The Stata Journal (2009)
9, Number 2, pp. 230–251

Two techniques for investigating interactions between treatment and continuous covariates in clinical trials

Patrick Royston
Cancer and Statistical Methodology Groups
MRC Clinical Trials Unit
London, UK
pr@ctu.mrc.ac.uk

Willi Sauerbrei
Institute for Medical Biometry and Medical Informatics
Freiburg University Medical Center
Freiburg, Germany
wfs@imbi.uni-freiburg.de

Abstract. There is increasing interest in the medical world in the possibility of tailoring treatment to the individual patient. Statistically, the relevant task is to identify interactions between covariates and treatments, such that the patient's value of a given covariate influences how strongly (or even whether) they are likely to respond to a treatment. The most valuable data are obtained in randomized controlled clinical trials of novel treatments in comparison with a control treatment. We describe two techniques to detect and model such interactions. The first technique, multivariable fractional polynomials interaction, is based on fractional polynomials methodology, and provides a method of testing for continuous-by-binary interactions and by modeling the treatment effect as a function of a continuous covariate. The second technique, subpopulation treatment-effect pattern plot, aims to do something similar but is focused on producing a nonparametric estimate of the treatment effect, expressed graphically. Stata programs for both of these techniques are described. Real data for brain and breast cancer are used as examples.

Keywords: st0164, mfpi, mfpi_plot, stepp_tail, stepp_window, stepp_plot, continuous covariates, treatment–covariate interaction, clinical trials, fractional polynomials, subpopulation treatment-effect pattern plot

1 Introduction

There is increasing interest in the medical world in the possibility of tailoring treatment to the individual patient. Recently (the field is still in its infancy), excitement has focused on the use of genetic testing to see if patients are likely to respond differently to given therapies according to their genetic makeup. More prosaically, there is increasing interest in examining data from randomized controlled clinical trials. Evidence of differential response to treatment is sought through interactions between treatment and covariates. Because treatment is, by design, asymptotically independent of covariates, such interactions, if they exist, should be essentially independent of other influential covariates.

One issue that has received little attention in the literature is how to handle interactions between continuous predictors and treatment. The traditional approach has been to categorize continuous factors and apply standard tests of interaction. However, categorization is inefficient, and the results may depend strongly on the cutpoints chosen. An alternative approach called multivariable fractional polynomials interaction (MFPI), proposed by Royston and Sauerbrei (2004), uses fractional polynomials for continuous predictors. A more exploratory technique, introduced by Bonetti and Gelber (2000) and further developed by Bonetti and Gelber (2004), is the subpopulation treatment-effect pattern plot (STEPP).

In this article, we aim to present software implementing the MFPI and STEPP algorithms and providing postestimation plots, with examples. We give a brief description of the two techniques here; detailed accounts can be found in the original articles and elsewhere.

2 MFPI

2.1 Default algorithm

To investigate possible interactions between treatment and continuous covariates, Royston and Sauerbrei (2004) proposed the MFPI algorithm as an extension of the multivariable fractional polynomial (MFP) algorithm (Sauerbrei and Royston 1999) for building regression models, by combining variable selection with determination of functional forms for continuous predictors. See also [R] **mfp** in the *Stata Reference Manual*. Variables are selected by backward elimination. The algorithm examines in a systematic fashion whether the effect of a continuous covariate is better modeled by a nonlinear member of the class of fractional polynomial (FP) functions or by a linear function.

An FP function with one power term is known as an FP1 function. It takes the form $\beta_1 x^{p_1}$, with the power, p_1, chosen from the set $S = (-2, -1, -0.5, 0, 0.5, 1, 2, 3)$, where x^0 denotes $\log x$ (Royston and Altman 1994). An FP function with two power terms is called an FP2 function and takes the form $\beta_1 x^{p_1} + \beta_2 x^{p_2}$, with p_1 and p_2 both chosen from S. In the mathematical limit as p_2 tends to p_1, a so-called "repeated-powers" FP2 function is obtained, taking the form $\beta_1 x^{p_1} + \beta_2 x^{p_2} \log x$. In all, there are 8 FP1 functions (including the linear function) and 36 FP2 functions (including eight repeated-powers functions).

The MFPI algorithm models the prognostic effect of z by FP2 transformations within treatment groups, but under the constraint of the same powers. As explained by Royston and Sauerbrei (2004), constraining the powers to be equal reduces instability due to overfitting in the fitted functions. This can be done in a univariate setting or by adjusting the model for other covariates. Assume that the treatment variable, t, has two levels, coded 0 and 1. The influence of covariate z on the estimated treatment effect is determined by $\widehat{f}(z) = \widehat{f}_1(z) - \widehat{f}_0(z)$, where $\widehat{f}_i(z)$, $i = 0, 1$, are the estimated functions for the prognostic effect of z in treatment group i. The plot of $\widehat{f}(z)$ together with a pointwise confidence interval is called a treatment-effect plot. Comparing the model with separate functions for z (i.e., functions with different βs, because the power terms are the same) in treatment groups with a "main" effects model with the same function (i.e., a function with an identical β) in both groups is a test of interaction. The difference in deviances is compared with χ^2 on 2 degrees of freedom (df). For the investigation of an interaction of treatment with binary or categorical variables, MFPI uses the usual likelihood-ratio test for an interaction in a model with main effects and multiplicative interaction terms. Categorical variables with greater than two levels require the usual assumptions and df. They cause the usual difficulties, including pairwise verses global comparisons and the need for trend tests for ordered categories.

MFPI allows adjustment for other variables in a multivariable setting in the context of different types of regression models. Royston and Sauerbrei (2004) propose the determination of an "adjustment" model as a preliminary step, preferably by MFP, without considering the covariate z. For more details, see Royston and Sauerbrei (2004).

2.2 Some variants

The algorithm just described was published by Royston and Sauerbrei (2004). However, some modifications are possible. For reasons that soon become apparent, let's term the published algorithm FLEX1 (i.e., the least flexible). The first variant (FLEX2) is to determine the best-fitting FP power(s) in the interaction model, constraining the powers of z to be the same for each level of the treatment variable. These same powers are then used for the main effect of z. For binary t, the $z \times t$ interaction model has 3 df (1 power, 2 βs) for FP1 functions and 6 df (2 powers, 4 βs) for FP2 functions of z. The corresponding main-effects models have 1 and 2 df. Therefore, the test of interaction has 2 and 4 df for FP1 and FP2 functions, respectively.

The second variant (FLEX3) takes the same approach to the interaction but reestimates the FP powers for the main effect. This increases the flexibility of the main effect function and thereby reduces the df for the interaction. The test of interaction has 1 and 2 df for FP1 and FP2 functions, respectively.

The third variant (FLEX4) optimizes the powers at all levels of the treatment variable for the interaction and for the main effect. The required powers in such a model actually have to be estimated by MFP, making FLEX4 the most computationally intensive of the 4 variants (the least intensive is FLEX1). The df for the test of interaction are the same as for FLEX2.

Significance tests for interactions in the FLEX2, FLEX3, and FLEX4 variants, based on the χ^2 distribution with the stated df, should be regarded as provisional. For example, the constraint on the powers for the interaction model in FLEX2 may affect the df but is not explicitly accounted for. Limited simulation studies have confirmed that likelihood-ratio tests for FLEX1 preserve the correct nominal size (Sauerbrei, Royston, and Zapien 2007). Simulation work is required to assess the type I error probabilities of the other three variants.

These variants are available via the `flex()` option of the `mfpi` routine, described in sections 4–6.

3 STEPP

The STEPP is based on dividing the observation into subgroups defined with respect to the covariate (z) of interest and estimating the effect of treatment (t) separately within each subpopulation. To increase the number of patients who contribute to each point estimate and hence to improve the precision of the individual estimates, subpopulations overlap.

Two ways of defining subpopulations are proposed, as indicated in figure 1.

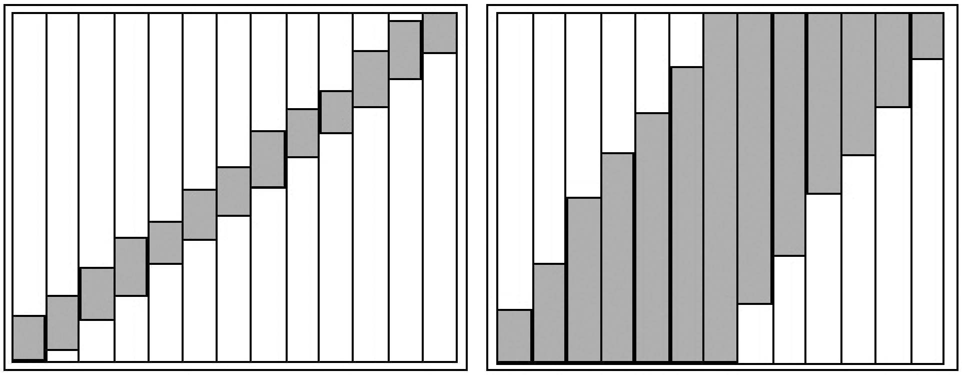

Figure 1. Diagrammatic representation of a STEPP. Left panel: sliding-window variant; right panel: tail-oriented variant. The horizontal axis indexes the various subpopulations (shaded area) for which treatment effects are estimated and shows the range of covariate values used to define the cohort of patients included in each subpopulation. Adapted from Sauerbrei, Royston, and Zapien (2007) by permission of Elsevier.

The horizontal axis in figure 1 indexes the various subpopulations for which treatment effects are estimated and shows the range of covariate values used to define the cohort of patients included in each subpopulation.

The tail-oriented (TO) variant has the overall population as the center group. With increasing distance from the center, more and more patients with high covariate values (to the left side) or low covariate values (to the right side) are deleted. Subpopulations in the sliding-window (SW) variant have an overlapping part and a part that differs between neighboring subpopulations. The number of subpopulations and the percentage of overlapping patients are important parameters of this variant. To define the size of the subpopulations, the SW variant has two parameters, n_1 and n_2. A subpopulation must have at least n_2 patients, of which at least $n_2 - n_1$ patients are required to be different between neighboring subpopulations. Therefore, n_2 must exceed n_1. The amount of discreteness of the continuous variable determines the size of each subpopulation. The TO variant has a parameter, g, giving $g - 1$ subpopulations where patients with larger values are eliminated and $g - 1$ subpopulations excluding patients with smaller values, a total of $2g - 1$ subpopulations. For further details on how the subpopulations are created, see Bonetti and Gelber (2000).

If there is no interaction between z and t, the estimated treatment effects in the subpopulations defined by z should be similar to the treatment effect in the overall population. Plots showing the estimated treatment effect, with confidence intervals, in the subpopulations and tests based on the deviation of treatment effects in the subpopulations from those in the overall population are suggested for the investigation of an interaction between z and t. Each z-based subgroup is represented by the mean of its z values. For more details, see Bonetti and Gelber (2004). We considered several tests; because testing seems to be a topic of active research, we did not include any tests in the present software.

Unlike with MFPI, although STEPP is defined in terms of a treatment effect, t, that varies with z, there is no requirement that t be a discrete variable. STEPP could be used equally well to investigate an interaction between z (modeled “nonparametrically”) and a continuous variable, t (modeled linearly). The plot would show the variation in a regression coefficient estimated in the various subgroups created by STEPP. As with MFPI, for a multilevel treatment variable, STEPP can be applied to produce a separate treatment-effect plot for each treatment in comparison with the chosen base level of t.

4 Syntax

```
mfpi regression_cmd [yvar] [xvarlist] [if] [in] [weight], with(withvar)
   [adjust(adj_list) adjvars(varlist) all detail df(df_list) flex(#)
   fp1(fp1_varlist) fp2(fp2_varlist) gendiff(stubname) genf(stubname)
   linear(linear_varlist) mfpopts(mfp_options) noscaling select(select_list)
   showmodel topcoded regression_cmd_options]
```

```
mfpi_plot varname [if] [in], stubname(stubname) [vn(#) level(#)
   plot(plot) graph_options]
```

```
stepp_tail regression_cmd [yvar] zvar [covarlist] [if] [in] [weight],
   gen(stubname) g(#) with(varlist) [regression_cmd_options]
```

`stepp_window` *regression_cmd* [*yvar*] *zvar* [*covarlist*] [*if*] [*in*] [*weight*] `,`
`gen(`*stubname*`)` `n1(`*#*`)` `n2(`*#*`)` `with(`*varlist*`)` [*regression_cmd_options*]

`stepp_plot` *stubname* [`,` `vn(`*#*`)` `plot(`*plot*`)` *graph_options*]

where *regression_cmd* may be `clogit`, `cnreg`, `glm`, `intreg`, `logistic`, `logit`, `mlogit`, `nbreg`, `ologit`, `oprobit`, `poisson`, `probit`, `qreg`, `regress`, `rreg`, `stcox`, `stpm`, `stpm2`, `streg`, or `xtgee`. All weight types supported by *regression_cmd* are allowed.

stubname for `stepp_plot` is as specified in the `gen(`*stubname*`)` option of `stepp_tail` and `stepp_window` (see also the `vn()` option described in section 6.2). *varname* for `mfpi_plot` is a member of *linear_varlist*, *fp1_varlist*, or *fp2_varlist*.

5 Description

`mfpi` is designed to investigate the interaction of a categorical covariate (*withvar*) with a covariate(s) specified by any combination of the `fp1()`, `fp2()`, or `linear()` options. Often *withvar* will be the treatment variable in a randomized controlled trial of one or more treatments against the control.

`mfpi_plot` produces a treatment-effect plot derived from variables saved with the `gendiff()` option of `mfpi`. With more than two treatments, the control arm is taken as the group with the lowest value of *withvar*. Only pairwise comparisons with this level are supported.

`stepp_tail` and `stepp_window` compute STEPP estimators for graphical exploration of a treatment–covariate interaction. `stepp_tail` provides the TO estimator, and `stepp_window` provides the SW estimator. *zvar* is the continuous covariate whose interaction with the treatment is to be studied, and *covarlist* is a list of other covariates used to adjust each fitted model to the treatment variable(s) defined by `with()`. The results can be plotted with `stepp_plot`.

6 Options

6.1 Options for mfpi

`with(`*withvar*`)` is required and defines the categorical variable whose interactions with variables specified in `linear()`, `fp1()`, or `fp2()` are of interest. *withvar* must have at least two distinct, nonmissing values, but the codes are arbitrary because they are mapped to 0, 1, 2, etc., internally. The category corresponding to the lowest value is taken as the reference category (level 0).

`adjust(`*adj_list*`)` determines the adjustment of FP-transformed variables. The most likely requirement is to suppress adjustment for all such variables, which is done with `adjust(no)`. The default behavior is adjustment to the mean of each continuous predictor. For further details, see the description of the same option in [R] **fracpoly**.

`adjvars(`*varlist*`)` includes *varlist* as linear main-effects terms in all the fitted models.

`all` allows prediction (see the `gendiff()` and `genf()` options) for all available cases, irrespective of exclusion by `if`, `in`, or weights.

`detail` gives additional details of the fitted models.

`df(`*df_list*`)` sets up the df for each predictor in *xvarlist*. The df (not counting the regression constant, `_cons`) are twice the degree of the FP, so, for example, a member of *xvarlist* fit as a second-degree FP (FP2) has 4 df. The first item in *df_list* can be either # or *varlist*`:`#. Subsequent items must be *varlist*`:`#. Items are separated by commas, and *varlist* is specified in the usual way for variables. With the first type of item, the df for all predictors are taken to be #. With the second type of item, all members of *varlist* (which must be a subset of *xvarlist*) have # df.

The default df for a predictor of type *varlist* specified in *xvarlist* but not in *df_list* are assigned according to the number of distinct (unique) values of the predictor, as follows:

No. of distinct values	Default df
1	(invalid—covariate has variance 0)
2–3	1
4–5	min(2, `dfdefault(`#`)`)
≥ 6	`dfdefault(`#`)`

`dfdefault(`#`)` is an option of `mfp` (see the `mfpopts()` option and `help mfp`); the default is `dfdefault(4)`, meaning an FP2 function.

Here are some examples of `df()`:

- In `df(4)`, all variables have 4 df.
- In `df(2, weight displ:4)`, `weight` and `displ` have 4 df, and all other variables have 2 df.
- In `df(1, weight displ:4, mpg:2)`, `weight` and `displ` have 4 df, `mpg` has 2 df, and all other variables have 1 df.
- In `df(weight displ:4, 2)`, all variables have 2 df because the final 2 overrides the earlier 4.

`flex(`#`)` defines the flexibility of the main-effects and interaction models, where $\# = 1$ is the least flexible and $\# = 4$ is the most flexible (see section 2.2). The default is `flex(1)`.

`fp1(`*fp1_varlist*`)` defines a list of continuous variables whose interactions with *withvar* are to be investigated by fitting FP functions of degree 1 (i.e., FP1 functions) to each member of *fp1_varlist* in turn, at each level of *withvar*. Also see `flex()`.

`fp2(`*fp2_varlist*`)` defines a list of continuous variables whose interactions with *withvar* are to be investigated by fitting FP functions of degree 2 (i.e., FP2 functions) to each member of *fp2_varlist* in turn, at each level of *withvar*. Also see `flex()`.

`gendiff(`*stubname*`)` generates a new variable(s) called *stubname#_j* that contains *fj* − *f0*, the difference between the estimated functions (at level *j* minus level 0 of *withvar*), for the #th member of the list composed of *linear_varlist*, *fp1_varlist*, and *fp2_varlist*, in that order. The difference *fj* − *f0* is an estimate of the covariate-specific effect of level *j* compared with level 0 (e.g., the covariate-specific treatment effect). `gendiff()` also creates new variables called *stubname#*`s_`*j*, which contains the standard error of *fj* − *f0*, and *stubname#*`lb_`*j* and *stubname#*`ub_`*j*, the lower and upper 95% confidence limits, thus providing the quantities necessary for a treatment-effect plot.

`genf(`*stubname*`)` generates new variables called *stubname#*`_0`, *stubname#*`_1`, etc., that contain the fitted functions at levels 0, 1, etc., of *withvar*, respectively, for the #th member of the list composed of *linear_varlist*, *fp1_varlist*, and *fp2_varlist*, in that order. For variables in *fp1_varlist* and *fp2_varlist*, the same FP transformation is used at each level of *withvar*. The estimated function at level 0 of *withvar* is adjusted to have mean 0.

`linear(`*linear_varlist*`)` defines a list of variables whose linear interactions with *withvar* are to be investigated. If a categorical variable in *linear_varlist* has more than two levels, the necessary dummy variables must be created and placed between parentheses to indicate that they should be tested together. More properly, *linear_varlist* is a list of variables of which some may be dummies for categorical variables.

For example, `linear((who2 who3))` binds binary predictors (dummy variables) `who2` and `who3` together to create a predictor with 2 df for its main effect.

`mfpopts(`*mfp_options*`)` supplies `mfp` options to `mfpi` for the creation of the adjustment model from *xvarlist*.

`noscaling` suppresses scaling of all the continuous variables that are subject to FP transformation. The default is automatic application of scale factors; the behavior can be turned off for all such variables by using `noscaling`. For further details, see the description of the same option for `fracpoly` and `fracgen` in [R] **fracpoly**.

`select(`*select_list*`)` sets the nominal *p*-values (significance levels) for variable selection among *xvarlist* by backward elimination. See the `select()` option of `mfp` for further details. A typical usage is `select(0.05)`, which selects all variables in *xvarlist* that are significant at the 5% level according to a backward stepwise algorithm.

`showmodel` shows the variables selected from *xvarlist* by `mfp`, together with their FP powers, where relevant. `showmodel` is a concise alternative to `detail`.

`topcoded` top-codes the `with()` variable, that is, the reference category is the highest value of *withvar*. The default is bottom-coding, that is, the reference category is the lowest value of *withvar*.

regression_cmd_options are any options for *regression_cmd*.

6.2 Options for mfpi_plot

`stubname(`*stubname*`)` is required if the `gendiff()` option to `mfpi` was not specified. If the `gendiff()` option to `mfpi` was specified, then *stubname* is the string specified in `gendiff()`.

`vn(`#`)` identifies the variable in `linear()`, `fp1()`, and `fp2()`, in that order, whose associated treatment effect is to be plotted. Only one variable can be plotted in a given call to `mfpi_plot`. For example, if `linear(x1 x2) fp1(x1 x3) fp2(x3)` was specified, the list of variables is `x1 x2 x1 x3 x3`, and the total number of variables is 5 (not 3). Thus # would be an integer between 1 and 5. The default is `vn(1)`.

`level(`#`)` defines the desired comparison between levels of *withvar*. Levels of *withvar* are coded 0, 1, 2, etc., with the reference category being level 0. For example, specifying `level(2)` would give a treatment-effect plot comparing level 2 of *withvar* with level 0. The default is `level(1)`.

`plot(`*plot*`)` provides a way to add other plots to the generated graph.

graph_options are any options of `graph twoway`, such as `xtitle()` and `ytitle()`.

6.3 Options for stepp_tail and stepp_window

`gen(`*stubname*`)` creates five new variables: *stubname*`b`, *stubname*`se`, *stubname*`mean`, *stubname*`lb`, and *stubname*`ub`. *stubname*`b` is the estimated regression coefficient in each subpopulation; *stubname*`se` is its standard error; *stubname*`mean` contains the mean of *zvar* in each subpopulation; and *stubname*`lb` and *stubname*`ub` are pointwise 95% confidence limits for *stubname*`b`. If `with()` includes more than one variable, the created variables have 2, 3, etc., appended to the names, e.g., *stubname*`b2`. The confidence level of the intervals can be altered by using the standard Stata `set level` # command.

`g(`#`)` (`stepp_tail` only) defines the number of subpopulation groups. The actual number of subpopulations used is $2\times\# - 1$.

`n1(`#`)` (`stepp_window` only) defines the number of individuals belonging to only one of two neighboring subpopulations.

`n2(`#`)` (`stepp_window` only) defines the number of individuals in a subpopulation. The overlap between two neighboring subpopulations is `n2()` minus `n1()` individuals.

`with(`*varlist*`)` defines the list of variables whose interactions with *zvar* are to be studied. Typically, *varlist* will comprise just one binary variable, representing the two arms of a parallel-group clinical trial.

regression_cmd_options are any options for *regression_cmd*.

6.4 Options for stepp_plot

`vn(#)` specifies an integer defining the variable number in `with()`, when more than one variable is specified. When only one variable is specified, `vn()` is optional.

`plot(`*plot*`)` provides a way to add other plots to the generated graph.

graph_options are any options of `graph twoway`, such as `xtitle()` and `ytitle()`.

7 Example 1: Glioma study

7.1 MFPI analysis

As with many Stata commands, `mfpi` is not as complex as it first appears in a formal description. It is best explained through illustrative examples.

A randomized trial to compare two chemotherapy regimes included 447 patients with malignant glioma, an aggressive type of brain cancer. At the time of the analysis, 293 patients had died, and the median survival time from the date of randomization was about 11 months. Survival times are analyzed with the Cox model. Apart from therapy, data on several variables that might affect survival time were recorded (see table 1).

Table 1. Glioma data

Name	Details	Name	Details
`sex`	Sex	`cort`	Cortisone (Y/N)
`time`	Interval to diagnosis	`epi`	Epilepsy (Y/N)
	(1 = short, 2 = long)	`amnesia`	Amnesia (Y/N)
`gradd1`	Malignancy grade[†]	`ops`	Organic psychosyndrome (Y/N)
`gradd2`	...	`aph`	Aphasia (Y/N)
`age`	Age, yr (continuous)	`karno`	Karnofsky index (continuous)
`surgd1`	Resection type[†]	`therapy`	Randomized treatment
`surgd2`	...	`survtime`	Time to death, days[*]
`convul`	Convulsions (Y/N)	`cens`	Censoring[*](0 = censored, 1 = died)

[*] response variable

[†] categorical predictor represented by two or more dummy variables and considered as an independent predictor

The two categorical variables, each with three levels (the type of surgical resection and the grade of malignancy), were each represented by two dummy variables. Here we consider the Karnofsky index as a continuous variable (it has 13 distinct values). Complete data on these predictors were available for 411 patients (274 events) and are used here. The study has previously been used in methodological investigations (e.g., Ulm et al. [1989]; Sauerbrei and Schumacher [1992]).

Suppose we wish to investigate interactions between `therapy` and the two available continuous predictors, `age` and `karno`. We apply `mfpi` to these variables and include all the other variables as possible confounders. We use the `select(0.05)` option to select the confounder model at the 5% significance level. The list of candidate variables for the confounder model includes `age` and `karno`; each of these variables is automatically removed from the confounder model when its interaction with `therapy` is considered, and the parameters for the other variables are reestimated. We consider linear, FP1, and FP2 as possible models for `age` and `karno`:

```
. use glioma
(Glioma, complete cases+cont var)
. mfpi stcox sex time gradd1 gradd2 age karno surgd1 surgd2 convul cort epi
> amnesia ops aph, with(therapy) linear(age karno) fp1(age karno) fp2(age karno)
> select(.05) showmodel
Variables in adjustment model
─────────────────────────────

      sex: not selected
     time: not selected
   gradd1: power(s) = 1
   gradd2: not selected
      age: power(s) = 1
    karno: not selected
   surgd1: power(s) = 1
   surgd2: not selected
   convul: not selected
     cort: not selected
      epi: power(s) = 1
  amnesia: not selected
      ops: not selected
      aph: not selected
Interactions with therapy (411 observations). Flex-1 model (least flexible)
───────────────────────────────────────────────────────────────────────────────
Var         Main        Interact     idf   Chi2     P      Deviance tdf   AIC
───────────────────────────────────────────────────────────────────────────────
age         Linear      Linear        1    3.19   0.0740   2684.515  3  2690.515
karno       Linear      Linear        1   13.65   0.0002   2669.695  3  2675.695
age         FP1(2)      FP1(2)        1    2.54   0.1107   2684.298  4  2692.298
karno       FP1(-2)     FP1(-2)       1    8.04   0.0046   2674.981  4  2682.981
age         FP2(1 2)    FP2(1 2)      2    2.96   0.2275   2683.881  7  2697.881
karno       FP2(-2 3)   FP2(-2 3)     2   13.83   0.0010   2669.050  7  2683.050
───────────────────────────────────────────────────────────────────────────────
idf = interaction degrees of freedom; tdf = total model degrees of freedom
```

A word of explanation about `idf` and `tdf` may be helpful. Both are df only for the interaction or main effects (or both) in question, ignoring the df for the adjustment model (if any). `tdf` is the total of the df for the function in each treatment group plus the df for the treatment effect. For example, in the above output, `tdf` for the FP2 × FP2 interaction is 7; this is made up of 1 for `therapy`, 4 for the four βs (two FP2 terms times two treatment groups) and 2 for the two FP2 powers. `idf` is the df for the interaction terms. In the example just discussed, the main effect has two fewer df than the interaction (two βs rather than four), so `idf` is 2. `tdf` is used when calculating the Akaike's information criterion (AIC), whereas `idf` is used in testing the deviance difference of the interaction.

The results regarding interactions are clear. Because all three p-values for `age` are not significant at the 5% level, there is no definite evidence of an interaction between `age` and `therapy`. We will not consider `age` any further. For `karno`, however, there is strong evidence of an interaction. We are faced with a choice of three models (linear, FP1, and FP2) for `karno` in its interaction with `therapy`. The AIC statistic presented in the final column offers a method of selecting among these models. The AIC is a type of likelihood that penalizes for model complexity. It is defined as the deviance (minus twice the maximized log likelihood for the interaction model) plus twice the df of the model. The model that minimizes the AIC can be chosen as the appropriate model. In this example, we select the linear interaction model because its AIC of 2675.695 is lower than that of the FP1 model (2682.981) and the FP2 model (2683.050).

To visualize the interaction between `karno` and `therapy`, we refit the linear interaction model with the `gendiff()` option, and we construct a treatment-effect plot from the saved `d1_1`, `d1lb_1`, and `d1ub_1` variables (see figure 2[1]):

```
. mfpi stcox, adjvars(gradd1 age surgd1) with(therapy) linear(karno) gendiff(d)
. mfpi_plot karno, ytitle("Log relative hazard")
```

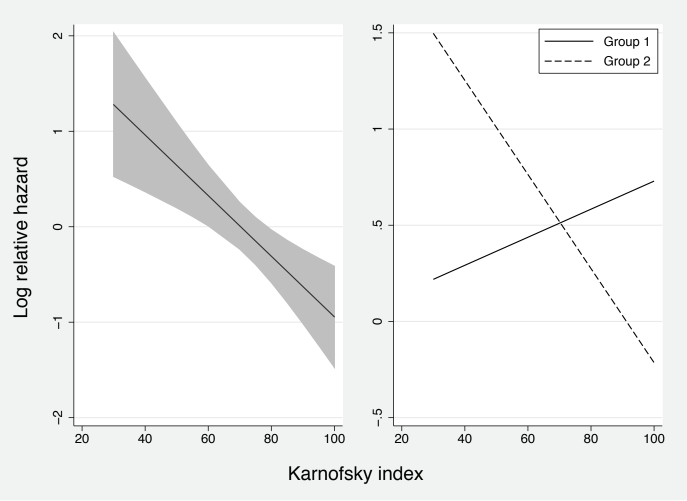

Figure 2. Glioma data. Left panel: treatment-effect plot for `therapy` by linear `karno` interaction; right panel: prognostic effect of `karno` in each `therapy` group. The model is adjusted for three other prognostic variables.

What we see is quite dramatic. For patients with low `karno`, the log-hazard ratio is about 1, indicating that the more aggressive experimental therapy (`therapy==1`) can actually kill people. Patients with a low Karnofsky index are very sick and may not

---

[1] The `mfpi_plot` command produces the left graph in figure 2. The command used to produce the graph on the right is not shown.

be able to tolerate toxic chemotherapy. For patients with high `karno`, the log-hazard ratio is about -0.8, indicating a substantial reduction in the hazard. This type of interaction, where there is a reversal of effect depending on the value of a covariate, is known as *qualitative* and is rarely seen in clinical research. Alternatively described, the prognostic effect of `karno` goes in opposite directions in the two treatment groups (right-hand graph).

As a crude check of a postulated interaction derived by complex FP modeling, Royston and Sauerbrei (2004) suggest dividing the continuous covariate into two to four groups and plotting Kaplan–Meier graphs of the treatment effect in each group. The resulting survival curves are not adjusted for other covariates, but if the interaction is robustly present in the data, the changing treatment effect should nevertheless be apparent in the resulting graphs. We create four approximately equal groups by categorizing `karno` at ≤ 60, 61 to 75, 76 to 85, and > 85 and produce the Kaplan–Meier survival estimates:

```
. generate byte kg = 1 + (karno>60) + (karno>75) + (karno>85)
. forvalues j = 1/4 {
.     sts graph if kg==`j´, by(therapy) xtitle("") ytitle("") title("Group `j´")
.     name(g`j´) legend(off) plot1opts(clp(l)) plot2opts(clp(-))
. }
. graph combine g1 g2 g3 g4, l1title("Survival probability")
> b2title("Survival time, yr")
```

Although the trend across `karno` categories is not perfect, the graphs (see figure 3) clearly show the reversal of the direction of the treatment effect in groups 1 and 4 that we expect from figure 2.

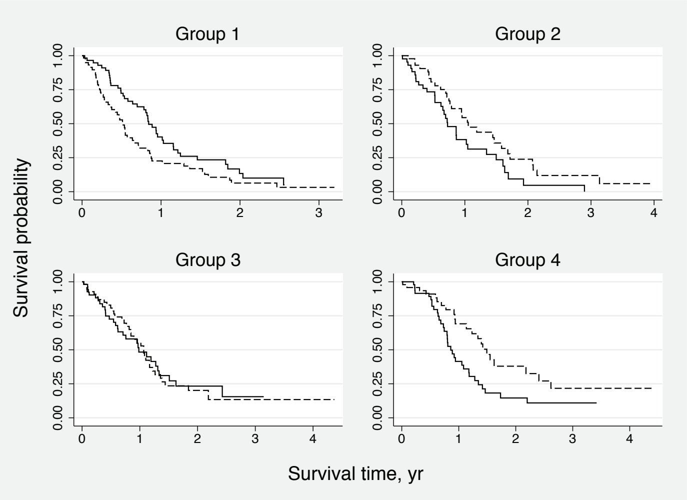

Figure 3. Glioma data. Kaplan–Meier plots of the treatment effect in four about equally sized groups according to `karno`. Solid lines, `therapy==0`; dashed lines, `therapy==1`.

The Kaplan–Meier graphs in figure 3 are unadjusted for confounders, whereas the MFPI model included adjustment for three important prognostic factors (`gradd1`, `age`, and `surgd1`). However, univariate analysis showed that the `karno` by `therapy` interaction was still highly significant.

In addition to the graphs, Royston and Sauerbrei (2004) suggest presenting unadjusted and adjusted estimates of the treatment effects in the covariate subgroups considered. In this example, adjustment does not make much difference.

7.2 STEPP analysis

The use of the STEPP technique has two main associated decisions:

- Tail or window variant?
- What parameter values?

We illustrate both variants. As we have pointed out elsewhere, the TO variant seems to produce much more interpretable graphs than the SW variant (Sauerbrei, Royston, and Zapien 2007), but for information and completeness we present both here.

The value of g for the TO variant should not be so small that one can hardly see any detail of the putative interaction, nor so large that the result is unstable. Values in the range $4 \leq g \leq 10$ appear quite suitable (giving between 7 and 19 points on the graph).

For the SW variant, n_2 is more critical than n_1 because it represents the subpopulation size, effectively the sample size for each fitted model. A reasonable range for n_2 may be 50 to 90. We took $n_2 = 50$ and n_1 to be 10 less than n_2.

To use the STEPP technique to investigate the interaction between the variables `karno` and `therapy`, we use the same confounder model (`gradd1`, `age`, `surgd1`, and `epi`) as derived by MFPI. For example, for the TO variant with $g = 6$,

```
. stepp_tail stcox karno gradd1 age surgd1 epi, gen(z) with(therapy) g(6)
. stepp_plot z, ytitle("Log relative hazard") xtitle("Karnofsky index")
```

Figure 4 shows the resulting plots for $g = 6$ and $g = 10$.

(*Continued on next page*)

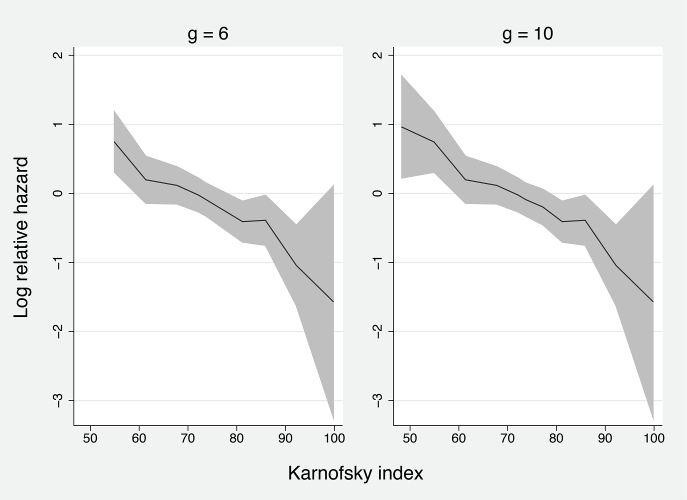

Figure 4. Glioma data. STEPPs (TO variant), adjusting for four prognostic factors.

The "message" from each plot is essentially the same: there is a linear interaction between `karno` and `therapy`. The cost of choosing a larger value of g is typically a slight increase in the uncertainty of the estimated treatment effect (reflected in wider pointwise confidence intervals). Because `karno` has only 13 distinct values (30 to 100 in steps of 5), increasing g in this example has much less of an effect than it would have with a truly continuous covariate (see, for example, Sauerbrei, Royston, and Zapien [2007]). MFPI always provides an estimate of the treatment effect over the complete range of values of a covariate.

Figure 5 illustrates the `karno` × `therapy` interaction using the SW variant of the STEPP with $n_1 = 40$ and $n_2 = 50$. Although the conclusion from the plot is essentially the same as from figures 3 and 4, the uncertainty band is wider and the result is more "noisy".

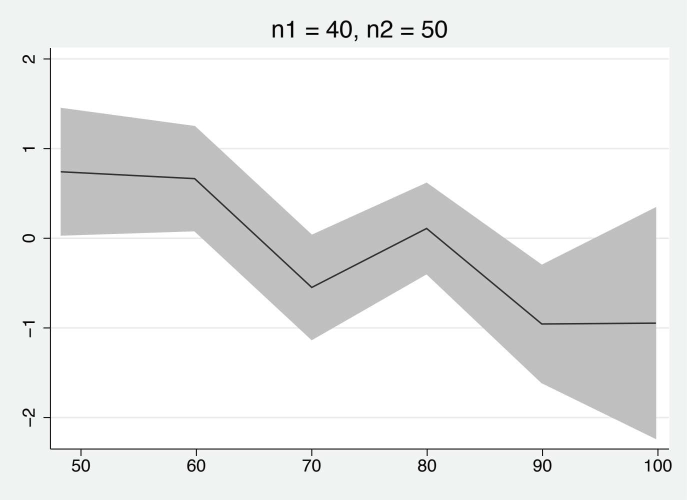

Figure 5. Glioma data. STEPP (SW variant), adjusting for three prognostic factors.

8 Example 2: Node-positive breast cancer

From July 1984 to December 1989, the German Breast Cancer Study Group recruited 720 patients with primary node-positive breast cancer into a "comprehensive cohort study" in which eligible patients are either randomized or treated according to one of the therapies under investigation (Schmoor, Olschewski, and Schumacher 1996). Hormonal treatment with tamoxifen (TAM) and the duration of CMF-chemotherapy (three versus six cycles) were evaluated in a 2 × 2 design. The recurrence-free survival time of 686 patients (299 events) with complete data for the standard prognostic factors age, menopausal status, tumor size, tumor grade, number of positive lymph nodes, and progesterone (PgR) and estrogen receptor (ER) status is analyzed.

With an effective sample size of 299, the study is too small for a sensitive investigation of interactions. Here is it used to illustrate some issues of the two approaches.

It is well established that ER status predicts response to hormonal adjuvant therapy with TAM. The risk of disease recurrence is reduced to a much greater extent by TAM in ER-positive patients than in ER-negative patients (Early Breast Cancer Trialists' Collaborative Group 1998). Here we will explore the TAM × ER interaction "naïvely", using the MFPI and STEPP methods.

8.1 MFPI

Applying `mfpi` with all covariates as potential confounding factors, selecting the confounder model at the 5% significance level, and looking for an interaction between `er` and `tam` gives the following results:

```
. use gbsg.dta, clear
(German breast cancer data)

. mfpi stcox age meno size gradd1 gradd2 nodes pgr, with(tam) linear(er)
> fp1(er) fp2(er) select(0.05) showmodel

Variables in adjustment model
─────────────────────────────

      age: power(s) = -2 -.5
     meno: not selected
     size: not selected
   gradd1: power(s) = 1
   gradd2: not selected
    nodes: power(s) = 1 2
      pgr: power(s) = .5

Interactions with tam (686 observations). Flex-1 model (least flexible)
─────────────────────────────────────────────────────────────────────────────
Var         Main        Interact     idf   Chi2     P     Deviance tdf   AIC
─────────────────────────────────────────────────────────────────────────────
er          Linear      Linear        1    0.02   0.8959  3419.810  3  3425.810
er          FP1(3)      FP1(3)        1    0.14   0.7094  3417.415  4  3425.415
er          FP2(-.5 3)  FP2(-.5 3)    2    3.97   0.1377  3412.954  7  3426.954
─────────────────────────────────────────────────────────────────────────────
idf = interaction degrees of freedom; tdf = total model degrees of freedom
```

The confounder model includes `gradd1` and FP transformations of `age`, `nodes`, and `pgr`. According to the reported p-values, there is no evidence of an interaction between `er` and `tam`. In "hypothesis generation mode" (i.e., where a dataset was being screened for the presence of interactions, with no predefined expectations), that would be the end of the matter. However, we have used only the least flexible of our variants to look for interactions. In the next step, we show that FLEX2 gives a different result.

With the following command, FLEX2, the first variant of MFPI, is invoked:

```
. mfpi stcox age meno size gradd1 gradd2 nodes pgr, flex(2) with(tam) linear(er)
> fp1(er) fp2(er) select(0.05)

Interactions with tam (686 observations). Flex-2 model (intermediate)
─────────────────────────────────────────────────────────────────────────────
Var         Main        Interact     idf   Chi2     P     Deviance tdf   AIC
─────────────────────────────────────────────────────────────────────────────
er          Linear      Linear        1    0.02   0.8959  3419.810  3  3425.810
er          FP1(-2)     FP1(-2)       2    5.99   0.0499  3414.315  4  3422.315
er          FP2(-2 3)   FP2(-2 3)     4    6.28   0.1793  3410.879  7  3424.879
─────────────────────────────────────────────────────────────────────────────
idf = interaction degrees of freedom; tdf = total model degrees of freedom
```

Changing the approach for investigating interactions has no impact on the adjustment model that is selected. However, the best FP1 and FP2 functions have different power terms than before. In particular, the FP1 power (-2) fits the data better than the power of 3 chosen with FLEX1 (deviances 3414.3 and 3417.4, respectively). The model with the lowest AIC is FP1. For that model, we find the interaction is just significant at the 5% level. The treatment-effect plot for the FP1 model is shown in figure 6, plotted on a log scale of `er` + 1 for legibility.

It is clear from figure 6 that the `er` × `tam` interaction is rather distinctive: patients with zero `er` do not respond to `tam`, patients with very low but still positive values may

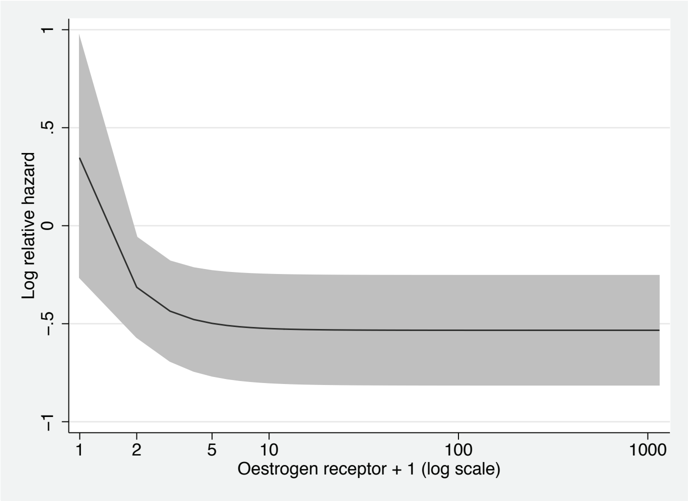

Figure 6. GBSG data. Treatment-effect plot for `er` × `tam` interaction, estimated by `mfpi` with `flex(2)` option. FP1 curves with power −2 were fit in each treatment group.

respond to a degree, and patients with higher values show a highly significant effect of `tam` (a hazard reduction of about 40%). Such an interpretation is entirely consistent with medical knowledge.

Why does the FLEX1 analysis fail so badly to detect the `er` × `tam` interaction? One reason is that when adjusted for other factors, `er` has little or no prognostic importance. `fracpoly` can be used to examine the prognostic effect of `er` in a model adjusted for the other variables and transformations chosen:

```
. fracpoly stcox er age -2 -0.5 gradd1 nodes 1 2 pgr 0.5 tam, compare
  (output omitted)
Fractional polynomial model comparisons:
---------------------------------------------------------------
er                df       Deviance   Dev. dif.  P (*)  Powers
---------------------------------------------------------------
Not in model       0       3420.726     3.807    0.433
Linear             1       3419.827     2.907    0.406  1
m = 1              2       3417.554     0.634    0.728  3
m = 2              4       3416.920       —         —   -.5 3
---------------------------------------------------------------
(*) P-value from deviance difference comparing reported model with m = 2 model
```

The *p*-value for testing an FP2 function for `er` against exclusion of `er` is 0.43, so an FP2 main effect of `er` would not be selected by a stepwise algorithm operating at a conventional significance level. More importantly, the FLEX1 (default) version of MFPI determines the required power transformation for `er`, considering only the main effect of `tam` and ignoring the possible interaction between `er` and `tam`. Because the main effect of `er` is uninfluential, the best-fitting power (3) for FP1 models is poorly estimated and depends on chance. By contrast, FLEX2 takes the interaction into account when finding the best-fitting power, which turns out to be quite different (−2) and gives a dissimilar functional form.

The p-values for the FP1 interaction between `er` and `tam` according to FLEX3 and FLEX4 are 0.07 and 0.14, respectively. FLEX3 reestimates the power for the main effect. Overall, this improves the model fit, but the interaction effect is slightly reduced. Similarly, the interaction effect in FLEX4 is reduced by allowing more flexible and therefore better-fitting main effects.

As stated earlier, the sample size is insufficient for MFPI to have sufficient power to detect the `er` × `tam` interaction.

8.2 STEPP

Using the adjustment model selected with MFP, figure 7 shows some examples of STEPPs of the `er` × `tam` interaction.

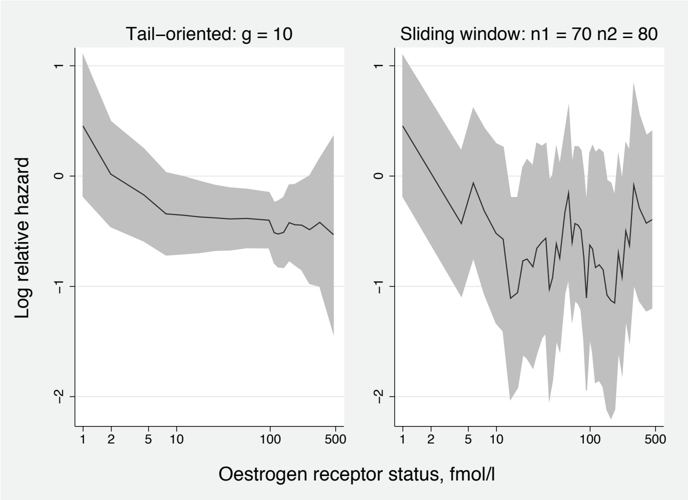

Figure 7. GBSG data. TO and SW STEPPs of the `er` × `tam` interaction, adjusted for other prognostic variables.

The dramatically better performance of the TO variant is apparent here. The plot demonstrates some instability of the treatment-effect function for the TO variant, mainly caused by selecting a large value for g ($g = 10$ gives 19 subgroups). Although the two graphs tell basically the same story as each other (and as figure 6), the SW variant is much noisier and consequently much harder to interpret. Lacking the result from TO or from MFPI, many researchers, being shown only the SW plot, would doubt the existence of an interaction. The main issue with the TO variant is the possibility of bias in the estimated treatment effect for very low values of `er`, which the FP1 analysis indicates is falling very rapidly as `er` increases. The averaging across subpopulations may slightly reduce the rate of change in this region of values, leading to bias.

Finally, Kaplan–Meier plots of the `tam` effect according to suitable ranges of `er` confirm the nature of the interaction (see figure 7.5 of Royston and Sauerbrei [2008]).

9 Discussion

We have presented a comprehensive implementation of the MFPI algorithm as described by Royston and Sauerbrei (2004) and have included some hitherto unpublished variants of the algorithm in the software. In our experience so far, the basic FLEX1 variant usually works well. It may fail in the situation we illustrate in the GBSG study, in which a variable with no main effect interacts with a treatment variable. Here the main effect of `er` was negligible because we include, in the adjustment model, the highly correlated variable `pgr`. Eliminating `pgr` from the list of candidates for the adjustment model, FLEX1 selects power -0.5 for `er`. The treatment-effect function is similar to figure 6. This situation may actually be quite common in clinical trials but is rarely looked for (perhaps to avoid a charge of "data-dredging"). We encourage users to explore at least the FLEX1 and FLEX2 variants with their data. FLEX3 and FLEX4 are provided because they seem to us to be natural alternatives; we hope that also making them available will encourage others to experiment with their own data, to build up further experience of the best approaches. Simulation studies are required to better understand the properties, advantages, and disadvantages of the four variants. Although the type I error for the interaction test in the FLEX1 variant of MFPI appears to be approximately correct (Sauerbrei, Royston, and Zapien 2007), that of the other three variants remains to be studied.

STEPP, as described in the original articles (Bonetti and Gelber 2000; Bonetti and Gelber 2004), includes statistical significance tests of the interaction. We decided not to implement these in our Stata routines. The main reason is that in some sense, the tests are not well defined because the results must to an extent depend on the parameters (g, or n_1 and n_2) that govern the STEPP estimates. As with categorization of a continuous covariate prior to regression modeling (a practice frowned on by some statisticians, including ourselves), the placing of cutpoints and ensuing interpretation of results is a process fraught with danger. We think that the main benefit of the STEPP technique is as an exploratory or confirmatory tool (confirmatory in the sense of providing independent backup for results determined using MFPI). MFPI can easily be used to search for possible interactions, e.g., when a longer list of potential predictive markers is available in a large randomized trial (Filipits et al. 2007). The TO variant of the STEPP technique can be used as a check for possible interactions identified with MFPI. Confirmation of the MFPI result with the STEPP technique verifies only that the significant interaction is not caused by mismodeling the data. Validation in independent data is still required.

An obvious issue with interaction research is that of multiplicity. Royston and Sauerbrei (2004) distinguished between the cases of a predetermined hypothesis (as with `er` and `tam` in primary breast cancer) and a "hypothesis-searching" situation in which interactions are trawled for (as in the glioma example). Because often in the latter situation many interactions are considered, it may be sensible to use a more stringent p-value for testing interactions, for example, 0.01 rather than the conventional 0.05 level or rather than the use of AIC as a model selection criterion. The problem with using a formal correction for multiplicity (e.g., Bonferroni or Bonferroni-Holm), as some have advocated, is loss of power in a situation in which power is already likely to be low. It seems that type II errors, meaning that real interactions are overlooked, need greater consideration. Clearly, interactions discovered in hypothesis-generation mode need to be validated in independent data.

A further extension of MFPI, called MFPIgen, is described in our book (Royston and Sauerbrei 2008). It models continuous by continuous interactions with FP methodology. A separate Stata routine implementing MFPIgen is being prepared and will be reported in a later article.

10 References

Bonetti, M., and R. D. Gelber. 2000. A graphical method to assess treatment–covariate interactions using the Cox model on subsets of the data. *Statistics in Medicine* 19: 2595–2609.

———. 2004. Patterns of treatment effects in subsets of patients in clinical trials. *Biostatistics* 5: 465–481.

Early Breast Cancer Trialists' Collaborative Group. 1998. Tamoxifen for early breast cancer: An overview of the randomised trials. *Lancet* 351: 1451–1467.

Filipits, M., R. Pirker, A. Dunant, S. Lantuejoul, K. Schmid, A. Huynh, V. Haddad, F. André, R. Stahel, J.-P. Pignon, J.-C. Soria, H. H. Popper, T. L. Chevalier, and E. Brambilla. 2007. Cell cycle regulators and outcome of adjuvant cisplatin-based chemotherapy in completely resected non–small-cell lung cancer: The international adjuvant lung cancer trial biologic program. *Journal of Clinical Oncology* 25: 2735–2740.

Royston, P., and D. G. Altman. 1994. Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. *Applied Statistics* 43: 429–467.

Royston, P., and W. Sauerbrei. 2004. A new measure of prognostic separation in survival data. *Statistics in Medicine* 23: 723–748.

———. 2008. *Multivariable Model-building: A Pragmatic Approach to Regression Analysis Based on Fractional Polynomials for Modelling Continuous Variables.* Chichester, UK: Wiley.

Sauerbrei, W., and P. Royston. 1999. Building multivariable prognostic and diagnostic models: Transformation of the predictors using fractional polynomials. *Journal of the Royal Statistical Society, Series A* 162: 71–94.

Sauerbrei, W., P. Royston, and K. Zapien. 2007. Detecting an interaction between treatment and a continuous covariate: A comparison of two approaches. *Computational Statistics and Data Analysis* 51: 4054–4063.

Sauerbrei, W., and M. Schumacher. 1992. A bootstrap resampling procedure for model building: Application to the Cox regression model. *Statistics in Medicine* 11: 2093–2109.

Schmoor, C., M. Olschewski, and M. Schumacher. 1996. Randomized and non-randomized patients in clinical trials: Experiences with comprehensive cohort studies. *Statistics in Medicine* 15: 263–271.

Ulm, K., C. Schmoor, W. Sauerbrei, G. Kemmler, Ü. Aydemir, B. Müller, and M. Schumacher. 1989. Strategien zur Auswertung einer Therapiestudie mit der Überlebenszeit als Zielkriterium. *Biometrie und Informatik in Medizin und Biologie* 20: 171–205.

About the authors

Patrick Royston is a medical statistician with more than 30 years of experience. He has a strong interest in biostatistical methodology and in statistical computing and algorithms. At present, he works in clinical trials and related research issues in cancer. Currently, he is focusing on problems of model building and validation with survival data, including prognostic factors studies, on parametric modeling of survival data, on multiple imputation of missing values, and on novel trial designs.

Willi Sauerbrei has worked for more than two decades as an academic biostatistician. He has extensive experience of randomized trials in cancer, with a particular concern for breast cancer. Having a long-standing interest in modeling prognosis and a PhD thesis in issues in model building, he has more recently concentrated on model uncertainty, meta-analysis, treatment–covariate interactions, and time-varying effects in survival analysis.