a variance term. So, we see that the first class has much higher variability than the third.

The first table in the output gives the coefficients for the latent class membership, next to `1.Class`, `2.Class`, and `3.Class` at the top of the table. These coefficients can be interpreted in the same manner as you interpret the coefficients from multinomial logistic regression (`mlogit`), which is to say that they are difficult to interpret. However, the postestimation command `estat lcprob` will turn them into probabilities.

```
. estat lcprob, nose
Latent class marginal probabilities                                Number of obs = 2,955
```

	Margin
Class	
1	.2215875
2	.708474
3	.0699385

We see that individuals in the population fall into the three classes in proportions 0.22, 0.71, and 0.07. Notice that we specified the `nose` option above. `estat lcprob` can be slow because it is time consuming to compute standard errors when there are a lot of covariates in the model. When fitting preliminary models, we might not be concerned about standard errors of the latent class probabilities, so we use the `nose` option to speed things up.

We have estimated that about 22% of observations are in group 1, about 71% are in group 2, and about 7% are in group 3. But, we still do not know which group corresponds to which spending class. If we want to calculate the level of spending for each group, we can use `estat lcmean` to calculate the marginal means for each class; see [FMM] `estat lcmean`.

```
. estat lcmean
Latent class marginal means                                Number of obs = 2,955
```

		Delta-method				
		Margin	std. err.	z	P> z	[95% conf. interval]
1	lmedexp	7.185846	.1572402	45.70	0.000	6.877661 7.494031
2	lmedexp	8.143981	.0469051	173.63	0.000	8.052049 8.235914
3	lmedexp	10.15809	.1712913	59.30	0.000	9.822369 10.49382

We see that class 1 corresponds to low spenders, class 2 corresponds to average spenders, class 3 corresponds to high spenders.

Because medical expenditures for class 1 and class 2 are relatively close to each other, compared with class 3, we may be tempted to fit a model with two classes. We may also compare our model with a model with one class, which reduces to a linear regression.

First, we store our estimates from the model with three latent classes with the name `fmm3` by using `estimates store`.

```
. estimates store fmm3
```

Then, we fit a model with two classes and then a model with one class, storing the results of each model in `fmm2` and `fmm1`, respectively.

```
. fmm 2: regress lmedexp income c.age##c.age totchr i.sex
(output omitted)
. estimates store fmm2
. fmm 1: regress lmedexp income c.age##c.age totchr i.sex
(output omitted)
. estimates store fmm1
```

Finally, we use `lcstats` to compare these fitted models.

```
. lcstats fmm1 fmm2 fmm3
```

Latent class statistics

	Classes	N	ll	Rank	Entropy	df	LMR	P>LMR
fmm1	1	2,955	-4,807.39	7				
fmm2	2	2,955	-4,758.18	15	0.5304	8	96.90	<0.001
fmm3	3	2,955	-4,727.67	23	0.5367	8	60.07	<0.001

LMR is the Lo-Mendell-Rubin-adjusted likelihood-ratio test statistic.

Likelihood-ratio tests compare the given model versus the same model with one less latent class.

`lcstats` reports the sample size, log likelihood, and rank for each fitted model. It also reports entropy, a measure of class separation, for models with 2 or more latent classes. Larger entropy values, closer to 1, correspond to better separation of classes. The specified estimates only differ in the number of latent classes, each having one more latent class than the previous, so `lcstats` also reports the Lo-Mendell-Rubin (LMR) adjusted likelihood-ratio test for two scenarios.

1. The first is reported in the row labeled `fmm2`, comparing this model with two latent classes versus `fmm1` with one latent class. We find evidence that the two class model fits better than the one class model..
2. The second scenario is reported in the row labeled `fmm3`, comparing this model with three latent classes versus `fmm2` with two latent classes. We find evidence that the three class model fits better than the two class model. scenario.

`lcstats` has options for reporting the usual information criteria. Here we add option `allic` to get all the information criteria. Adding these statistics makes the table wide, so we also add option `split` to request that `lcstats` partition the reported statistics into two tables.

```
. lcstats fmm1 fmm2 fmm3, allc split
```

Latent class statistics

	N	Rank	AIC	BIC	AICc	CAIC	Entropy
fmm1	2,955	7	9,628.77	9,670.71	9,628.81	9,677.71	
fmm2	2,955	15	9,546.35	9,636.22	9,546.52	9,651.22	0.5304
fmm3	2,955	23	9,501.35	9,639.15	9,501.72	9,662.15	0.5367

AIC is the Akaike information criterion.

BIC is the Bayesian information criterion.

AICc is the corrected Akaike information criterion.

CAIC is the consistent Akaike information criterion.

BIC, AICc, and CAIC use N = number of observations.

	Classes	ll	df	LMR	P>LMR
fmm1	1	-4,807.39			
fmm2	2	-4,758.18	8	96.90	<0.001
fmm3	3	-4,727.67	8	60.07	<0.001

LMR is the Lo-Mendell-Rubin-adjusted

likelihood-ratio test statistic.

Likelihood-ratio tests compare the given model
versus the same model with one less latent class.

The Akaike information criterion (AIC) and its sample-size corrected version (AICc) clearly favor the three-component model, whereas the Bayesian information criterion (BIC) and the consistent version of AIC (CAIC) marginally favor the two-component model; see [R] [estat ic](#) for more information about these information criteria.

We will proceed with the three-component model.

□ Technical note

Prior to the addition of `lcstats` to Stata, we might have been tempted to use the standard likelihood-ratio test (see [R] [lrtest](#)) to help us decide how many latent classes to fit. However, a model with $C - 1$ classes with covariates for the mean is not nested in the model extended to C classes because of the additional equation for the mean of the C th component. The model with $C - 1$ classes could be viewed as the model with C classes with variance components of the C th class model going to zero. But the parameter value of zero lies on the boundary of the parameter space, and the standard regularity conditions necessary for the likelihood-ratio test do not hold. See [McLachlan and Peel \(2000, 185\)](#) for a detailed explanation.

□

References

- Cameron, A. C., and P. K. Trivedi. 2022. *Microeconometrics Using Stata*. 2nd ed. College Station, TX: Stata Press.
- McLachlan, G. J., and D. Peel. 2000. *Finite Mixture Models*. New York: Wiley. <https://doi.org/10.1002/0471721182>.

Also see

[FMM] **fmm intro** — Introduction to finite mixture models

[FMM] **fmm: regress** — Finite mixtures of linear regression models

[FMM] **estat lcmean** — Latent class marginal means

[FMM] **estat lcprob** — Latent class marginal probabilities

[FMM] **lcstats** — Latent class model-comparison statistics

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on [citing Stata documentation](#).

