New methods of interpretation using marginal effects for nonlinear models

Scott Long[1]

[1]Departments of Sociology and Statistics
Indiana University

EUSMEX 2016: Mexican Stata Users Group

Mayo 18, 2016

Version: 2016-05-14

Road map for talk

Goals

1. Present new methods of interpretation using marginal effects
2. Show how to implement these methods with Stata

Outline

1. Statistical background
   - Binary logit model
   - Standard definitions of marginal effects
   - Generalizations to concept of marginal effects
2. Stata commands
   - Estimation
   - Post-estimation using `margins` and `lincom`
   - SPost13's `m*` commands
3. Example modeling the occurrence of diabetes

Logit model

Outcome of probability or odds

$$\pi(\mathbf{x}) = Prob(y = 1 \mid \mathbf{x}) \quad \text{and} \quad \Omega(\mathbf{x}) = \pi(\mathbf{x})/[1 - \pi(\mathbf{x})]$$

Multiplicative in odds

$$\Omega(\mathbf{x}) = \frac{\pi(\mathbf{x})}{1 - \pi(\mathbf{x})} = exp(\mathbf{x}'\boldsymbol{\beta}) = exp(\beta_0) \times exp(\beta_1 x_1) \times \ldots$$

Odds ratio: multiplicative change in $\Omega(\mathbf{x})$ for change in x_k holding other variables constant.

Nonlinear in probability

$$\pi(\mathbf{x}) = \frac{\exp(\mathbf{x}'\boldsymbol{\beta})}{1 + \exp(\mathbf{x}'\boldsymbol{\beta})} = \Lambda(\mathbf{x}'\boldsymbol{\beta}) = \Lambda(\beta_0 + \beta_1 x_1 + \ldots)$$

Discrete change: additive change in probability for change in x_k holding other variables at specific values.

Definition of discrete change

1. x_k changes from start to end
2. Remaining x's held constant at specific values $\mathbf{x} = \mathbf{x}^*$
3. Discrete change $\text{DC}(x_k)$

$$\frac{\Delta\pi(\mathbf{x})}{\Delta x_k(\text{start} \rightarrow \text{end})} = \pi(x_k = \text{end}, \mathbf{x} = \mathbf{x}^*) - \pi(x_k = \text{start}, \mathbf{x} = \mathbf{x}^*)$$

4. Interpretation

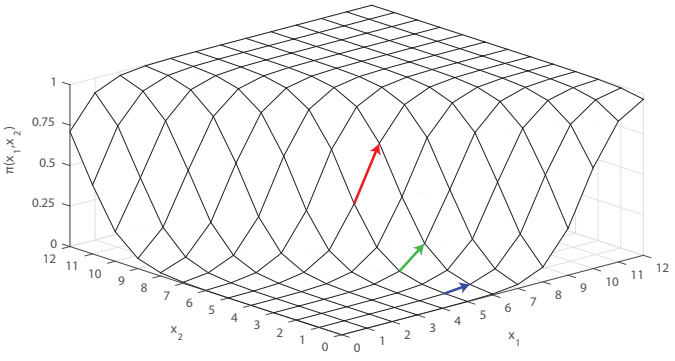
   For a change in x_k from start to end, the probability changes by $DC(x_k)$, holding other variables at the specified values.

5. Everything that follows could be done using marginal changes

$$\frac{\partial\,\pi(\mathbf{x})}{\partial\, x_k} = \frac{\partial\,\Lambda(\beta_0 + \beta_1 x_1 + \ldots)}{\partial x_k}$$

Summarizing the effect of x_k

Since $\Delta\pi$ / Δx_k depends on where it is evaluated, how can it be summarized?



Summary measures

DC at the mean: change at the center of the data

$$\text{DCM}(x_k) = \frac{\Delta\pi(\mathbf{x} = \bar{\mathbf{x}})}{\Delta x_k(\text{start} \rightarrow \text{end})}$$

For someone who is average on all variables, increasing x_k from start to end changes the probability by $DCM(x_k)$.

Average DC: average change in estimation sample

$$\text{ADC}(x_k) = \frac{1}{N}\sum_{i=1}^{N} \frac{\Delta\pi(\mathbf{x} = \mathbf{x}_i)}{\Delta x_{ik}(\text{start} \rightarrow \text{end})}$$

On average, increasing x_k from start to end changes the probability by $ADC(x_k)$.

Generalized discrete change

My talk focuses on generalizing these standard measures

Variations in computing discrete change

Standard effects shown in black; generalized effects in red

Conditional and average change
- Conditional on specific values
- Averaged in the estimation sample
- Averaged in a subsample

Amount of change
- Constant change
- Proportional change
- Change as function of x's
- Change of a component in a multiplicative measure

Number of variables changed
- One variable
- Two or more related variables

Stata requirements

1. Stata 14.1 (most things can be done with Stata 13)
2. `search spost13_ado` to install SPost13
3. `search eusmex` to download example, dataset, and slides

Stata commands

Steps in analysis using official Stata

1. Fit model using factor syntax

   `logit` *depvar* `i.female c.age c.age#c.age`
2. Store estimates using `estimates store` *Model*
3. Make predictions from regression using `margins`, `post`
   - `post` replaces regression results with `margins` results
4. Estimate linear functions of predictions using `lincom`
5. `estimates restore` *Model* restores the regression estimates

Using SPost13

1. `mchange`, `mtable`, `mgen` and `mlincom` are SPost wrappers
2. They simplify things, but everything can be done without them

Modeling diabetes

Cross-sectional data from Health and Retirement Survey[1]

```
. use hrs-gme-analysis2, clear
(hrs-gme-analysis2.dta | Health & Retirement Study GME sample | 2016-04-08)
```

Variable	Mean	Min	Max	Label
diabetes	.205	0	1	Respondent has diabetes?
age	69.3	53	101	Age
bmi	27.9	10.6	82.7	Body mass index (weight/height^2)
weight	174.9	73	400	Weight in pounds
height	66.3	48	89	Height in inches
white	.772	0	1	Is white respondent?
female	.568	0	1	Is female?
hsdegree	.762	0	1	Has high school degree?

N=16,071

[1] Steve Heeringa generously provided the data used in *Applied Survey Data Analysis* (Heeringa et al., 2010). Complex sampling is not used in my analyses.

Two logit model specifications

1. Diabetes
   1.1 Given the diseases burden, small effects are substantively important
   1.2 With N=16,071 small effects are statistically significant
2. Two models that vary in how body mass is included
3. Model Mbmi uses the BMI index

```
logit diabetes c.bmi ///
              i.white c.age##c.age i.female i.hsdegree
estimates store Mbmi
```

4. Model Mwt uses height and weight

```
logit diabetes c.weight c.height ///
              i.white c.age##c.age i.female i.hsdegree
estimates store Mwt
```

5. The estimates are...

Odds ratios and p-values: nuisance parameters...

Variable	Mbmi	Mwt
bmi	1.1046*	
weight		1.0165*
height		0.9299*
white White	0.5412*	0.5313*
age	1.3091*	1.3093*
c.age#c.age	0.9983*	0.9983*
female Women	0.7848*	0.8743#
hsdegree HS degree	0.7191*	0.7067*
_cons	0.0000*	0.0001*
bic	14991.26	14982.03

Note: # significant at .05 level; * at the .001 level.

Average discrete change

1. After estimation I always run `mchange`

```
. estimates restore Mbmi
. mchange, amount(sd) // compute average discrete change
logit: Changes in Pr(y) | Number of obs = 16071
```

	Change	p-value
bmi		
+SD	0.097	0.000
white		
White vs Non-white	-0.099	0.000

(*output omitted*)

2. Interpretation

   Increasing BMI by one standard deviation on average increases the probability of diabetes .097 ($p < .001$).

   On average, the probability of diabetes is .099 less for white respondents than non-white respondents ($p < .001$).

3. How were the DCs computed?

Tool: `margins, at( ... )` and `atmeans`

1. By default, `margins`
   1.1 Makes predictions for every case conditional on observed values
   1.2 These conditional predictions are then averaged
2. Options allow counterfactual predictions
3. Average prediction imagining everyone is white

```
margins, at(white=1)
```

4. Average predictions under two conditions

```
margins, at(white=1) at(white=0)
```

5. Conditional prediction for someone white and average for other variables

```
margins, at(white=1) atmeans
```


ADC for binary x_k: ADC(white)

1. ADC(white) is the difference in average probabilities

$$\text{ADC} = \frac{1}{N}\sum_i \pi(\text{white} = 1, \mathbf{x} = \mathbf{x}_i) - \frac{1}{N}\sum_i \pi(\text{white} = 0, \mathbf{x} = \mathbf{x}_i)$$

2. Compute the two averages

```
. margins, at(white=0) at(white=1) post
Expression   : Pr(diabetes), predict()
1._at        : white           =           0
2._at        : white           =           1
```

	Margin	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at						
1	.2797806	.0073107	38.27	0.000	.265452	.2941092
2	.1805306	.0034215	52.76	0.000	.1738245	.1872367

3. Option `post` save the predictions to matrix `e(b)`

ADC for binary x_k: ADC(white)

4. The posted predictions from `margins`

```
. matlist e(b)
```

	1._at	2._at
y1	.2797806	.1805306

5. `lincom` computes ADC(white) matching prior results

```
. lincom _b[2._at] - _b[1._at]
 ( 1)  - 1bn._at + 2._at = 0
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
(1)	-.09925	.0082362	-12.05	0.000	-.1153927	-.0831073

On average, being white decreases the probability of diabetes by .099 ($p < .001$).

Tool: `margins, at( varnm = generate(exp) )`

1. `at( varnm = generate(exp) )` is powerful but poorly documented
2. Trivially, average prediction at observed values of `bmi`

```
margins, at( bmi = gen(bmi) )
```

3. Average prediction at the observed `bmi` plus 1

```
margins, at( bmi = gen(bmi + 1) )
```

4. Two average predictions

```
margins, at( bmi = gen(bmi) ) at( bmi = gen(bmi + 1) )
```

5. Average at observed plus standard deviation

```
1]  quietly sum bmi // summary statics
2]  local sd = r(sd) // retrieve standard deviation
3]  margins, at( bmi = gen(bmi + `sd') )
```


ADC for continuous x_k: ADC(bmi + sd)

1. Compute probabilities at observed `bmi` and observed + sd

```
. quietly sum bmi
. local sd = r(sd)
. margins, at(bmi = gen(bmi)) at(bmi = gen(bmi + `sd')) post
Expression   : Pr(diabetes), predict()
1._at        : bmi             =           bmi
2._at        : bmi             =           bmi + 5.770835041238605
```

	Margin	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at						
1	.2047166	.0030338	67.48	0.000	.1987704	.2106627
2	.3017056	.005199	58.03	0.000	.2915159	.3118954

2. ADC(bmi + sd)

```
. lincom _b[2._at] - _b[1._at]
 ( 1)  - 1bn._at + 2._at = 0
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
(1)	.0969891	.0035648	27.21	0.000	.0900023	.1039759

On average, increasing BMI by one standard deviation, about 6 points, increases the probability of diabetes by .097 ($p < .001$).

Tool: mlincom simplifies lincom

1. lincom requires column names from e(b)

```
. lincom _b[2._at] - _b[1._at]
 ( 1)  - 1bn._at + 2._at = 0
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
(1)	.0969891	.0035648	27.21	0.000	.0900023	.1039759

2. mlincom uses column numbers in e(b) or rows in margins output

```
. mlincom 2 - 1, stats(all)
```

	lincom	se	zvalue	pvalue	ll	ul
1	0.097	0.004	27.208	0.000	0.090	0.104

3. Why use mlincom?

```
lincom (_b[2._at#1.white] - _b[1._at#1.white]) ///
     - (_b[2._at#0.white] - _b[1._at#0.white])

mlincom (4-2) - (3-1)
```


Generalized measures of discrete change

1. DCM and ADC can be computed more easily with other commands
2. However, the commands showed are essential tools for computing generalized marginal effects
3. Examples of generalizations
   - 3.1 Proportional change in x_k
   - 3.2 Changing linked variables
   - 3.3 Distribution of effects
   - 3.4 Testing effects within a model
   - 3.5 Testing effects across models
   - 3.6 Testing effects across groups
   - 3.7 Changing a component of an interaction

Tool: mtable wrapper for margins

1. margins output is complete, not compact
2. mtable executes margins and simplifies the output and creates tables
   - 2.1 To list the margins commands used, add option commands
   - 2.2 To list margins and mtable output, add option details

Proportional change in x_k

1. Body mass is measured using height and weight

```
logit diabetes c.weight c.height ///
                i.white c.age##c.age i.female i.hsdegree
estimates store Mwt
```

2. ADC(weight + 25) increases weight by 25 pounds, which is
   : a 25% increase if you weigh 100 pounds
   : an 8% increase if you weigh 300 pounds
3. Does increasing weight proportionally make more substantive sense?
4. We compute ADC(weight+25) first, then ADC(weight*1.14)

Proportional change in x_k: ADC(weight+25)

1. Compute ADC(weight + 25)

```
. estimates restore Mwt

. mtable, at(weight = gen(weight)) at(weight = gen(weight + 25)) post
Expression: Pr(diabetes), predict()
```

	Pr(y)
1	0.205
2	0.271

```
.  mlincom 2 - 1, rowname(ADC add)
```

(*output omitted*)

Proportional change in x_k: ADC(weight*1.14)

2. A simple change computes ADC(weight * 1.14)

```
. estimates restore Mwt

. mtable, at(weight = gen(weight)) at(weight = gen(weight * 1.14)) post
Expression: Pr(diabetes), predict()
```

	Pr(y)
1	0.205
2	0.273

```
. mlincom 2 - 1, rowname(ADC pct) add
```

	lincom	pvalue	ll	ul
ADC add	0.067	0.000	0.062	0.071
ADC pct	0.068	0.000	0.063	0.073

3. The effects are deceptively similar as shown below

Discrete change with linked variables

Mathematically linked variables

1. With polynomials multiple variables must change together

$$\frac{\Delta\pi(\mathbf{x})}{\Delta \text{age}(50 \rightarrow 60)} = \pi(\text{age}=60, \text{agesq}=60^2) - \pi(\text{age}=50, \text{agesq}=50^2)$$

2. With factor syntax `margins` handles this automatically

Substantively linked variables

1. Sometimes it makes sense to change multiple variables that are not mathematically linked
2. If two people have the same body mass, is the larger person more likely to have diabetes (the person who it taller and proportionally heavier)?
3. I compute an effect where height and weight change proportionally
4. Use `height` to predict `weight`
5. Use `at(...=gen())` to change `height` and `weight` together

Linked variables: ADC(height, weight)

1. Regress `weight` on `height` and `height` squared

```
. regress weight c.height##c.height, noci
  (output omitted)
R-squared       =    0.2575
```

weight	Coef.	Std. Err.	t	P>|t|
height	-6.338708	1.61073	-3.94	0.000
c.height#c.height	.0855799	.0120867	7.08	0.000
_cons	217.5991	53.5548	4.06	0.000

2. Save the estimates

```
. scalar b0 = _b[_cons]
. scalar b1 = _b[height]
. scalar b2 = _b[c.height#c.height]
```

3. Weight can be predicted

```
weighthat = b0 + b1*height + b2*height#height
```


Linked variables: ADC(height, weight)

4. `at( gen(...) )` predicts weight for a 6 inch change in height

```
1 ]   . mtable, post ///
2a]   >     at( height = gen(height)                  /// observed
2b]   >         weight = gen(weight) )                ///
3a]   >     at( height = gen(height+6)                /// +6 inches height
3b]   >         weight = gen(b0 + b1* (height+6)      /// +estimated weight
3c]   >                     + b2*((height+6)^2)) ) //
```

```
Expression: Pr(diabetes), predict()
```

	Pr(y)
1	0.205
2	0.208

```
. mlincom 2 - 1
```

	lincom	pvalue	ll	ul
1	0.004	0.601	-0.010	0.017

5. Interpretation

There is no evidence that being physically larger without greater body mass contributes to the incidence of diabetes.

Distribution of effects: limitations of summaries

1. ADC and DCM use averages
2. Average discrete change

$$\text{ADC}(x_1) = \frac{1}{N}\sum_i \left[\frac{\Delta\pi}{\Delta(x_1|\mathbf{x}=\mathbf{x}_i)} \right]$$

3. Discrete change at the mean

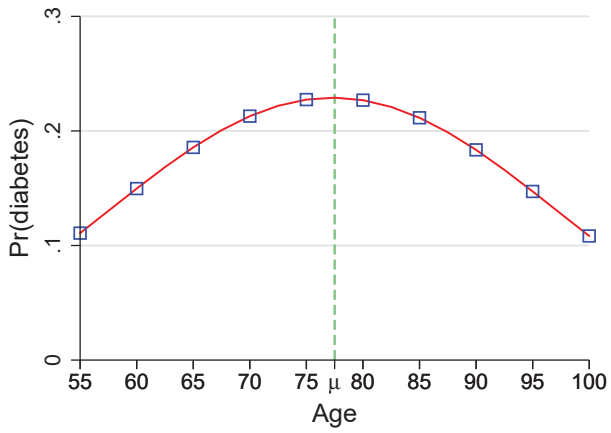
$$\text{DCM}(x_1) = \frac{\Delta\pi}{\Delta(x_1|\mathbf{x}=\bar{\mathbf{x}})} \text{ where } \bar{x}_k = \frac{1}{N}\sum_i x_{ik}$$

4. Sometimes the averages distort the effect of a variables
5. Age has a large impact on diabetes, but ADC and DCM are small. Why?

	Change	p-value
ADC(age+10)	0.018	0.000
DCM(age+10)	0.018	0.000

Distribution of effects: ADC and DCM

Hypothetical data



Undocumented Tool: `margins, generate()`

1. `margins, generate(stub)` creates variables containing predictions for each observation

```
. margins, generate(Prob)
Predictive margins                              Number of obs     =     16,071
Expression   : Pr(diabetes), predict()
```

	Margin	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
_cons	.2047166	.0030316	67.53	0.000	.1987747	.2106584

```
. sum Prob1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
Prob1	16,071	.2047166	.1229016	.0123593	.9067207

2. For details, `help margins generate`

Distribution of effects: ADC(age)

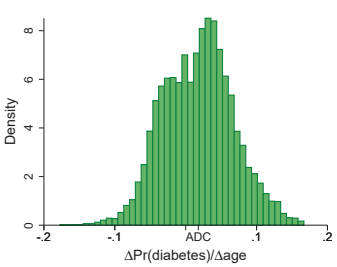
1. To evaluate ADC(age) look at the distribution of DC(age$_i$)
2. Create a variable with the DC for each observation

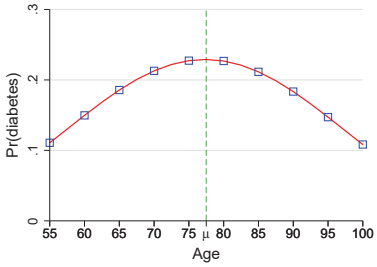
```
1a]   margins, generate(PRage) ///
1b]      at(age = gen(age)) at(age = gen(age+10))

2a]   gen DCage10 = PRage2 - PRage1
2b]   lab var DCage10 "DC for 10 year increase in age"
```

3. Since age-squared was specified using factor syntax, when age is changed age#age is automatically changed
4. A histogram shows why ADC(age) is small

Distribution of effects: ADC(age)





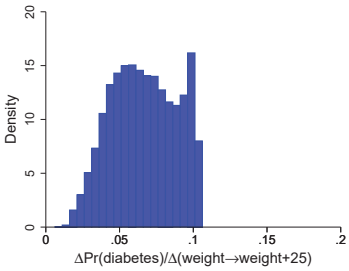
1. The average effect of age is small
2. The effect is large and negative for some people
3. The effect is large and positive for others

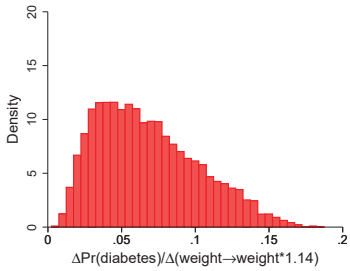
Distribution of effects: ADC(weight)

1. Early we computed ADC(weight+25) and ADC(weight*1.14)

	ADC
ADC add	0.067
ADC pct	0.068

2. The ADCs are similar but the distributions are quite different





Distribution of effects: limitations of summaries

1. ADC and DCM can be useful summaries, but in nonlinear models any summary measures can be misleading
2. The distribution of effects is valuable for assessing effects
3. This is simple with `margins, generate()`
4. Long and Freese (2014) show how do this in earlier versions of Stata

Comparing ADCs for two variables

1. Consider ADC(race) and ADC(bmi+sd)

```
. est restore Mbmi
(results Mbmi are active now)

. mchange bmi white, amount(sd)
logit: Changes in Pr(y) | Number of obs = 16071
Expression: Pr(diabetes), predict(pr)
```

	Change	p-value
bmi		
+SD	0.097	0.000
white		
White vs Non-white	-0.099	0.000

2. Do the effects have the same size?
3. To answer this, the effects must be estimated simultaneously

Comparing ADC(white) and ADC(bmi)

4. Merge the commands for ADC(white) and ADC(bmi)

```
. quietly sum bmi
. local sd = r(sd)
. margins, at(white = 0)      at(white = 1) ///
>          at(bmi = gen(bmi)) at(bmi = gen(bmi + `sd´)) post
Predictive margins                              Number of obs     =     16,071
Model VCE    : OIM
Expression   : Pr(diabetes), predict()
1._at        : white           =           0
2._at        : white           =           1
3._at        : bmi             = bmi
4._at        : bmi             = bmi + 5.770835041238605
```

	Margin	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at						
1	.2797806	.0073107	38.27	0.000	.265452	.2941092
2	.1805306	.0034215	52.76	0.000	.1738245	.1872367
3	.2047166	.0030338	67.48	0.000	.1987704	.2106627
4	.3017056	.005199	58.03	0.000	.2915159	.3118954

Comparing ADC(white) and ADC(bmi)

5. Compute ADCs and test equality

```
. qui mlincom (2-1), rowname(ADC white)
. qui mlincom (4-3), rowname(ADC bmi) add
.     mlincom (2-1) + (4-3), rowname(Sum of ADCs) add
```

	lincom	pvalue	ll	ul
ADC white	-0.099	0.000	-0.115	-0.083
ADC bmi	0.097	0.000	0.090	0.104
Sum of ADCs	-0.002	0.809	-0.021	0.016

6. Conclusion

The health cost of being non-white is equivalent to a standard deviation increase in body mass ($p > .80$).

Comparing ADCs across models

1. Is ADC(female) the same across model specifications?
2. **Tool**: `margins, dydx(female)` computes DC(female) since `i.female`
3. Compute ADC(female) for two models separately

```
. qui logit diabetes c.bmi i.female i.white i.female c.age##c.age i.hsdegree
. qui mtable, dydx(female) rowname(ADC(female) with Mbmi) clear

. qui logit diabetes c.weight c.height i.female i.white c.age##c.age i.hsdegree
. mtable, dydx(female) rowname(ADC(female) with Mwt) below
```

Expression: Pr(diabetes), predict()

	d Pr(y)
ADC(female) with Mbmi	-0.036
ADC(female) with Mwt	-0.020

4. To test if effects are equal, they must be estimated simultaneously

Tool: simultaneous model estimation with gsem

1. gsem simultaneously fits multiple generalized linear models
2. The obvious approach does not work since

```
gsem ///
    (diabetes <- c.bmi             i.female, logit) ///
    (diabetes <- c.weight c.height i.female, logit)
```

is interpreted as

```
gsem ///
    (diabetes <- c.bmi i.female c.weight c.height, logit)
```

3. The solution is a cloned outcome for each model

```
clonevar lhsbmi = diabetes // outcome for Mbmi
clonevar lhswt  = diabetes // outcome for Mwt
```


Comparing ADC(female) across models

1. Fit two models simultaneously with robust standard errors

```
. gsem ///
>     (lhsbmi <- c.bmi i.female i.white c.age##c.age i.hsdegree, logit) ///
>     (lhswt  <- c.weight c.height i.female i.white c.age##c.age i.hsdegree ///
>         , logit) ///
> , vce(robust)
```

```
Generalized structural equation model           Number of obs     =     16,071
Response       : lhsbmi
Family         : Bernoulli
Link           : logit
Response       : lhswt
Family         : Bernoulli
Link           : logit
Log pseudolikelihood = -14914.007
```

	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
lhsbmi <-						
bmi	.099441	.003747	26.54	0.000	.092097	.1067851
female						
Women	-.2423701	.0413006	-5.87	0.000	-.3233177	-.1614225

(*output omitted*)

Comparing ADC(female) across models

2. Simultaneously estimate ADC(female) for both models

```
. margins, dydx(female) post
Average marginal effects                        Number of obs     =     16,071
Model VCE    : Robust
dy/dx w.r.t. : 1.female
1._predict   : Predicted mean (Respondent has diabetes?), predict(pr
               outcome(outcome(lhsbmi))
2._predict   : Predicted mean (Respondent has diabetes?), predict(pr
               outcome(lhswt))
```

	dy/dx	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
1.female						
_predict						
1	-.0360559	.0061773	-5.84	0.000	-.0481631	-.0239487
2	-.0199213	.0089687	-2.22	0.026	-.0374997	-.0023429

Note: dy/dx for factor levels is the discrete change from the base level.

3. The estimates are identical to those estimate earlier

Comparing ADC(female) across models

4. Testing if the effects are equal

```
. mlincom 1-2, stats(all)
```

	lincom	se	zvalue	pvalue	ll	ul
1	-0.016	0.006	-2.526	0.012	-0.029	-0.004

5. Interpretation

The effect of being female is significantly larger when body mass is measured with the BMI index then when height and weight are used to measure body mass ($p < .02$).

Comparing effects across models: summary

1. Jointly fitting models and estimating effects with `margins` is a general approach for comparing effects across models (Mize et al., 2009)
2. The `gsem` command
   2.1 Fits GLM models only
   2.2 `margins` is slow (grumble, grumble), but easy to use
3. Alternatively, the `suest` command
   3.1 Fits a much wider class of models
   3.2 `margins` is fast, but hard to use (grumble, grumble)
4. `suest` and `gsem` produce identical results
5. Specialized commands like `khb` (Kohler et al., 2011) are available

Comparing ADC across subsamples

1. An ADC is typically averaged over the entire sample
2. By averaging within groups, we can examine effects for different groups
   - Is the average effect of BMI the same for whites and non-whites?
3. To test if effects are equal across groups, we estimate the two effects simultaneously `margins, over()`

Tool: `margins, over()`

1. By default, `margins` averages all observations
2. Average for the non-white subsample

```
margins if white==0, ///
    at(bmi = gen(bmi)) at(bmi = gen(bmi+'sd'))
```

3. Average for the white subsample

```
margins if white==1, ///
    at(bmi = gen(bmi)) at(bmi = gen(bmi+'sd'))
```

4. Average for both subsamples simultaneously

```
margins, over(white) ///
    at(bmi = gen(bmi)) at(bmi = gen(bmi+'sd'))
```


Comparing ADC(bmi) by race

1. To compute components for group specific ADC(bmi)

```
. margins, over(white) at(bmi = gen(bmi)) at(bmi = gen(bmi + `sd´)) post
Expression   : Pr(diabetes), predict()
over         : white
1._at        : 0.white
                    bmi           = bmi
               1.white
                    bmi           = bmi
2._at        : 0.white
                    bmi           = bmi + 5.770835041238605
               1.white
                    bmi           = bmi + 5.770835041238605
```

	Margin	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at#white						
1#Non-white	.3097249	.0072773	42.56	0.000	.2954616	.3239881
1#White	.173629	.0032892	52.79	0.000	.1671824	.1800757
2#Non-white	.4302294	.009226	46.63	0.000	.4121468	.448312
2#White	.2636564	.0054903	48.02	0.000	.2528955	.2744172

Comparing ADC(bmi) by race

2. Computing ADC(bmi) by group

```
. qui mlincom  4-2, clear rowname(White: ADC bmi)
. mlincom      3-1, add   rowname(Non-white: ADC bmi)
```

	lincom	pvalue	ll	ul
White				
ADC bmi	0.090	0.000	0.083	0.097
Non-white				
ADC bmi	0.121	0.000	0.112	0.129

3. A second difference compares effects for the groups

```
. mlincom (4-2) - (3-1), rowname(Difference: ADC bmi)
```

	lincom	pvalue	ll	ul
Difference				
ADC bmi	-0.030	0.000	-0.034	-0.027

4. Interpretation

   The average effect of BMI is significantly larger for non-whites than whites ($p < .001$).

Decomposing an effect

1. The BMI index measures relative weight

$$\begin{aligned} \text{BMI} &= 703 \times \frac{weight_{lb}}{height_{in}^2} \\ &= 703 \times weight \times height^{-1} \times height^{-1} \end{aligned}$$

2. With BMI in the model, can we compute the effect of weight change?
   - Why do this? DC(weight) is clearer to patients than DC(bmi)

Decomposing BMI: BMI is an interaction

1. Create components of BMI

```
generate  heightinv = 1/height
    label var heightinv "1/height"
generate  S = 703
    label var S "scale factor to convert from metric"
```

2. These models are *identical*

```
logit diabetes c.S#c.weight#c.heightinv#c.heightinv ///
                    i.white c.age##c.age i.female i.hsdegree
estimates store MbmiFV
logit diabetes c.bmi i.white c.age##c.age i.female i.hsdegree
estimates store Mbmi
```

3. The estimates are identical

Variable	MbmiFV	Mbmi	
c.S#c.weight# c.heightinv# c.heightinv	1.104553		<== odds ratio for BMI
	0.000		
bmi		1.1045533	<== odds ratio for BMI
		0.000	
white			
White	.5411742	.5411742	
	0.000	0.000	

Decomposing BMI: ADC(weight)

4. `margins` with factor syntax makes the rest easy

5. ADC(weight) in MbmiFV changes only weight

```
. qui estimates restore MbmiFV
. mchange weight, amount(sd) delta(25)
logit: Changes in Pr(y) | Number of obs = 16071
Expression: Pr(diabetes), predict(pr)
```

	Change	p-value
weight		
+25	0.065	0.000

Conclusions

Model interpretation and Stata

1. Too often interpretation ends with estimated coefficients
   - Interpretation using predictions is more informative
   - I think of regression coefficients as "nuisance parameters"
2. Methods of interpretation must be practical
   - `margins` makes hard things easy, very hard things merely hard

Conclusions

Which method of interpretation?

1. `mchange` makes it easy make marginal effects a routine part of analysis; marginal effects are almost always more useful than odds ratios
2. Generalized marginal effects can be tailored to your research
3. But, marginal effects might not be the best method of interpretation
4. Tables and plots might be more useful (Long and Freese, 2014) and are easy with `margins` and the `m*` commands
5. The best interpretation is motivated by your substantive question

Thanks to many people

Thank *you* for listening

Collaborators Parts of this work were developed with Long Doan, Jeremy Freese, Trent Mize, and Sarah Mustillo. Jeff Pitblado and David Drukker provided valuable help. Mistakes are my own.

Relevant publications There is a large literature on marginal effects and interpreting models. Long and Freese (2014) include many citations. The references directly related to this presentation are given below.

Bibliography

Allison, P. D. 1999. Comparing logit and probit coefficients across groups. *Sociological Methods & Research* 28(2): 186–208.

Cameron, A. C., and P. K. Trivedi. 2010. *Microeconometrics using Stata*. Revised ed. College Station, Tex.: Stata Press.

Heeringa, S., B. West, and P. Berglund. 2010. *Applied survey data analysis*. Boca Raton, FL: Chapman and Hall/CRC.

Kohler, U., K. B. Karlson, and A. Holm. 2011. Comparing coefficients of nested nonlinear probability models. *Stata Journal* 11(3): 420–438.

Long, J. S. 2009. Group comparisons in logit and probit using predicted probabilities.

Long, J. S., and J. Freese. 2014. *Regression Models for Categorical Dependent Variables Using Stata. Third Edition.* College Station, Texas: Stata Press.

Mize, T. D., L. Doan, and J. S. Long. 2009. A General Framework for Comparing Marginal Effects Across Models.

Additional examples

1. Comparing ADC(weight) across models
2. Discrete change with polynomials
3. Comparing ADCs across models with suest
4. Comparing groups: outcomes and marginal effects
5. Computing DCMs
6. Comparing DCRs

Comparing ADC(weight) across models

1. Recall that

$$\text{BMI} = 703 \times \frac{weight_{lb}}{height_{in}^2}$$

2. Create components of BMI

```
generate heightinv = 1/height
   label var heightinv "1/height"
generate S = 703
   label var S "scale factor to convert from metric"
```

3. These models are *identical*

```
logit diabetes c.bmi i.white c.agec.age i.female i.hsdegree

logit diabetes c.S#c.weight#c.heightinv#c.heightinv ///
                    i.white c.age##c.age i.female i.hsdegree
```


Comparing ADC(weight) across models

4. To compare ADC(weight) requires joint estimation

```
. clonevar lhsbmi = diabetes
. clonevar lhswt  = diabetes
. gsem ///
>     (lhsbmi <- c.s#c.weight#c.heightinv#c.heightinv ///
>         i.white c.age##c.age i.female i.hsdegree, logit) ///
>     (lhswt  <- c.weight c.height ///
>         i.white c.age##c.age i.female i.hsdegree, logit) ///
>     , vce(robust)

Generalized structural equation model           Number of obs     =     16,071
Response       : lhsbmi
Family         : Bernoulli
Link           : logit
Response       : lhswt
Family         : Bernoulli
Link           : logit
Log pseudolikelihood = -14914.007

  (output omitted)
```


Comparing ADC(weight) across models

5. Computing the average predictions for both equations

```
. margins, at(weight=gen(weight)) at(weight=gen(weight+25)) post
Predictive margins                              Number of obs     =     16,071
Model VCE    : Robust
1._predict   : Predicted mean (Diabetes?), predict(pr outcome(lhsbmi))
2._predict   : Predicted mean (Diabetes?), predict(pr outcome(lhswt))
1._at        : weight          = weight
2._at        : weight          = weight+25
```

	Delta-method Margin	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_predict#_at						
1 1	.2047166	.0030419	67.30	0.000	.1987546	.2106786
1 2	.2701404	.0044591	60.58	0.000	.2614007	.27888
2 1	.2047166	.0030394	67.35	0.000	.1987595	.2106737
2 2	.271305	.0044054	61.58	0.000	.2626705	.2799394

Comparing ADC(weight) in two models

6. ADC(weight) for each model and their difference

```
. qui mlincom  2-1, rowname(Mbmi ADC) clear
. qui mlincom  4-3, rowname(Mwt ADC) add
. mlincom (4-3) - (2-1), rowname(Difference) add
```

	lincom	pvalue	ll	ul
Mbmi ADC	0.065	0.000	0.061	0.070
Mwt ADC	0.067	0.000	0.062	0.071
Difference	0.001	0.029	0.000	0.002

7. Conclusion

The effect of weight on diabetes are nearly identical whether body mass is measured with BMI or with height and weight ($p = .03$).

Discrete change with polynomials

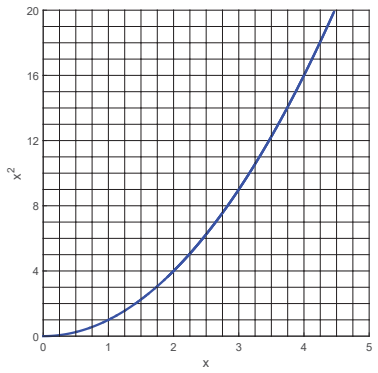
1. With polynomials multiple variables must change together
2. For example,

$$\frac{\Delta\pi(\mathbf{x})}{\Delta\text{age}(50 \rightarrow 60)} = \pi(\text{age}=60, \text{agesq}=60^2) - \pi(\text{age}=50, \text{agesq}=50^2)$$

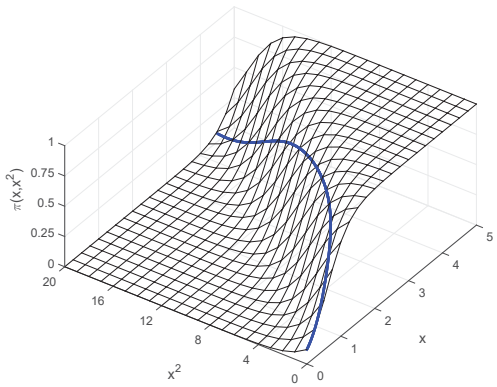
3. This can be computed two ways
   - 3.1 Automatically with factor syntax
   - 3.2 Explicitly with `at(... = gen(...))`

Discrete change with polynomials

1. With x and x^2 only values on the blue curve are mathematically possible



Discrete change with polynomials



2. Changes in the probability reflect linked changes in x and x^2

Tool: factor notation for polynomials

Without factor notation

1. Create the squared term

```
generate agesq = age * age
```

2. Then fit

```
logit diabetes c.age c.agesq ...
```

With factor notation

1. Fit the model

```
logit diabetes c.age##c.age ...
```

2. `c.age##c.age` automatically
   - 2.1 Adds `c.age` to the model
   - 2.2 Creates `c.age#c.age` ≡ `age*age` ≡ `agesq`
   - 2.3 Adds `c.age#c.age` to the model
3. When `c.age` changes, `margins` automatically changes `c.age#c.age`

Discrete change with age & age^2

Correct ADC(age) with factor notation

1. age and age#age automatically change together

```
. logit diabetes c.age##c.age c.bmi i.white i.female i.hsdegree, or
  (output omitted)
. mtable, at(age = gen(age)) at(age = gen(age+10)) post
Expression: Pr(diabetes), predict()
```

	Pr(y)
1	0.205
2	0.223

```
. mlincom 2 - 1
```

	lincom	pvalue	ll	ul
1	0.018	0.000	0.011	0.024

2. Interpretation

   On average, a ten-year increase in age increases the probability of diabetes by .02 ($p < .001$).

Discrete change with age & age^2

Same results without factor notation

```
1] . logit diabetes c.age c.agesq c.bmi i.white i.female i.hsdegree, or
   (output omitted)
2 ]   . mtable, post ///
3a]   >     at( age   = gen( age)          ///
3b]   >         agesq = gen( agesq) )      ///
4a]   >     at( age   = gen( age+10)       ///
4b]   >         agesq = gen((age+10)^2) )
   (output omitted)
5]  . mlincom 2 - 1
   (output omitted)
```

Why use at(gen()) instead of factor syntax

1. `at(gen())` does many things that factor syntax cannot do (gripe)

Comparing ADCs across models with suest

1. Does the effect of a variable change with model specification?
2. Computing ADC(female) for two models

```
. qui logit diabetes c.bmi i.female i.white i.female c.age##c.age i.hsdegree
. estimate store Mbmi
. qui mtable, dydx(female) rowname(ADC(female) with Mbmi) clear

. qui logit diabetes c.weight c.height i.female i.white c.age##c.age i.hsdegree
. estimate store Mwt
. mtable, dydx(female) rowname(ADC(female) with Mwt) below

Expression: Pr(diabetes), predict()
```

	d Pr(y)
ADC(female) with Mbmi	-0.036
ADC(female) with Mwt	-0.020

3. To test if they are equal, the effects must be estimated simultaneously

Comparing effects across models: ADC(female)

Joint estimation with suest

4. The stored estimates are combined and stored

```
. suest Mbmi Mwt, noci
Simultaneous results for Mbmi, Mwt
                                  Number of obs     =     16,071
```

	Coef.	Robust Std. Err.	z	P>|z|
Mbmi_diabetes				
bmi	.099441	.003747	26.54	0.000
white				
White	-.614014	.0480926	-12.77	0.000
:::				
Mwt_diabetes				
weight	.0163568	.0005901	27.72	0.000
height	-.0726272	.0078904	-9.20	0.000
white				
White	-.6324228	.0481997	-13.12	0.000
:::				

```
. qui estimates store Msuest
```


Tool: equation, predict, and expression

1. The two stored models are equations in the suest model

   `Mbmi` becomes `equation(diabetes_Mbmi)`

   `Mwt` becomes `equation(diabetes_Mwt)`

2. With `logit`, `margins` by default computes the "expression" for predicted probabilities

   `Expression : Pr(diabetes), predict()`

3. With `suest`, `margins` only computes $\mathbf{x}'\boldsymbol{\beta}$

   `Expression : Linear prediction, predict()`

4. Sadly, `margins, predict(pr)` does not work with `suest`

5. The solution is the `expression()` option

Tool: equation, predict, and expression

6. With `suest`, `margins` computes $\mathbf{x}'\widehat{\boldsymbol{\beta}}$, but we need $\widehat{\boldsymbol{\pi}}(\mathbf{x}) = \Lambda(\mathbf{x}'\widehat{\boldsymbol{\beta}})$
7. Option `predict(equation(Mbmi_diabetes)` computes $\mathbf{x}'\widehat{\boldsymbol{\beta}}$ for `Mbmi`
8. The logistic CDF function `logistic()` transforms $\mathbf{x}'\widehat{\boldsymbol{\beta}}$ to $\widehat{\boldsymbol{\pi}}(\mathbf{x})$
9. The expression for $\widehat{\boldsymbol{\pi}}(\mathbf{x})$ is

   `expression(logistic(predict(equation(Mbmi_diabetes))`

10. To make code easier, save expressions for `Mbmi` and `Mwt` in locals

```
local EXPR_Mbmi logistic(predict(equation(Mbmi_diabetes)))
local EXPR_Mwt  logistic(predict(equation(Mwt_diabetes)))
```

11. The rest is "easy"

Comparing ADCs across models: ADC(female)

ADC with suest

1. For model `Mbmi`, ADC(female) is

```
. mtable, expression(`EXPR_Mbmi´) at(female=1) at(female=0) post
Expression: , logistic(predict(equation(Mbmi_diabetes)))
```

	female	Margin
1	1	0.189
2	0	0.225

```
. qui mlincom 1 - 2, rowname(ADC Mbmi) clear
```

2. For model `Mwt`

```
. qui mtable, expression(`EXPR_Mwt´) at(female=1) at(female=0) post
. mlincom 1 - 2, rowname(ADC Mwt) add
```

	lincom	pvalue	ll	ul
ADC Mbmi	-0.036	0.000	-0.048	-0.024
ADC Mwt	-0.020	0.026	-0.037	-0.002

3. The estimates match those from `margins` after the individual models; standard errors are robust

Comparing ADCs across models: ADC(female)

Second differences with suest

1. The ADCs from the two models are

$$ADC_{\texttt{Mbmi}} = \widehat{\boldsymbol{\pi}}_{\texttt{Mbmi}}(\texttt{female} = 1, \mathbf{x}) - \widehat{\boldsymbol{\pi}}_{\texttt{Mbmi}}(\texttt{female} = 0, \mathbf{x})$$
$$ADC_{\texttt{Mwt}} = \widehat{\boldsymbol{\pi}}_{\texttt{Mwt}}(\texttt{female} = 1, \mathbf{x}) - \widehat{\boldsymbol{\pi}}_{\texttt{Mwt}}(\texttt{female} = 0, \mathbf{x})$$

2. Since `margins` can't compute these in one step, we compute the parts

$$\widehat{\boldsymbol{\pi}}_{\texttt{Mbmi}}(\texttt{female} = 0, \mathbf{x}) - \widehat{\boldsymbol{\pi}}_{\texttt{Mwt}}(\texttt{female} = 0, \mathbf{x})$$
$$\widehat{\boldsymbol{\pi}}_{\texttt{Mbmi}}(\texttt{female} = 1, \mathbf{x}) - \widehat{\boldsymbol{\pi}}_{\texttt{Mwt}}(\texttt{female} = 1, \mathbf{x})$$

3. Subtracting these is the second difference we want to test

$$ADC_{\texttt{Mbmi}} - ADC_{\texttt{Mwt}} = [\widehat{\boldsymbol{\pi}}_{\texttt{Mbmi}}(\texttt{female} = 1, \mathbf{x}) - \widehat{\boldsymbol{\pi}}_{\texttt{Mbmi}}(\texttt{female} = 0, \mathbf{x})] - [\widehat{\boldsymbol{\pi}}_{\texttt{Mwt}}(\texttt{female} = 1, \mathbf{x}) - \widehat{\boldsymbol{\pi}}_{\texttt{Mwt}}(\texttt{female} = 0, \mathbf{x})]$$

4. The results from `margins` follow

Comparing ADCs across models: ADC(female)

Second differences with suest

5. Using the locals defined earlier

```
. mtable, expression(`EXPR_Mbmi´-`EXPR_Mwt´) ///
>     at(female=1) at(female=0) post brief
Expression: , logistic(predict(equation(Mbmi_diabetes)))
            -logistic(predict(equation(Mwt_diabetes)))
```

	female	Margin
1	1	-0.007
2	0	0.009

6. The 2nd difference is

```
. mlincom 1 - 2, title(Ho: ADC female equal for Mwt & Mbmi)
Ho: ADC female equal for m_wt & m_bmi
```

	lincom	pvalue	ll	ul
1	-0.016	0.012	-0.029	-0.004

7. Interpretation

   The effect of being female is significantly larger when body mass is measured with BMI than with weight and height ($p < .02$).

Comparing effects across models: summary

1. Jointly fitting models and computing effects with `margins` is a general approach for comparing effects across models (Mize et al., 2009)
2. `gsem`
   2.1 Fits generalized linear models only
   2.2 `margins` is slow (grumble, grumble), but easy to use
3. `suest`
   3.1 Fits a much wider class of models
   3.2 `margins` is fast, but hard to use (grumble, grumble)
4. `suest` and `gsem` produce identical results

Comparing groups

Linear regression

1. Coefficients differ by group such as $\beta^W_{\texttt{female}}$ and $\beta^N_{\texttt{female}}$
2. Chow tests are used to test H_0: $\beta^N_{\texttt{female}} = \beta^W_{\texttt{female}}$

Logit and probit

1. Coefficients differ by group such as $\beta^W_{\texttt{female}}$ and $\beta^N_{\texttt{female}}$
2. The estimates combines
   2.1 The effect of x_k which can differ by group
   2.2 The variance of the error which can differ by group
3. Regression coefficients are identified to a scale factor, so standard tests of H_0: $\beta^N_k = \beta^W_k$ are invalid (Allison, 1999)
4. Probabilities and marginal effects are identified (Long, 2009)

Comparing groups: outcomes and effects

Group differences can be examined two ways

1. Differences in probabilities

   $$H_0\text{: } \pi_W(\mathbf{x} = \mathbf{x}^*) = \pi_N(\mathbf{x} = \mathbf{x}^*)$$

   Is the probability of diabetes the same for white and non-white respondents who have the same characteristics?

2. Differences in marginal effects

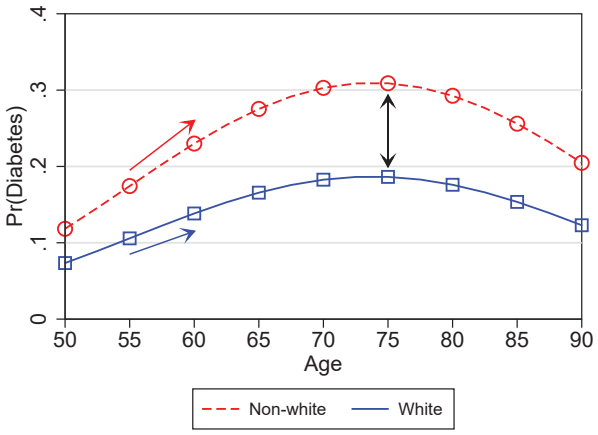
   $$H_0\text{: } \frac{\Delta \pi_W}{\Delta x_k} = \frac{\Delta \pi_N}{\Delta x_k}$$

   Is the effect of x_k the same for whites and non-whites?

3. These dimensions of difference are shown in the next graph

Comparing groups: outcomes and effects

Hypothetical data



Comparing groups: model estimation

1. Factor syntax allows coefficients to differ by `white`

```
logit diabetes ibn.white ///
    ibn.white#(i.female i.hsdegree c.age##c.age c.bmi), nocon
```

2. This is equivalent to simultaneously estimating

```
logit diabetes i.female i.hsdegree c.age##c.age c.bmi if white==1
logit diabetes i.female i.hsdegree c.age##c.age c.bmi if white==0
```

3. Resulting in these estimates

Variable	Whites	NonWhites	
female			
Women	0.713	1.024	<== odds ratios
	0.000	0.755	<== p-values
hsdegree			
HS degree	0.706	0.743	
	0.000	0.000	
age	1.278	1.369	
	0.000	0.000	
:::	:::::	:::::	

Group differences in probabilities by age

1. Compute DC(white) at different ages

```
. mtable, dydx(white) at(age=(55(10)85)) atmeans stats(est p)
Expression: Pr(diabetes), predict()
```

	age	d Pr(y)	p	
1	55	-0.078	0.000	<== DCR(white | age=55)
2	65	-0.124	0.000	<== DCR(white | age=65)
3	75	-0.129	0.000	<== DCR(white | age=75)
4	85	-0.092	0.000	<== DCR(white | age=85)

```
Specified values of covariates
```

	0. white	1. white	1. female	1. hsdegree	bmi
Current	.228	.772	.568	.762	27.9

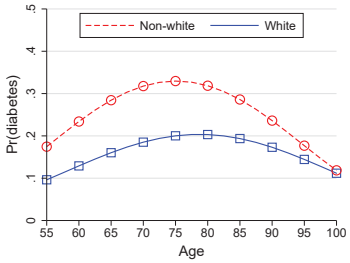
2. Example of interpretation

   For average respondents who are 55, the probability of diabetes is significantly larger for non-whites than whites ($p<.01$).

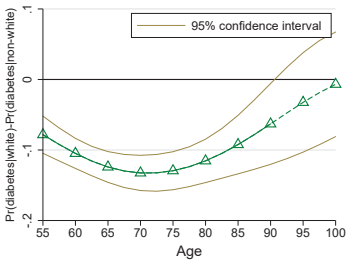
3. Graphically we can show effects at multiple ages

Group differences in probabilities by age

A: Probabilities



B: DCR(race)

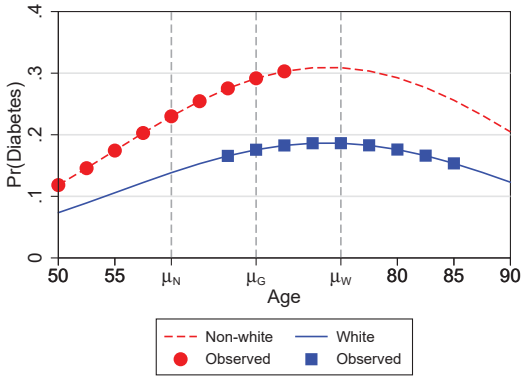


Note: these plots can be computed with `mgen` or `marginsplot`

Group differences in effects

Hypothetical data

1. ADC reflects coefficients and the distribution of predictors
2. DCR is the effect at specific values



diabetes-youngW-red groups-didactic-AMEvMEMV6.do 2016-04-20

Group differences in effects: summary

Comparing ADCs

1. Group differences in ADCs are determined by two things
   - 1.1 Group differences in the probability curves
   - 1.2 Group differences in distribution of variables

Comparing DCRs

1. Group differences in DCRs are determined by two things
   - 1.1 Group differences in the probability curves
   - 1.2 The specific location where they are evaluated
2. They do not depend on group differences in the distribution of variables

Which to use?

1. The answer depends on what you want to know?

Group differences in ADC(bmi + 5)

1. To compute ADC(bmi + 5) by race

```
. mtable, over(white) at(bmi = gen(bmi)) at(bmi = gen(bmi+5)) post
Expression: Pr(diabetes), predict()
```

	Pr(y)
0.white#c.1	0.310
1.white#c.1	0.174
0.white#c.2	0.391
1.white#c.2	0.257

```
. qui mlincom  3-1,          rowname(ADC(bmi) non) stats(est p) clear
. qui mlincom  4-2,          rowname(ADC(bmi) wht) stats(est p) add
.     mlincom (4-2) - (3-1), rowname(Difference)   stats(est p) add
```

	lincom	pvalue
ADC(bmi) non	0.082	0.000
ADC(bmi) wht	0.083	0.000
Difference	0.002	0.826

2. Conclusion

   The average effects of BMI are not significantly different for whites and non-whites (p=.83).

Group differences in DCR(age + 10)

1. ADC(age) might not be useful due to nonlinearity
2. We compare DCR(age+10) at different ages
   - 2.1 Other variables are held at sample means
   - 2.2 Group specific means could be used (Long and Freese, 2014)
3. For example, DCR(age + 10) at 55

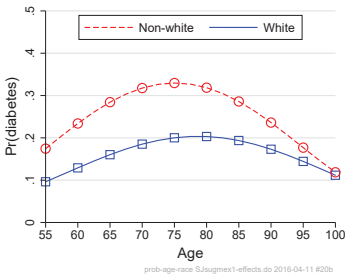
```
mtable, atmeans post ///
    at(age=55 white=0) at(age=55 white=1) ///
    at(age=65 white=0) at(age=65 white=1)
mlincom  3-1,          rowname(DC nonwhite) stats(est p) clear
mlincom  4-2,          rowname(DC white)    stats(est p) add
mlincom (4-2) - (3-1), rowname(Dif at 55)   stats(est p) add
```

4. And so on, with the following results

Group differences in DCR(age + 10)

5. DCRs show group differences in effect of age at different ages

		lincom	pvalue
55:	DC non	0.110	0.000
	DC white	0.064	0.000
	Difference	-0.046	0.001
70:	DC non	0.001	0.940
	DC white	0.018	0.001
	Difference	0.017	0.180
85:	DC non	-0.109	0.000
	DC white	-0.049	0.000
	Difference	0.060	0.003



prob-age-race SJsugmex1-effects.do 2016-04-11 #20b

6. The differences in DCRs do not depend on group differences in the distribution of age or other variables

DCM for continuous x_k: DCM(bmi + sd)

Discrete change at the mean

1. Let bmi increase from mean(bmi) to mean(bmi) + sd(bmi)

```
. qui sum bmi
. local mn = r(mean)
. local mnplus = r(mean) + r(sd)
```

2. Option atmeans holds other variables at their means

```
. margins, atmeans at(bmi = `mn´) at(bmi = `mnplus´) post

Expression   : Pr(diabetes), predict()
1._at        : bmi             =    27.89787
               0.white         =    .2284239 (mean)
               1.white         =    .7715761 (mean)
               age             =    69.29276 (mean)
               0.female        =    .4315226 (mean)
               1.female        =    .5684774 (mean)
               0.hsdegree      =    .2375086 (mean)
               1.hsdegree      =    .7624914 (mean)
<continued>
```


DCM for continuous x_k: DCM(bmi + sd)

```
2._at        : bmi             =    33.6687
               0.white         =    .2284239 (mean)
               1.white         =    .7715761 (mean)
               age             =    69.29276 (mean)
               0.female        =    .4315226 (mean)
               1.female        =    .5684774 (mean)
               0.hsdegree      =    .2375086 (mean)
               1.hsdegree      =    .7624914 (mean)
```

	Delta-method Margin	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at						
1	.2097641	.0045531	46.07	0.000	.2008401	.2186881
2	.3202789	.0066246	48.35	0.000	.307295	.3332628

3. For complex models the output gets very long, so mtable was written.

Tool: mtable wrapper for margins

1. margins output is complete, not compact
2. mtable executes margins and simplifies the output (and more)
   - 2.1 To see the margins commands being used, add option commands
   - 2.2 To see margins and mtable output, add option details

DCM for continuous x_k: DCM(bmi + sd)

2. mtable obtains identical results as margins

```
. mtable, atmeans at(bmi = `mn´) at(bmi = `mnplus´) post
Expression: Pr(diabetes), predict()
```

	bmi	Pr(y)
1	27.9	0.210
2	33.7	0.320

Specified values of covariates

	1. white	age	1. female	1. hsdegree
Current	.772	69.3	.568	.762

3. Computing DCM(bmi + sd)

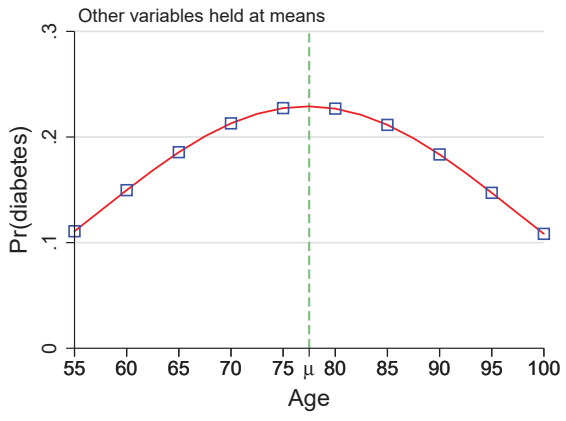
```
. mlincom 2 - 1
```

	lincom	pvalue	ll	ul
1	0.111	0.000	0.102	0.119

For someone who is average, increasing BMI by one standard deviation increases the probability of diabetes by .111 ($p < .001$).

Comparing DCRs

1. Is the effect of age significantly different at different ages?



Comparing DCR(age) at different ages

2. Compute probabilities at four ages with other variables at means

```
. mtable, at(age=(60(10)90)) post atmeans
Expression: Pr(diabetes), predict()
```

	age	Pr(y)
1	60	0.150
2	70	0.213
3	80	0.227
4	90	0.183

Specified values of covariates

	bmi	1. white	1. female	1. hsdegree
Current	27.9	.772	.568	.762

3. DCRs at different ages

```
. mlincom 2-1, clear rowname(DCR60)
. mlincom 3-2, add   rowname(DCR70)
. mlincom 4-3, add   rowname(DCR80)
```

Comparing DCR(age) at different ages

4. Test differences in DCRs

```
. mlincom (2-1) - (3-2), add rowname(DCR60 - DCR70)
. mlincom (2-1) - (4-3), add rowname(DCR60 - DCR80)
. mlincom (3-2) - (4-3), add rowname(DCR70 - DCR80)
```

5. Summarizing

```
. mlincom, twidth(14)
```

	lincom	pvalue	ll	ul
DCR60	0.063	0.000	0.054	0.073
DCR70	0.014	0.004	0.004	0.023
DCR80	-0.043	0.000	-0.061	-0.026
DCR60 - DCR70	0.049	0.000	0.037	0.062
DCR60 - DCR80	0.107	0.000	0.083	0.130
DCR70 - DCR80	0.057	0.000	0.046	0.069

6. Interpretation

The effects of a ten-year increase in age are significantly different at ages 60, 70, and 80 ($p < .001$).

The end

No more examples!