

# New methods of interpretation using marginal effects for nonlinear models

Scott Long<sup>1</sup>

<sup>1</sup>Departments of Sociology and Statistics  
Indiana University

*EUSMEX 2016: Mexican Stata Users Group*

Mayo 18, 2016

# 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 of marginal effects
2. Stata commands
  - ▶ Estimation using factor notation, storing estimates, and `gsem`
  - ▶ Post-estimation using `margins` and `lincom`
  - ▶ SPost13's `m*` commands
3. Example modeling the occurrence of diabetes

# Logit model

## Nonlinear in probability

$$\pi(\mathbf{x}) = \frac{\exp(\mathbf{x}'\beta)}{1 + \exp(\mathbf{x}'\beta)} = \Lambda(\mathbf{x}'\beta)$$

Marginal effect: additive change in probability for change in  $x_k$   
holding other variables at specific values

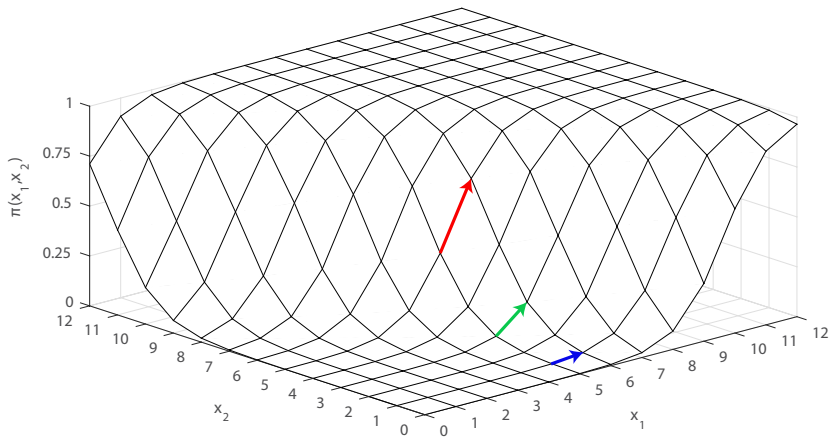
## Multiplicative in odds

$$\Omega(\mathbf{x}) = \frac{\pi(\mathbf{x})}{1 - \pi(\mathbf{x})} = \exp(\mathbf{x}'\beta)$$

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

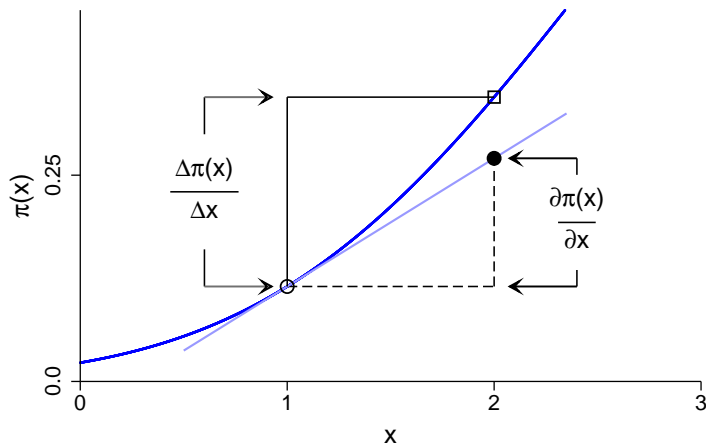
# Logit model: measures of effect

1. **Odds ratios:** identical at each arrow
2. **Marginal effects:** different at each arrow



# Marginal effects

1. **Marginal change:** instantaneous rate of change in  $\pi(x)$
2. **Discrete change:** change in  $\pi(x)$  for discrete change in  $x$



dcVSmc brm-me-dcV14.do 2015-06-10

## Definition of discrete change

1. Variable  $x_k$  changes from start to end
2. The remaining  $x$ 's are held constant at specific values  $\mathbf{x} = \mathbf{x}^*$
3. Discrete change for  $x_k$

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

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.*

## Examples of discrete change

1. DC conditional on the specific values  $\mathbf{x}^*$

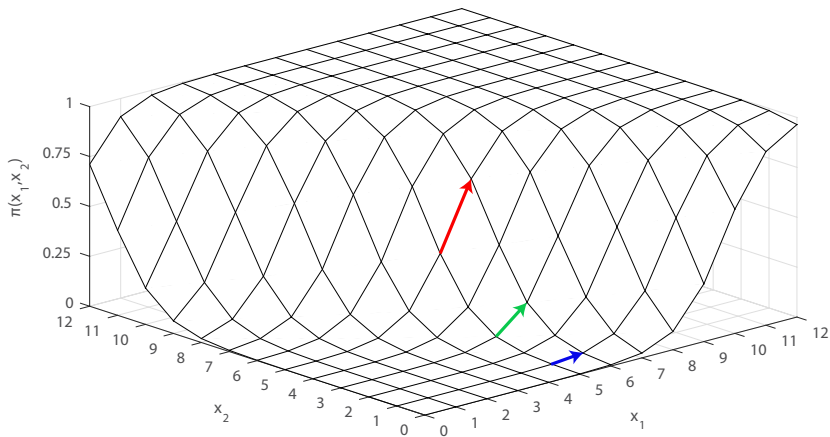
$$\frac{\Delta\pi(\mathbf{x} = \mathbf{x}^*)}{\Delta x_k(0 \rightarrow 1)} = \pi(x_k = 1, \mathbf{x} = \mathbf{x}^*) - \pi(x_k = 0, \mathbf{x} = \mathbf{x}^*)$$

2. DC conditional on the observed values for observation  $i$

$$\frac{\Delta\pi(\mathbf{x} = \mathbf{x}_i)}{\Delta x_{ik}(x_{ik} \rightarrow x_{ik} + 1)} = \pi(x_k = x_{ik} + 1, \mathbf{x} = \mathbf{x}_i) - \pi(x_k = x_{ik}, \mathbf{x} = \mathbf{x}_i)$$

## The challenge of summarizing the effect of $x_k$

Since the value of  $\Delta\pi / \Delta x_k$  depends on where it is evaluated, how do you summarize the effect?





# Common summary measures of discrete change

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

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

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

## 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  $\text{ADC}(x_k)$ .*

# Variations in computing discrete change

## Conditional and average change

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

## Type of change

- Additive change
- Proportional change
- Changes as a function of  $x$ 's
- Change of a component of a multiplicative measure

## Number of variables changed

- One variable
- Two or more mathematically linked variables
- Two or more substantively related variables

# Stata installation, data, and do-files

1. Examples use Stata 14.1, but most things can be done with Stata 13
2. Requires the `spost13_ado` package
3. Examples and slides available with `search eusmex`

# Stata commands

1. Fitting logit model with factor syntax

```
logit depvar i.var c.var c.var1#c.var2
```

2. Regression estimates are stored and restored

```
estimates store ModelName
```

```
estimates restore ModelName
```

3. `margins` estimates predictions from current regression results
4. `margins, post` stores these predictions allowing `lincom` to estimate functions of predictions
5. `mchange`, `mtable`, `mgen` and `mllincom` are SPost wrappers

# Modeling diabetes

1. Cross-section data from Health and Retirement Survey<sup>1</sup>
2. Outcome is self-report of diabetes
  - 2.1 Small changes are substantively important
  - 2.2 Since changes can be statistically significant since N=16,071
3. Road map for examples
  - 3.1 Compute standard measures of change to explain commands
  - 3.2 Extend these commands to compute complex types of effects
  - 3.3 Illustrate testing equality of effects within and across models

---

<sup>1</sup>Steve Heringa generously provided the data used in *Applied Survey Data Analysis* (Heringa et al., 2010). Complex sampling is not used in my analyses.

# Dataset and variables

```
. 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?
white	.772	0	1	Is white respondent?
bmi	27.9	10.6	82.7	Body mass index (weight/height <sup>2</sup> )
weight	174.9	73	400	Weight in pounds
height	66.3	48	89	Height in inches
age	69.3	53	101	Age
female	.568	0	1	Is female?
hsdegree	.762	0	1	Has high school degree?

N=16,071

# Two primary model specifications

1. Model **Mbmi** includes the BMI index

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

2. Model **Mwt** includes height and weight

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

3. The estimates are...

## Odds ratios and p-values tell us little

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. `mchange` is a useful first step after fitting a model

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

		Change	p-value
<code>bmi</code>			
	+SD	0.097	0.000
<code>white</code>			
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.*

*On average, the probability of diabetes is .099 less for white respondents than non-white respondents.*

3. Where did these numbers come from?

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

1. By default,
  - 1.1 margins computes prediction for every observation
  - 1.2 Then the predictions are averaged
2. Options allow predictions at “counterfactual” values of variables
3. Average prediction assuming everyone is white  
`margins, at(white=1)`
4. Two average predictions under two conditions  
`margins, at(white=1) at(white=0)`
5. Conditional prediction if white with means 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. margins computes the two averages

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

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

```
1._at           : white           =           0
```

```
2._at           : white           =           1
```

		Delta-method				
		Margin	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. 1.\_at is the average treating everyone as nonwhite

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

4. 2.\_at is the average treating everyone as white

## ADC for binary $x_k$ : ADC(white)

### 5. Option post saves the predictions to e(b)

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

	1. _at	2. _at
y1	.2797806	.1805306

### 6. lincom computes ADC(white)

```
. 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

### 7. Interpretation

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

## Tool: `m`lincom simplifies `lincom`

1. `lincom` requires column names from `e(b)` that can be complex

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

2. `m`lincom uses column numbers in `e(b)` or rows in margins output

```
m
```

lincom (4-2) - (3-1)

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

1. margins, at( varnm = generate(exp) ) is a powerful, nearly undocumented option that generates values for making predictions

2. Trivially, average prediction at observed values of bmi

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

3. Average prediction at observed values 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
```

```
2] local sd = r(sd)
```

```
3] margins, at(bmi = gen(bmi + 'sd'))
```

# ADC for continuous $x_k$ : $\text{ADC}(\text{bmi} + \text{sd})$

## 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. $\text{ADC}(\text{bmi} + \text{sd})$

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

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

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

## Tool: mtable wrapper for margins

1. `margins` output is complete, not compact
2. `mtable` executes `margins` and simplifies the output (and more)
  - ▶ `mtable, commands` lists the margins commands used
  - ▶ `mtable, detail` shows margins output and mtable output



# 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)  
2._at       : bmi           = 33.6687  
              0.white       = .2284239 (mean)  
              1.white       = .7715761 (mean)
```

```
<continued>
```

## DCM for continuous $x_k$ : DCM(bmi + sd)

```
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

## DCM for continuous $x_k$ : DCM(bmi + sd)

2. Alternatively, `mtable` runs margins and reformats the results

```
. 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. DCM(bmi + sd)

```
. mlincom 2 - 1
```

	lincom	pvalue	ll	ul
1	0.111	0.000	0.102	0.119

*For an average person, increasing BMI by one standard deviation increases the probability of diabetes by .111 ( $p < .001$ ).*

# Generalized measures of discrete change

1. `mchange` makes the above computations automatically
2. I did it the hard way to illustrate powerful commands
3. Now these commands are used for some interesting things

## Proportional change in $x_k$

1. Body mass can be measured with height and weight

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

2.  $ADC(\text{weight} + 25)$  increases weight by 25 pounds for everyone
3. Increasing weight 25 pound is a
  - ▶ 25% increase from 100 pounds
  - ▶ 14% increase from average weight
  - ▶ 8% increase from 300 pounds
4. Is the effect of a percentage increase in weight more meaningful than an additive increase?
5. First, compute  $ADC(\text{weight}+25)$ ...

# Proportional change in $x_k$ : ADC(weight+25)

## 1. Computing 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

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

## Proportional change in $x_k$ : $\text{ADC}(\text{weight} * 1.14)$

2. A simple change computes  $\text{ADC}(\text{weight} * 1.14)$

```
. 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

# Discrete change with polynomials

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

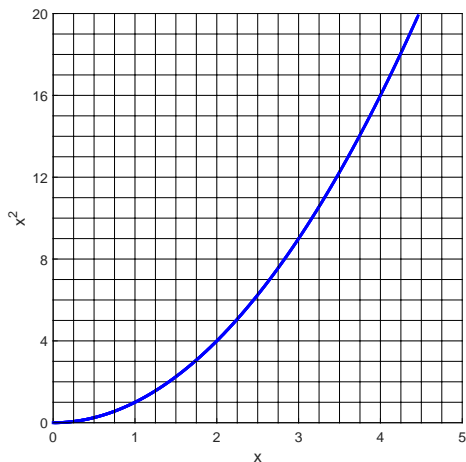
$$\frac{\Delta\pi(\mathbf{x})}{\Delta\text{age}(50 \rightarrow 60)} = \pi(\text{age}=60, \text{agesq}=60^2) - \pi(\text{age}=50, \text{agesq}=50^2)$$

3. This can be done two ways
  - 3.1 The easy way with factor syntax
  - 3.2 The hard way with `at(... = gen(...))`

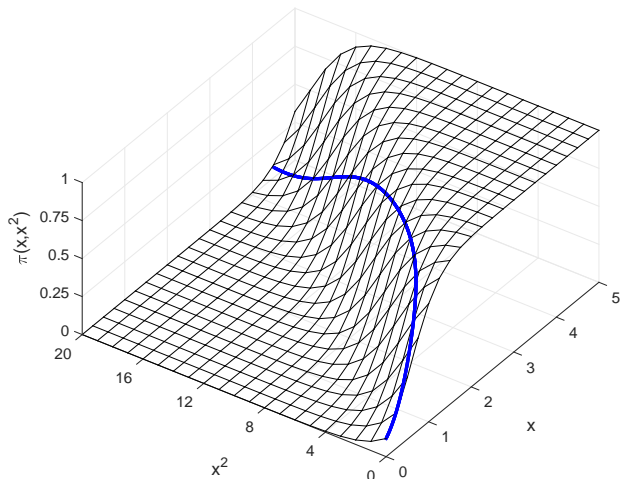


# Discrete change with polynomials

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

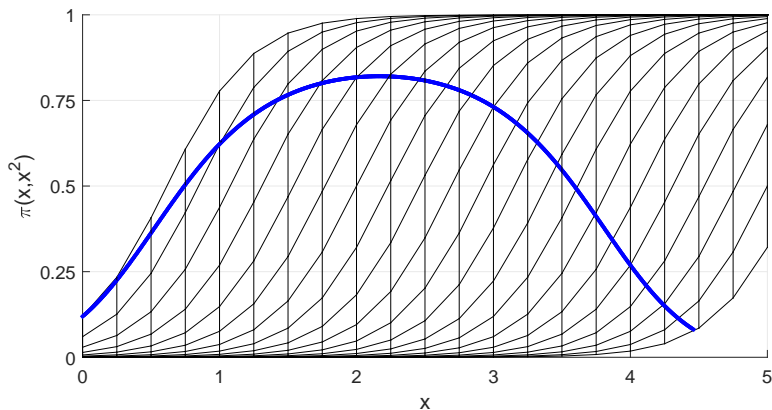


## Discrete change with polynomials



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

## Discrete change with polynomials



3. The probability increases and decreases as  $x$  and implicitly  $x^2$  change

# Tool: factor notation for polynomials

## Without factor notation

1. Generate age-squared

generate `agesq = age * age`

2. Model specification

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

## With factor notation

1. Model specification where `c.` is necessary

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

2. `c.age##c.age` does three things

2.1 Adds `c.age` to the model

2.2 Creates `c.age#c.age`  $\equiv$  `c.age*c.age`

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<sup>2</sup>

## Correct ADC 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, rowname(FV)
```

	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<sup>2</sup>

## 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, at( age = gen( age)          ///
3] >             agesq = gen( agesq) )    ///
4] >             at( age = gen( age+10)    ///
5] >             agesq = gen((age+10)^2) ) ///
6] >             post
    (output omitted)

7] . mlincom 2 - 1
    (output omitted)
```

## The power of at( gen() )

1. With factor syntax you do not need at(...=gen(...)) for polynomials
2. However, at(...=gen(...)) allows complex links among variables

# Discrete change with associated variables

1. Age and age-squared are **mathematically linked**
2. Other variables could be **substantively associated**
3. Example: To examine the effect of cultural capital on health, change all cultural assets together, not a single asset
4. Example: Are “larger people” (taller people with the same body mass) more likely to have diabetes?
  - ▶ Use height to predict weight
  - ▶ Use margins, `at(...=gen())` to change height and weight together

This example illustrates the power of `margins, at(...=gen(...))`

## Associated 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 estimates

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



## Associated variables: ADC(height, weight)

3. Use `at(gen(...))` to predict weight assuming a 6" change in height

```
1] . mtable, post ///
2] > at( height = gen(height)           /// observed height
3] >    weight = gen(weight) )         /// observed weight
4] > at( height = gen(height+6)       /// +6 inches
5] >    weight = gen(b0 + b1*(height+6) /// +estimated weight
6] >    + 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

4. Interpretation

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

# Distribution of effects

1. ADC and DCM are common summary measures of change
2. Each uses the mean to summarize a distribution
3. ADC: average discrete change

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

4. DCM: discrete change at the mean

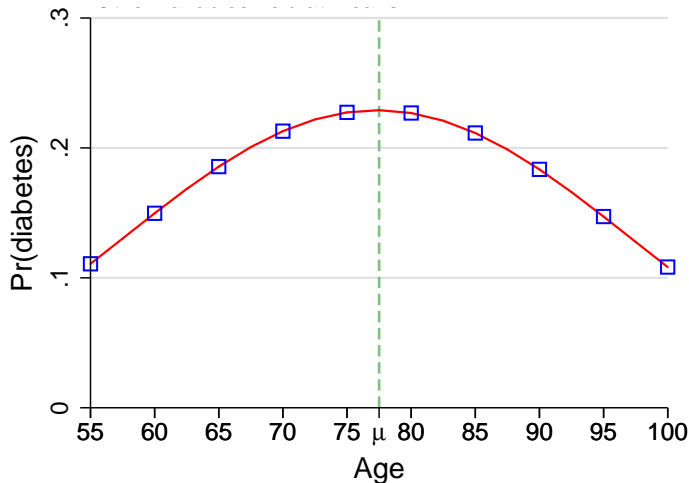
$$\text{DCM}(x_1) = \frac{\Delta\pi}{\Delta(x_1|\mathbf{x} = \bar{\mathbf{x}})} \text{ where } \bar{x}_k = \frac{1}{N} \sum_i x_{ik}$$

5. Hypothetical data shows why means can be misleading

# Distribution of effects: ADC and DCM

Hypothetical data

6. Does age affect diabetes since  $ADC(\text{age})$  and  $DCM(\text{age})$  are near 0?



## Undocumented Tool: margins, generate()

1. margins, gen(*stub*) creates variables with predictions for each observation (help margins generate)
2. For example, to save probabilities for 16,071 cases and average them

```
. margins, gen(Prob)
```

```
Predictive margins                                Number of obs      =      16,071
```

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

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

```
. sum Prob1 // matches margins estimate
```

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

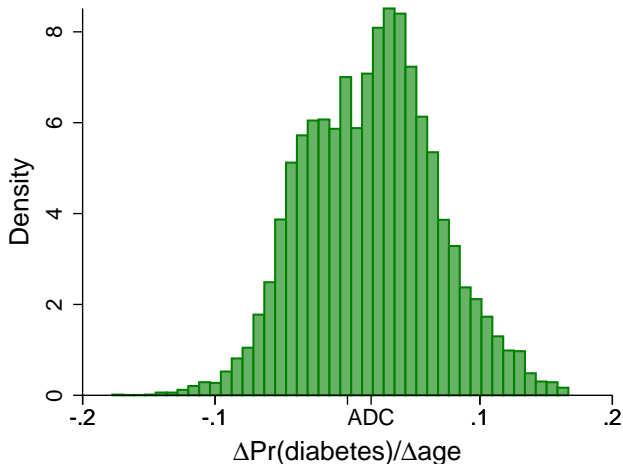
## Distribution of effects: ADC(age)

1. To evaluate ADC(age) look at the distribution of DC(age<sub>i</sub>)
2. Create a variable with the DC for each observation

```
1] margins, generate(PRage) ///  
2]       at(age = gen(age)) at(age = gen(age+10))  
3]       gen DCage10 = PRage2 - PRage1  
4]       lab var DCage10 "DC for 10 year increase in age"
```

## Distribution of effects: $ADC(\text{age})$

3. The average effect of age is small, but the effect is large and negative for some people and large and positive for others

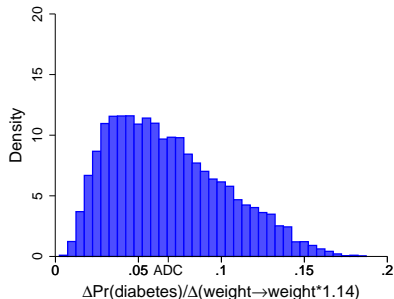
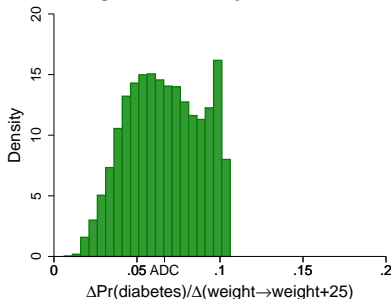


# Distribution of effects: ADC(weight)

## 1. Now consider ADC(weight+25) and ADC(weight\*1.14)

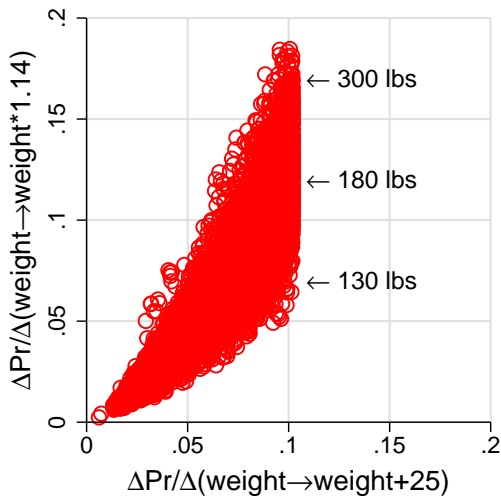
```
1] mtable, gen(PRadd) at(weight=gen(weight)) at(weight=gen(weight+25)) post
2] generate DCadd = PRadd2 - PRadd1
3] lab var DCadd "DC for 25 pound increase"
4] mtable, gen(PRpct) at(weight=gen(weight)) at(weight=gen(weight*1.14)) post
5] generate DCpct = PRpct2 - PRpct1
6] lab var DCpct "DC for 14 percent increase in weight"
```

## 2. The changes have very different distributions



## Distribution of effects: $ADC(\text{weight})$

3. While the ADCs are close, effects for individuals can differ greatly



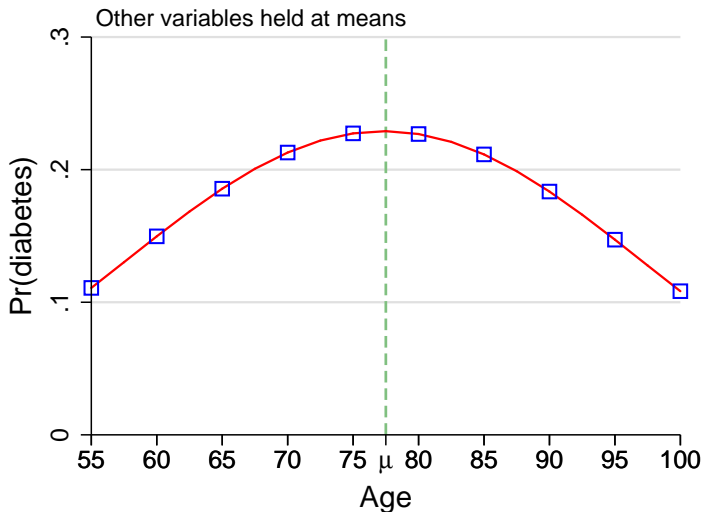


## Distribution of effects: limitations of summaries

1. ADC and DCM are more useful than odds ratios!
2. In nonlinear models, any summary measures can be misleading
3. The distribution of effects is valuable for assessing effects and is simple with `margins`, `generate()`
  - ▶ Long and Freese (2014) show how do this without the `gen()` option
4. For age, multiple DCRs are more useful than ADC or DCM

# Comparing DCRs

1. Are the DCR(age) significantly different at different ages?



# Comparing DCR(age) at different ages

## 2. Compute probabilities at 4 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 ( $p < .001$ ).*

# Comparing ADCs for two variables

1. ADC(race) and ADC(bmi+sd) have similar size, but different signs

```
. 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. Can you justify saying the effects have the same size?
3. To test equality they must be **estimated simultaneously**

# Comparing ADC(white) and ADC(bmi)

## 4. Simultaneously compute components of ADC(white) and ADC(bmi)

```
. quietly sum bmi
. local sd = r(sd)
. margins, at(white=(0 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
```

	Delta-method					
	Margin	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)

## 4. Compute effects and test equality

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

	lincom	pvalue	ll	ul
ADC female	-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

## 5. Conclusion

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

# Comparing ADC across subsamples

1. An ADC is typically averaged over the estimation 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. This requires `margins, over()`



## Tool: margins, over()

1. By default, margins averages over all observations
2. Averages on subsamples are possible with if and over()
3. Averaging for the non-white subsample

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

4. For the white subsample

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

5. For both subsamples simultaneously

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

# Comparing ADC(bmi) by race

## 1. Use over() 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
```

	Delta-method				
	Margin	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 ( $p < .001$ ).*

# Comparing ADCs across models

1. Does the effect of a variable change with model specification?
2. **Tool:** margins, dydx(female) computes DC(female) since i.female
3. Computing ADC(female) for two models

```
. 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 they are equal, the effects must be estimated simultaneously

# Tool: simultaneous model estimation with gsem

1. `gsem` simultaneously fits multiple GLM models
2. The obvious approach does not work since

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

is interpreted as a single model

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

3. The solution is to create cloned outcomes for each model

```
. clonevar lhsbmi = diabetes // outcome for bmi model
. clonevar lhswt = diabetes // outcome for weight height model
```

# Comparing ADC(female) across models

## 1. Estimating the models simultaneously

```
. 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. Estimate ADC(female) for both models simultaneously

```
. qui est restore Mgsem
. 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))
```

		Delta-method			[95% Conf. Interval]	
	dy/dx	Std. Err.	z	P> z		
1.female						
_predict						
1	<b>-0.0360559</b>	.0061773	-5.84	0.000	-.0481631	-.0239487
2	<b>-0.0199213</b>	.0089687	-2.22	0.026	-.0374997	-.0023429

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

## Comparing ADC(female) across models

### 3. Test if ADC(female) is the same in both models

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

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

### 4. Interpretation

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



## Comparing effects across models: summary

1. Jointly estimating models with `gsem` and computing effects with `margins` is a general approach for comparing effects across models (Mize et al., 2009)
2. `gsem`
  - 2.1 Fits the GLM class of models, but does not fit non-GLM models
  - 2.2 `margins` is slow (grumble, grumble)
3. `suest`
  - 3.1 Fits a much wider class of models
  - 3.2 `margins` is fast, but much harder to use (grumble, grumble)
4. `suest` and `gsem` produce identical results
5. Specialized commands like `khb` (Kohler et al., 2011) are available

# Comparing groups

## Linear regression

1. Coefficients differ by group such as  $\beta_{\text{female}}^W$  and  $\beta_{\text{female}}^N$
2. Analysis focuses on Chow tests such as  $H_0: \beta_{\text{female}}^N = \beta_{\text{female}}^W$

## Logit and probit

1. Coefficients differ by group such as  $\beta_{\text{female}}^W$  and  $\beta_{\text{female}}^N$
2. The coefficients 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. Since regression coefficients are identified to a scale factor, Chow-type tests of  $H_0: \beta_k^N = \beta_k^W$  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

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

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

2. Differences in marginal effects

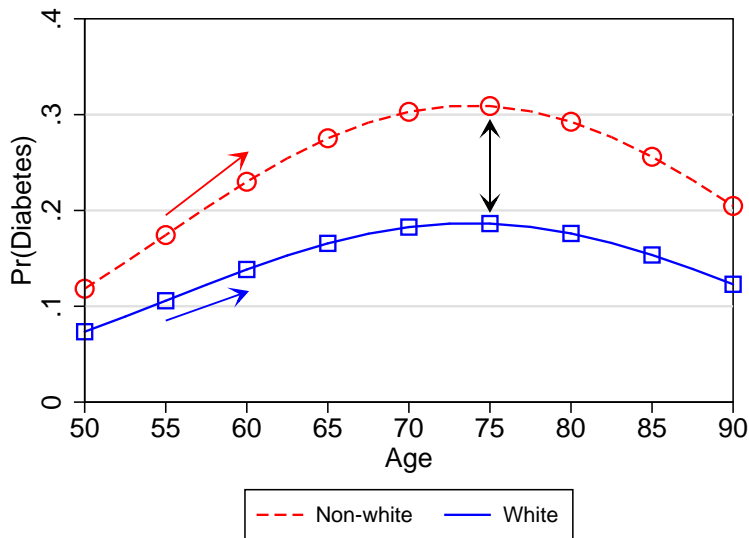
$$H_0: \frac{\Delta\pi_W}{\Delta x_k} = \frac{\Delta\pi_N}{\Delta x_k}$$

*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	
:::	:::::	:::::	

# Comparing groups: probabilities by age

## 1. Compute DC(white) at various 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

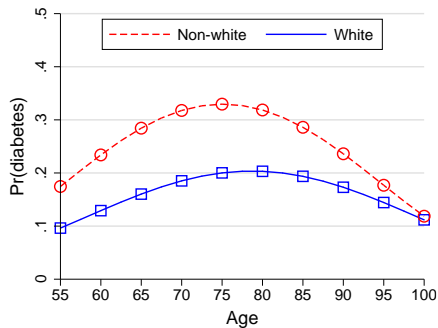
## 2. Example of interpretation

*The probability of diabetes is significantly larger for 55 year-old non-whites than whites who are average on other characteristics ( $p < .01$ ).*

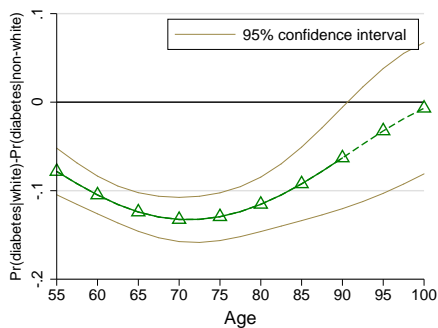
## 3. Graphically we can show effects at multiple ages

# Comparing groups: probabilities by age

## A: Probabilities



## B: DCR(race)

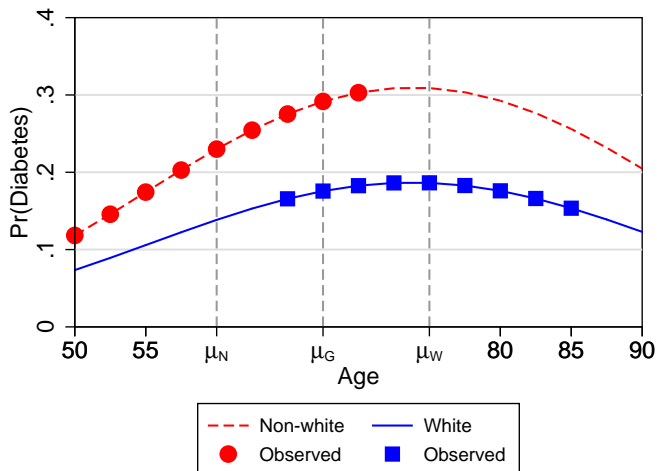


Note: these plots can be computed with `mgen` or `marginsplot`

# Comparing groups: ADC or DCM?

Hypothetical data

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





# Comparing groups: ADC or DCM?

## Comparing ADCs

1. Differences in ADCs are determined by both
  - 1.1 Differences in the probability curves
  - 1.2 Differences in distribution of variables

## Comparing DCRs

1. DCRs show differences in probability curves at a specific location
2. DCRs do not depend on the distribution of variables

## Which to use?

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

# Comparing groups: $ADC(bmi + 5)$

## 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 ( $p=.83$ ).*

## Comparing groups: $\text{DCR}(\text{age} + 10)$

1. Since  $\text{ADC}(\text{age})$  might not be useful due to nonlinearity, we compare  $\text{DCR}(\text{age}+10)$  at different ages
  - 1.1 Other variables are held at sample means
  - 1.2 Group specific means could be used (Long and Freese, 2014)
2. For example,  $\text{DCR}(\text{age} + 10)$  at 55

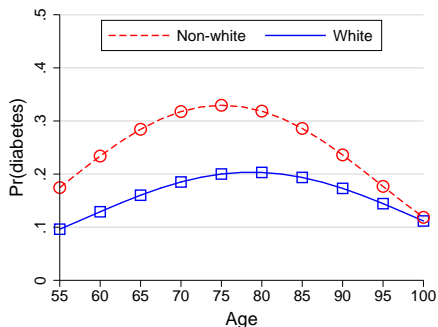
```
mtable, at(age=55 white=(0 1)) at(age=65 white=(0 1)) atmeans post
mlincom 3-1,          rowname(DC nonwhite) stats(est p) clear
mlincom 4-2,          rowname(DC white)     stats(est p) add
mlincom (4-2) - (3-1), rowname(Dif at 55)   stats(est p) add
```

3. And so on, with the following results

# Comparing groups: DCR(age+10)

## 4. 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

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

## \* Decomposing an effect

1. The BMI index measures relative weight or body mass

$$\text{BMI} = \frac{\text{weight}_{kg}}{\text{height}_m^2} = 703 \times \frac{\text{weight}_{lb}}{\text{height}_{in}^2}$$

2. Question 1: With BMI in the model, what is the effect of weight?

- ▶ Why do this? DC(weight) is clearer to patients than DC(bmi)

3. Question 2: Does DC(weight) depend on how body mass is measured?

4. To answer these questions think of BMI as an interaction

$$\text{BMI} = 703 \times \text{weight} \times \text{height}^{-1} \times \text{height}^{-1}$$

# Decomposing BMI: BMI as 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.bmi i.white c.age##c.age i.female i.hsdegree
estimates store Mbmi
```

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

## 3. The estimates are identical

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

## Decomposing BMI: ADC(weight)

4. margins with factor syntax makes the rest trivial
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

6. ADC(weight) in `Mwt` is slightly larger

```
. qui estimates restore Mwt
. 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	0.067	0.000

## Decomposing an effect: summary

1. Factor variables and margins make the difficult decompositions trivial
2. Factor syntax understands interactions in model specifications
3. `margins` in turn understands interactions and handles the messy details



## \* Comparing ADCs across models

### 1. 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) in two models

## 2. 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

### 3. 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

### 4. Conclusion

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

# Conclusions

## Model interpretation and Stata

1. Too often interpretation ends with estimated coefficients
2. Interpretation using predictions is more informative
  - ▶ I think of regression coefficients as “nuisance parameters”
3. Methods of interpretation must be practical
4. `margins` makes hard things easy, very hard things merely hard
5. Hopefully, Stata 15 will make impossible things possible

# Conclusions

## Marginal effects are only one method of interpretation

1. Standard marginal effects are more useful than odds ratios
2. `mchange` allows marginal effects to be a routine part of analysis
3. Today's talk shows how to customize marginal effects for the substantive application
4. However, marginal effects are **not** the only or best method of interpretation
5. Tables and plots are often valuable (Long and Freese, 2014)
6. 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.