MRC Clinical Trials Unit

Handling interactions in Stata, especially with continuous predictors

Patrick Royston & Willi Sauerbrei

German Stata Users' meeting, Berlin, 1 June 2012

Interactions – general concepts

- General idea of a (two-way) interaction in multiple regression is **effect modification**:
  - $\eta(x_1,x_2) = f_1(x_1) + f_2(x_2) + f_3(x_1,x_2)$
- Often, $\eta(x_1,x_2) = E(Y \mid x_1,x_2)$, with obvious extension to GLM, Cox regression, etc.
- Simplest case: $\eta(x_1,x_2)$ is linear in the x's and $f_3(x_1,x_2)$ is the product of the x's:
  - $\eta(x_1,x_2) = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$
- Can extend to more general, non-linear functions

The simplest type of interaction

- Binary x binary
- E.g. in the MRC RE01 trial in kidney cancer
- 12 month % survival since randomisation
- Substantial treatment effect in patients with low WCC
- Little or no treatment effect in those with high WCC
- But really, WCC is a continuous variable …

Treatment group	White cell count low (<=10)	White cell count high (>10)
MPA	34% (se 4)	24% (se 4)
Interferon	49% (se 4)	21% (se 7)

Overview

- Interactions and factor variables (Stata 11/12)
  - Note: I am not an expert on factor variables! I sometimes use them.
- General interactions between continuous covariates in observational studies
  - Focus on continuous covariates …
  - … because people don't appear to know how to handle them!
- Special case: interactions between treatment and continuous covariates in randomized controlled trials

Interactions and factor variables

Scope

- We introduce the topic with a brief introduction to factor variables
- In this part, we consider only linear interactions:
  - Binary x binary (2 x 2 table)
  - Binary x continuous
  - Continuous x continuous

Factor variables: brief notes

- Implemented via prefixes (unary operators) and binary interaction operators
  - see **`help fvvarlist`**
- There are four factor-variable operators:

Operator	Description
i.	unary operator to specify indicators (dummies)
c.	unary operator to treat as continuous
#	binary operator to specify interactions
##	binary operator to specify factorial interactions

- Dummy variables are 'virtual' – not created *per se*
- Names of regression parameters easily found by inspecting the post-estimation result matrix **`e(b)`**

Factor variables: `i.` prefix

- Example from Stata manual [U]11.4.3:

```
. list group i.group in 1/5
  group 1b.group  2.group  3.group
1.     1         0         0         0
2.     1         0         0         0
3.     2         0         1         0
4.     2         0         1         0
5.     3         0         0         1
```

Example dataset

- MRC RE01 trial in advanced kidney cancer
- Of 347 patients, only 7 censored, the rest died
- For simplicity, as a continuous response variable, Y, we use months to death, _t
  - Ignore the small amount of censoring
- There are several prognostic factors that may influence time to death
- Some are binary, some categorical, some continuous

Example: factor variable parameters

```
. regress _t i.who

      Source |       SS       df       MS              Number of obs =     347
-------------+------------------------------           F(  2,   344) =   15.72
       Model |  7780.81413     2  3890.40707            Prob > F      =  0.0000
    Residual |  85126.5686   344  247.460955            R-squared     =  0.0837
-------------+------------------------------           Adj R-squared =  0.0784
       Total |  92907.3828   346  268.518447            Root MSE      =  15.731

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         who |
          1  |    -4.2783   2.026215    -2.11   0.035    -8.263629   -.2929707
          2  |  -12.94365   2.354542    -5.50   0.000    -17.57476   -8.312534
             |
       _cons |   19.17358   1.622518    11.82   0.000     15.98227    22.36488
------------------------------------------------------------------------------

. matrix list e(b)

e(b)[1,4]
             0b.           1.           2.
            who          who          who        _cons
y1            0   -4.2782996  -12.943646   19.173577
```

Basic analysis to understand binary x binary: the 2 x 2 table of means

- Example:

```
. table rem sex, contents(mean _t) format(%6.2f)

----------------------------
x6:         |
kidney      |
removed     |     x2: sex
(Y/N)       |    male   female
------------+---------------
          0 |   13.50     9.34
          1 |   13.77    18.19
----------------------------
```

Fitting an interaction model

- Consider 3 methods:
  - Method 1: binary operator to specify interactions
    - **`regress _t rem#sex`**
  - Method 2: binary operator to specify factorial interactions
    - **`regress _t rem##sex`**
  - Method 3: create multiplicative term(s) yourself
    - **`gen byte remsex = rem * sex`**
    - **`regress _t rem sex remsex`**
- Models are identical - all give the same fitted values
- Parameterisation of method 1 is different

Method 1: Binary operator

```
. regress _t rem#sex

      Source |       SS       df       MS              Number of obs =     347
-------------+------------------------------           F(  3,   343) =    2.73
       Model |  2168.10384     3  722.701279            Prob > F      =  0.0437
    Residual |  90724.0188   343  264.501513            R-squared     =  0.0233
-------------+------------------------------           Adj R-squared =  0.0148
       Total |  92892.1227   346  268.474343            Root MSE      =  16.264

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     rem#sex |
        0 1  |  -4.160995   2.888457    -1.44   0.151    -9.842314    1.520323
        1 0  |   .2724983   2.137064     0.13   0.899    -3.930903    4.475899
        1 1  |    4.68417   2.581164     1.81   0.070    -.3927323    9.761072
             |
       _cons |   13.49939   1.610327     8.38   0.000     10.33204    16.66675
------------------------------------------------------------------------------
```

Notes on parameters:

rem#sex 0 1 (-4.16) is (sex=1) - (sex=0) at rem=0

rem#sex 1 0 (+0.27) is (rem=1) - (rem=0) at sex=0

rem#sex 1 1 (+4.68) is [(rem=1) | sex=1] - [(rem=0) | sex=0]

_cons (13.50) is intercept (mean of Y at rem=0 & sex=0)

I don't recommend this parameterisation!

Method 2: Binary operator

```
. regress _t rem##sex

      Source |       SS       df       MS              Number of obs =     347
-------------+------------------------------           F(  3,   343) =    2.73
       Model |  2168.10384     3  722.701279            Prob > F      =  0.0437
    Residual |  90724.0188   343  264.501513            R-squared     =  0.0233
-------------+------------------------------           Adj R-squared =  0.0148
       Total |  92892.1227   346  268.474343            Root MSE      =  16.264

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       1.rem |   .2724983   2.137064     0.13   0.899    -3.930903    4.475899
       1.sex |  -4.160995   2.888457    -1.44   0.151    -9.842314    1.520323
             |
     rem#sex |
        1 1  |   8.572667   3.792932     2.26   0.024     1.112333      16.033
             |
       _cons |   13.49939   1.610327     8.38   0.000     10.33204    16.66675
------------------------------------------------------------------------------
```

Notes on parameters:

1.rem (0.27) is (rem=1) - (rem=0) at sex=0

1.sex (-4.16) is (sex=1) - (sex=0) at rem=0

rem#sex (+8.57) is [(rem=1) - (rem=0) at sex=1] - [(rem=1) - (rem=0) at sex=0]

_cons (13.50) is intercept (mean of Y at rem=0 & sex=0)

This is a 'standard' parameterisation with P-value as given above

Method 3: DIY multiplicative term

```
. generate byte remsex = rem * sex

. regress _t rem sex remsex
```

```
      Source |       SS       df       MS              Number of obs =     347
-------------+------------------------------           F(  3,   343) =    2.73
       Model |  2168.10384     3  722.701279            Prob > F      =  0.0437
    Residual |  90724.0188   343  264.501513            R-squared     =  0.0233
-------------+------------------------------           Adj R-squared =  0.0148
       Total |  92892.1227   346  268.474343            Root MSE      =  16.264

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         rem |   .2724983   2.137064     0.13   0.899    -3.930903    4.475899
         sex |  -4.160995   2.888457    -1.44   0.151    -9.842314    1.520323
      remsex |   8.572667   3.792932     2.26   0.024     1.112333      16.033
       _cons |   13.49939   1.610327     8.38   0.000     10.33204    16.66675
------------------------------------------------------------------------------
```

Notes on parameters:
rem is 1.rem in Method 2
sex is 1.sex in Method 2
remsex is rem#sex in Method 2
This is the same parameterisation as Method 2

Interactions in non-normal errors models

Key points:

1. In a 2 x 2 table, an interaction is a ‘difference of differences’
2. Tabulate the 2 x 2 table of mean values of the linear predictor
3. May back-transform values via the inverse link function
   - e.g. exponentiation in hazards models

Binary x continuous interactions

- Use `c.` prefix to indicate continuous variable
- Use the ## operator

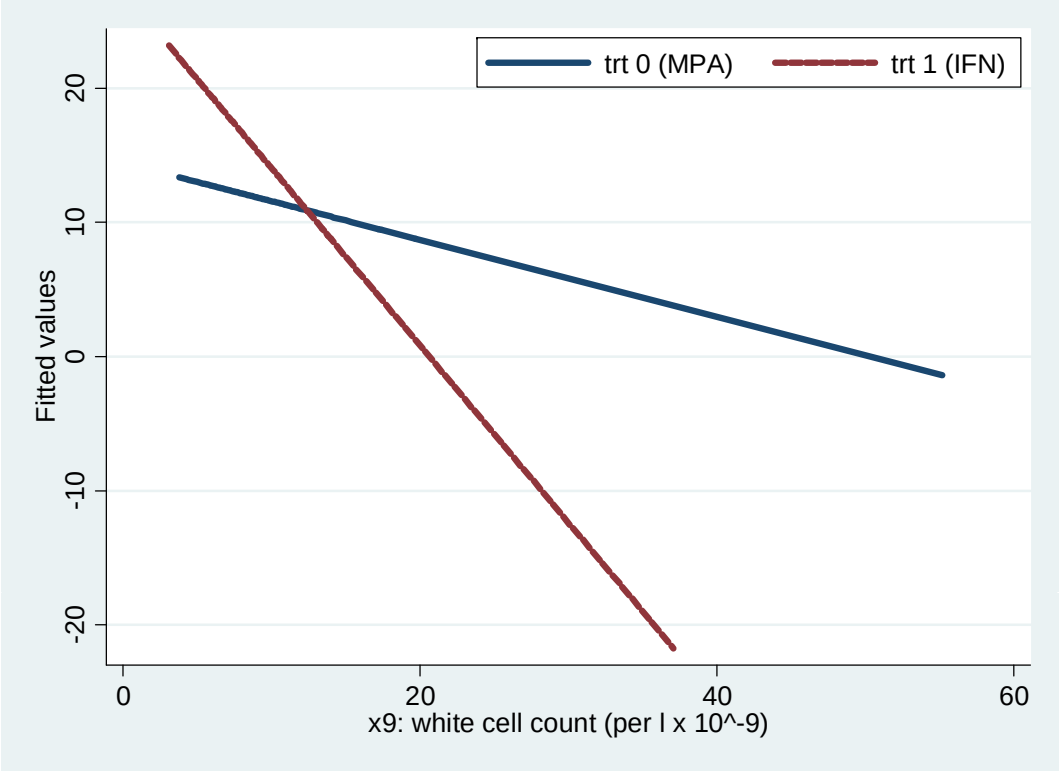
```
. regress _t trt##c.wcc

      Source |       SS       df       MS              Number of obs =     347
-------------+------------------------------           F(  3,   343) =    7.44
       Model |  5678.62935     3  1892.87645            Prob > F      =  0.0001
    Residual |  87228.7534   343  254.311234            R-squared     =  0.0611
-------------+------------------------------           Adj R-squared =  0.0529
       Total |  92907.3828   346  268.518447            Root MSE      =  15.947

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       1.trt |   12.81405   4.124167     3.11   0.002     4.702208    20.92589
         wcc |  -.2867831   .2741174    -1.05   0.296    -.8259457    .2523796
             |
   trt#c.wcc |
           1 |  -1.034239   .4327233    -2.39   0.017    -1.885365   -.1831142
             |
       _cons |   14.45292   2.712383     5.33   0.000     9.117919    19.78791
------------------------------------------------------------------------------
```

Binary x continuous interactions (cont.)

- The main effect of **`wcc`** is the slope in group 0
- The interaction parameter is the difference between the slopes in groups 1 & 0
- Test of **`trt#c.wcc`** provides the interaction parameter and test
- Results are nicely presented graphically
  - Predict linear predictor **`xb`**
  - Plot **`xb`** by levels of the factor variable
  - Also, 'treatment effect plot' (*coming later*)

Plotting a binary x continuous interaction

```
. regress _t trt##c.wcc
. predict fit
. twoway (line fit wcc if trt==0, sort) (line fit wcc
   if trt==1, sort lp(-)), legend(lab(1 "trt 0 (MPA)")
   lab(2 "trt 1 (IFN)") ring(0) pos(1))
```



Continuous x continuous interaction

- Just use **`c.`** prefix on each variable

```
. regress _t c.age##c.t_mt

      Source |       SS       df       MS              Number of obs =     347
-------------+------------------------------           F(  3,   343) =   10.35
       Model |  7714.26052     3  2571.42017            Prob > F      =  0.0000
    Residual |  85193.1223   343   248.37645            R-squared     =  0.0830
-------------+------------------------------           Adj R-squared =  0.0750
       Total |  92907.3828   346  268.518447            Root MSE      =   15.76

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   .0719063   .0876542     0.82   0.413    -.1005011    .2443137
        t_mt |   .0659781   .0128802     5.12   0.000      .040644    .0913122
             |
c.age#c.t_mt |  -.0008783   .0001861    -4.72   0.000    -.0012443   -.0005124
             |
       _cons |   8.055213   5.256114     1.53   0.126     -2.28306    18.39349
------------------------------------------------------------------------------
```

Continuous x continuous interaction

- Results are best explored graphically
- Consider in more detail next

Continuous x continuous interactions

Motivation: continuous x continuous intn.

- Many people only consider linear by linear interactions
- Not sensible if main effect of either variable is non-linear
- Mismodelling the main effect may introduce spurious interactions
  - E.g. false assumption of linearity can create a spurious linear x linear interaction
- Or they categorise the continuous variables
  - Many problems, including loss of power

The MFPIgen approach (1)

- MFP = multivariable fractional polynomials
- I = interaction
- gen = general
- Fractional polynomials (FPs) can be used to model relationships that may be non-linear
- In Stata, FPs are implemented through the standard **`fracpoly`** and **`mfp`** commands
- MFPIgen is implemented through a user-written command, **`mfpigen`**

The MFPIgen approach (2)

- MFPIgen aims to identify non-linear main effects and their two-way interactions
- Assume x_1, x_2 continuous and z confounders
- Apply MFP to x_1 and x_2 and z
  - Force x_1 and x_2 into the model
  - FP functions $FP_1(x_1)$ and $FP_2(x_2)$ are selected for x_1 and x_2
  - Linear functions could be selected
- Add term $FP_1(x_1) \times FP_2(x_2)$ to the model chosen
- Apply likelihood ratio test of interaction

The MFPIgen approach in practice

- Start with a list of covariates
- Check all pairs of variables for an interaction
- Simultaneously, apply MFP to adjust for confounders
- Use a low significance level to detect interactions, e.g. 1%
- Present interactions graphically
- Check interactions for artefacts graphically
- Use forward stepwise if more than one interaction remains

Example: Whitehall 1

- Prospective cohort study of 17,260 Civil Servants in London
- Studied various standard risk factors for common causes of death
- Also studied social factors, particularly job grade
- We consider 10-year all-cause mortality as the outcome
- Logistic regression analysis

Example: Whitehall 1 (2)

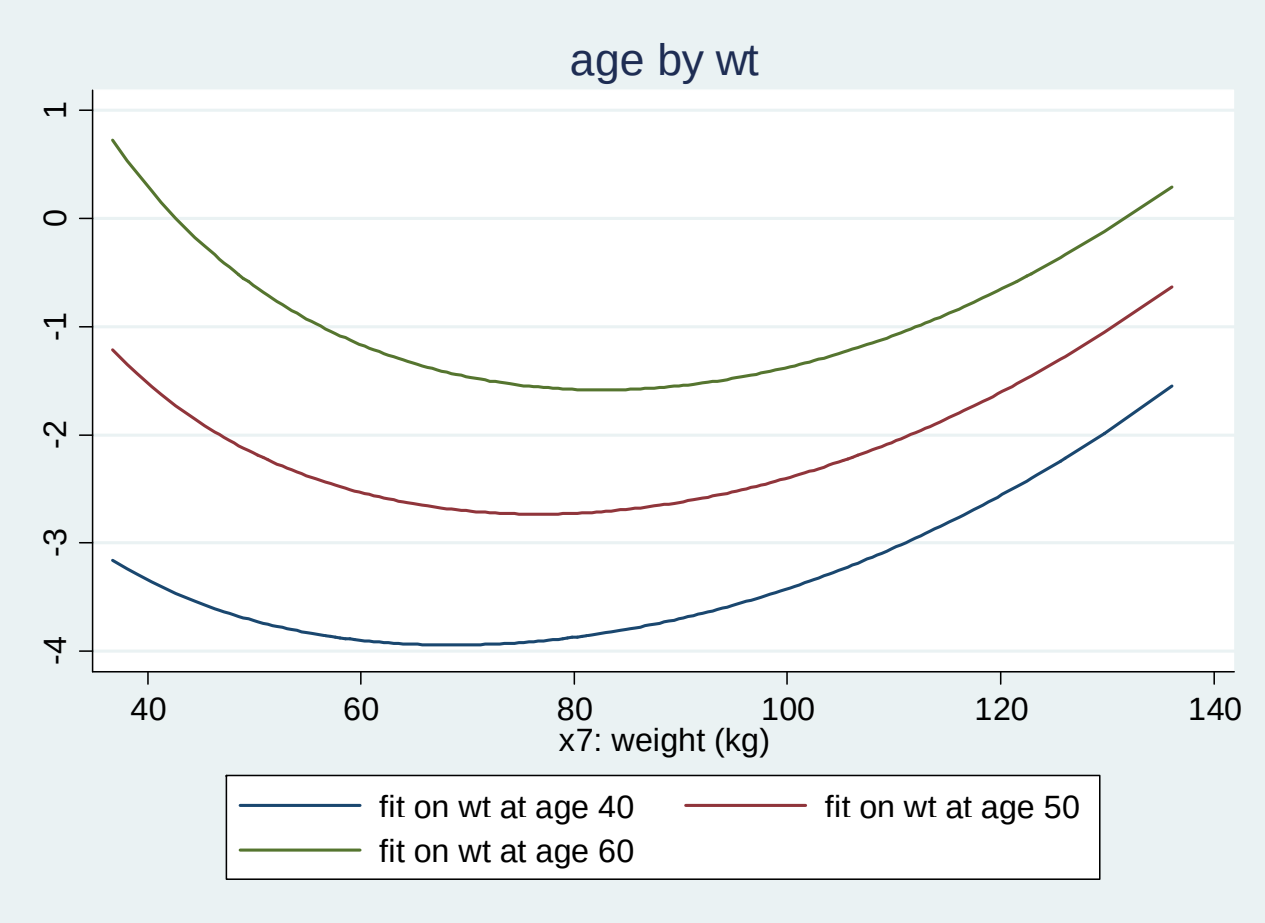
- Consider weight and age

```
. mfpigen: logit all10 age wt

MFPIGEN - interaction analysis for dependent variable all10
-------------------------------------------------------------------------------
variable 1   function 1   variable 2   function 2  dev. diff.  d.f.     P   Sel
-------------------------------------------------------------------------------
age          Linear        wt           FP2(-1 3)      5.2686     2   0.0718  0
-------------------------------------------------------------------------------
Sel = number of variables selected in MFP adjustment model
```

- Age function is linear, weight is FP2(-1, 3)
- No strong interaction (P = 0.07)

Plotting the interaction model

```
. mfpigen, fplot(40 50 60): logit all10 age wt
```



Mis-specifying the main effects function(s)

- Assume age and weight are linear
- The **`dfdefault(1)`** option imposes linearity

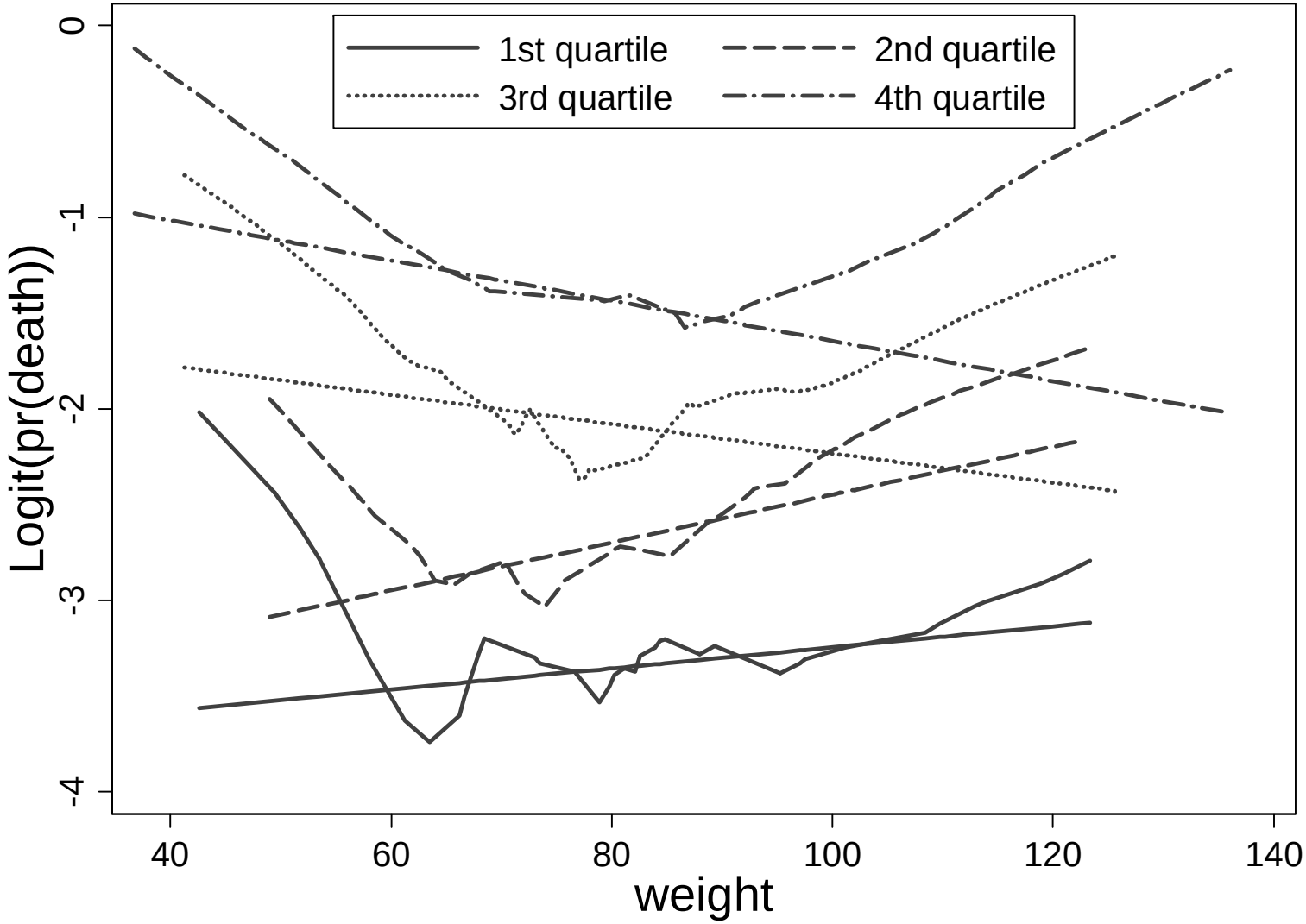
```
. mfpigen, dfdefault(1): logit all10 age wt

MFPIGEN - interaction analysis for dependent variable all10
-------------------------------------------------------------------------------
variable 1   function 1   variable 2   function 2  dev. diff.  d.f.     P   Sel
-------------------------------------------------------------------------------
age          Linear       wt           Linear         8.7375     1   0.0031  0
-------------------------------------------------------------------------------
Sel = number of variables selected in MFP adjustment model
```

- There appears to be a highly significant interaction (P = 0.003)

Checking the interaction model

- Linear age x weight interaction seems important
- Check if it's real, or the result of mismodelling
- Categorize age into (equal sized) groups
  - for example, 4 groups
- Compute running line smooth of the binary outcome on weight in each age group, transform to logits
- Plot results for each group
- Compare with the functions predicted by the interaction model

Whitehall 1: Check of age x weight linear interaction



Interpreting the plot

- Running line smooths are roughly parallel across age groups $\Rightarrow$ no (strong) interactions
- Erroneously assuming that the effect of weight is linear $\Rightarrow$ estimated slopes of weight in age-groups indicate strong interaction between age and weight
- We should have been more careful when modelling the main effect of weight

Whitehall 1: 7 variables, any interactions?

```
. mfpigen, select(0.05): logit all10 cigs
sysbp age ht wt chol i.jobgrade

MFPIGEN - interaction analysis for dependent variable all10
-------------------------------------------------------------------------------
variable 1   function 1   variable 2   function 2  dev. diff.  d.f.     P    Sel
-------------------------------------------------------------------------------
cigs         FP1(.5)      sysbp        FP2(-2 -2)     0.7961     2   0.6716  5
             FP1(.5)      age          Linear         0.0028     1   0.9576  5
             FP1(.5)      ht           Linear         2.1029     1   0.1470  5
             FP1(.5)      wt           FP2(-2 3)      0.1560     2   0.9249  5
             FP1(.5)      chol         Linear         1.7712     1   0.1832  5
             FP1(.5)      i.jobgrade   Factor         4.3061     3   0.2303  5

sysbp        FP2(-2 -2)   age          Linear         3.1169     2   0.2105  5
```

(remaining output omitted)

What **`mfpigen`** is doing

- FP functions for each pair of continuous variables are selected
  - Functions are simplified if possible
  - Closed test procedure in **`mfp`**
  - Controlled by the **`alpha()`** option
- The **`select(0.05)`** option tests confounders for inclusion in each interaction model at the 5% significance level
- The **`Sel`** column in the output shows how many variables are actually included in each confounder model

Results: P-values for interactions

Variable	cigs*	sysbp*	age	height	weight*	chol
cigs*	–					
sysbp*	0.7	–				
age	0.9	0.2	–			
height	0.1	0.5	1.0	–		
weight*	0.9	0.5	0.1	0.4	–	
chol	0.2	0.07	0.001	0.8	0.2	–
grade	0.2	0.2	0.2	0.2	0.04	0.4

*FP transformations were selected; otherwise, linear

Graphical presentation of age x chol interaction

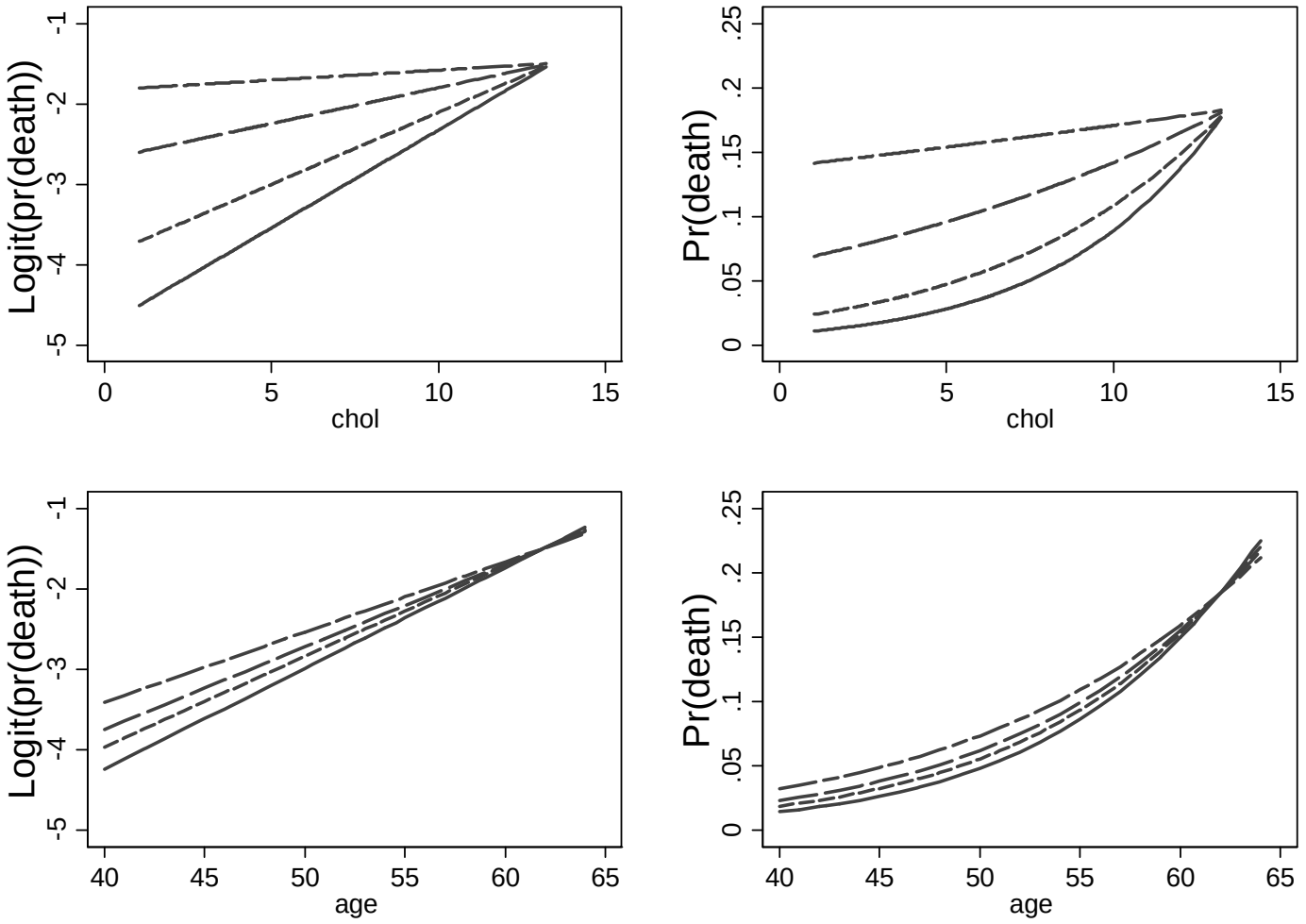
```
. fracgen cigs .5, center(mean)
. fracgen sysbp -2 -2, center(mean)
. fracgen wt -2 3, center(mean)

. mfpigen, linadj(cigs_1 sysbp_1 sysbp_2
> wt_1 wt_2 ht i.jobgrade) df(1)
> fplot(%10 35 65 90): logit all10 age chol
```

Alternatively:

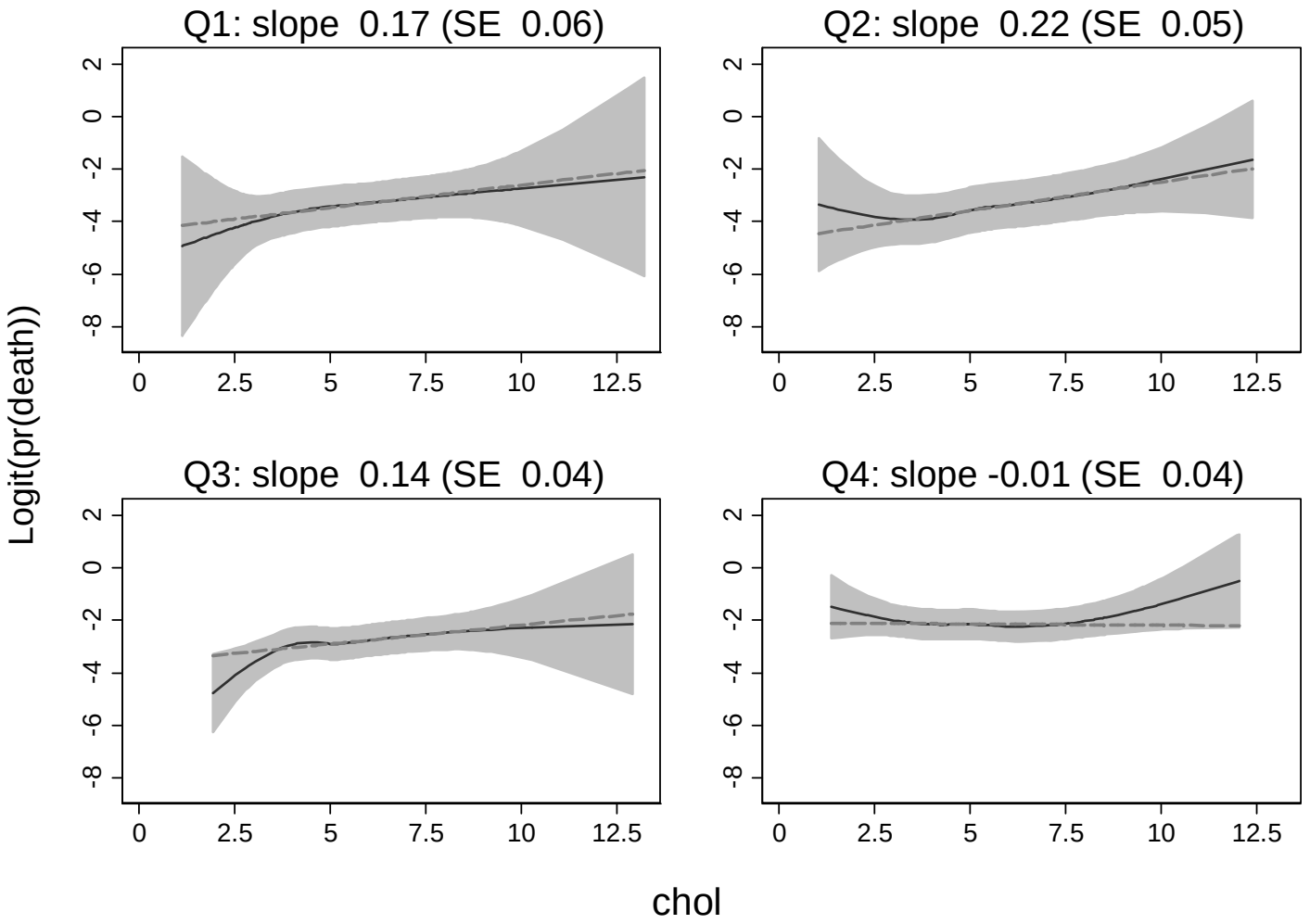
```
. logit all10 c.age##c.chol cigs_1 sysbp_1 sysbp_2
> wt_1 wt_2 ht i.jobgrade
. sliceplot age chol, sliceat(10 35 65 90) percent
```

- **`sliceplot`** is a new user-written command

Graphical presentation of age x chol intn.



Check of chol x age interaction



Interactions with continuous covariates in randomized trials

MFPI method (Royston & Sauerbrei 2004)

- Continuous covariate *x* of interest, binary treatment variable *t* and other covariates *z*
- Independent of *x* and *t*, use MFP to select an 'adjustment' (confounder) model *z** from *z*
- Find best FP2 function of *x* (in all patients) adjusting for *z** and *t*
- Test FP2(*x*) × *t* interaction (2 d.f.)
  - Estimate β's in each treatment group
  - Standard test for equality of β's
- May also consider simpler FP1 or linear functions – choose e.g. by min AIC

MFPI in Stata

- MFPI is implemented as a user command, **`mfpi`**
- **`mfpi`** is available on SSC
- Details are given by Royston & Sauerbrei, *Stata Journal* **9**(2): 230-251 (2009)
- Program was updated in 2012 to support factor variables

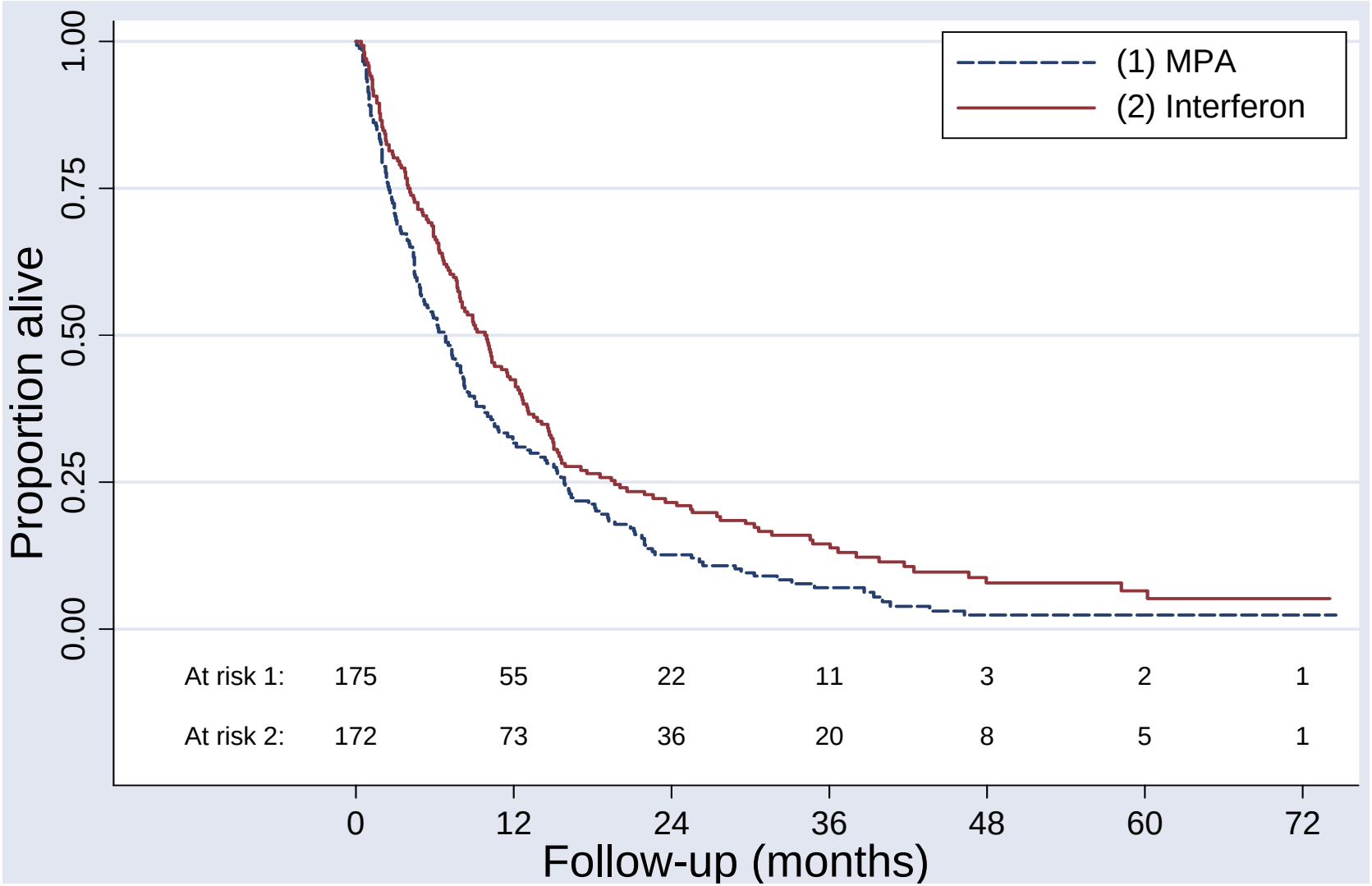
Treatment effect function

- Have estimated two FP2 functions – one per treatment group
- Plot the difference between functions against *x* to show the interaction
  - i.e. the treatment effect at different *x*
- Pointwise 95% CI shows how strongly the interaction is supported at different values of *x*
  - i.e. variation in the treatment effect with *x*

Example: MRC RE01 trial in kidney cancer

- Main analysis: Interferon improves survival
- HR: 0.76 (0.62 - 0.95), P = 0.015
- Is the treatment effect similar in all patients?
- Nine possible covariates available for the investigation of treatment-covariate interactions – only one is significant (WCC)

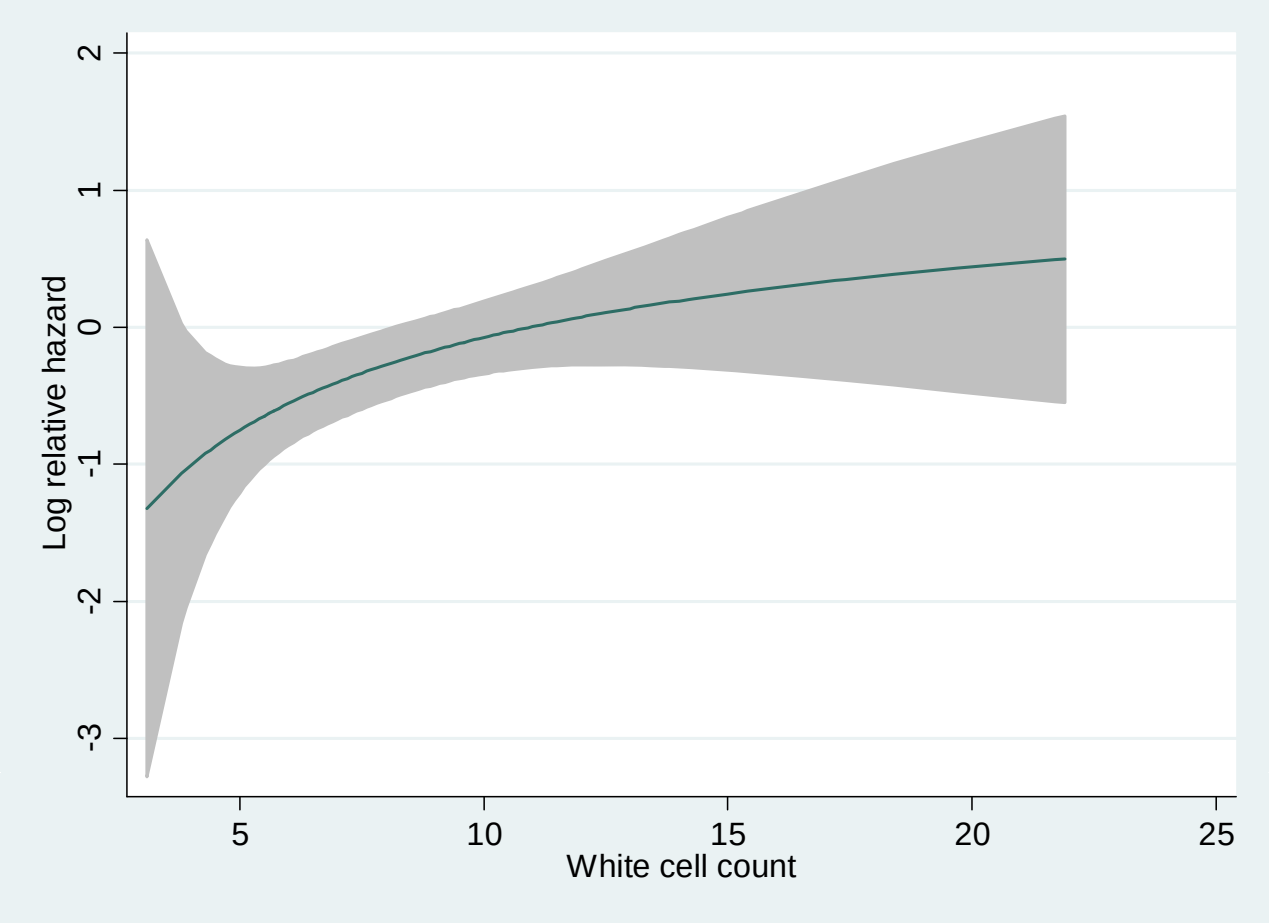
Kaplan-Meier showing treatment effect



The **mfpi** command

```
. mfpi, select(0.05) fp2(wcc) with(trt) gendiff(d): stcox
(whod1 whod2) t_dt t_mt rem mets haem

Interactions with trt (347 observations). Flex-1 model (least flexible)

-------------------------------------------------------------------------------
Var         Main         Interact     idf   Chi2     P      Deviance tdf   AIC
-------------------------------------------------------------------------------
wcc         FP2(-1 -.5) FP2(-1 -.5)    2    6.91   0.0316  3180.194  7  3194.194
-------------------------------------------------------------------------------
idf = interaction degrees of freedom; tdf = total model degrees of freedom

. mfpi_plot wcc
[using variables created by gendiff(d)]
```

Treatment effect plot for `wcc`



About 25% of patients, those with WCC > 10 seem not to benefit from interferon

Concluding remarks

- MFPIgen and MFPI should help researchers detect, model and visualize interactions with continuous covariates
- Usually, we are searching for interactions, so small P-values are required
- Other methods not considered
  - STEPP – mainly graphical
  - …

Thank you.

Cox (1984) paper: *Interaction*

- Cox identifies 3 types of variable that might appear in interactions:
- Treatment variables
  - Can be modified or imposed
  - Treatments, e.g. chemotherapy, surgery
  - Behaviours, e.g. smoking, drinking
- Intrinsic variables
  - Cannot be modified
  - Often demographic, e.g. sex, age
- Unspecific variables
  - e.g. structural blocks, `random' factors