AgEcon SEARCH
RESEARCH IN AGRICULTURAL & APPLIED ECONOMICS

The World's Largest Open Access Agricultural & Applied Economics Digital Library

This document is discoverable and free to researchers across the globe due to the work of AgEcon Search.

Help ensure our sustainability.

Give to AgEcon Search

AgEcon Search
http://ageconsearch.umn.edu
aesearch@umn.edu

*Papers downloaded from **AgEcon Search** may be used for non-commercial purposes and personal study only. No other use, including posting to another Internet site, is permitted without permission from the copyright owner (not AgEcon Search), or as allowed under the provisions of Fair Use, U.S. Copyright Act, Title 17 U.S.C.*

No endorsement of AgEcon Search or its fundraising activities by the author(s) of the following work or their employer(s) is intended or implied.

heck for updates

The Stata Journal (2019)
19, Number 3, pp. 566–580 DOI: 10.1177/1536867X19874226

Visualizing effect modification on contrasts

Niels Henrik Bruun
Department Of Public Health
Aarhus University
Aarhus, Denmark
nhbr@ph.au.dk

Abstract. A recurring problem in statistics is estimating and visualizing nonlinear dependency between an effect and an effect modifier. One approach to handle this is polynomial regressions of some order. However, polynomials are known for fitting well only in limited ranges. In this article, I present a simple approach for estimating the effect as a contrast at selected values of the effect modifier. I implement this approach using the flexible restricted cubic splines for the point estimation in a new simple command, `emc`. I compare the approach with other classical approaches addressing the problem.

Keywords: st0567, emc, effect modification

1 Introduction

A classical approach on effect modifiers is to regress the outcome (all-cause mortality, `all10`, No/Yes) on the exposure (`smoker`, Yes/No), the effect modifier (mean arterial pressure, `map`, in mm Hg), and the interaction term between the exposure and the effect modifier. This makes the measure of association (or its log) linear in the effect modifier. Sometimes, the effect modifier is stratified into intervals before modeling to detect any nonlinear dependency of the measure of association on the effect modifier (mean arterial pressure). Examples of stratifying are age divided into age spans and body mass index divided into groups like "underweight", "normal weight", and "overweight". When one stratifies, the effect modifier is estimated by a regression coefficient for each strata, and the value for the effect modifier in each strata has to be approximated, for example, by the midpoint of strata interval. Using polynomial regressions, one can model nonlinear dependencies between the effect modifier and the contrast using margins and marginal effects. In Stata, `margins` (see [R] **margins** and Mitchell [2012]) is a powerful postregression command to estimate margins and marginal effects. However, polynomial regressions offer a limited set of functional shapes and can give a poor fit to the data.

In this article, I present an approach where the measure of association between smoking and mortality is split into two log odds-functions dependent on the mean arterial pressure—one for nonsmokers and one for smokers. Contrasting these two log odds-functions gives an estimate of the log odds-ratio at selected values of the effect modifier (mean arterial pressure). Exponentiated, the log odds-ratios give the estimates of the odds ratios. Restricted cubic splines are a simple, flexible way of estimating the possibly nonlinear functions for the unexposed (nonsmokers) and the exposed (smokers)

© 2019 StataCorp LLC

groups. I introduce a simple command, `emc`, using restricted cubic splines for getting flexible estimates of the nonlinear dependence.

This article proceeds as follows. Section 2 derives and explains the contrast function in the linear and general cases. Section 3 introduces restricted cubic splines. Section 4 presents the example dataset. Section 5 describes commands in Stata for restricted cubic splines. Section 6 presents a new prefix command, `emc`, based on restricted cubic splines. Section 7 demonstrates usage and comparisons with the classical approaches when the measure of association is linear in the effect modifier and when the effect modifier is stratified. Section 7 also demonstrates the use of `margins`. Finally, section 8 gives some concluding remarks.

2 The contrast function

The expected all-cause mortality (`all10`), $[E(\texttt{all10}|\texttt{smoker},\texttt{map})]$, of both smoking (`smoker`) and mean artery pressure (`map`) can be modeled as

$$E(\texttt{all10}|\texttt{smoker}=s,\texttt{map}=m)=\begin{cases} f_0(m) & s=0,\text{Nonsmoker} \\ \beta_{\texttt{smoker}}+f_1(m) & s=1,\text{Smoker} \end{cases}$$

In a logit regression, the expected all-cause mortality is the log odds-function of the mean artery pressure handled independently for nonsmokers and smokers. The contrast function, estimating the log odds-ratio of the effect modifier (`map`) based on the above, becomes

$$\begin{aligned} E(\texttt{all10}|\texttt{smoker}=1,\texttt{map}=m)-E(\texttt{all10}|\texttt{smoker}=0,\texttt{map}=m)=\beta_{\texttt{smoker}} \\ +f_1(m)-f_0(m) \end{aligned}$$

One simple, flexible way to approximate nonlinear functions like $f_0(m)$ and $f_1(m)$ from data points is to use restricted cubic splines.

3 Restricted cubic splines

Cubic splines are functional approximations based on piecewise cubic polynomials (see Harrell [2015], Buis [2009], and Croxford [2016], Orsini and Greenland [2011]).

When a function is possibly nonlinear and unknown except for some data points, cubic splines are a flexible way to approximate the unknown function. Cubic splines are a set of transformations on the independent variable, which together linearly approximate the unknown function. This makes cubic splines useful for nonlinear regression modeling.

A single polynomial

$$Y = b_0 + b_1 \times X + b_2 \times X^2 + b_3 \times X^3$$

will usually fit observed pairs of X and Y values poorly at some parts of the curve. In splines, X values $(k_i)_{i=1}^{n}$-named knots are chosen. The combined polynomial spline is the overall cubic polynomial added cubic effects starting at each knot $b_{i+3} \times \max(0, X - k_i)^3$. By design, the cubic spline has continuous second-order derivatives. Also by design, it is easy to generate variables dependent only on X and the knots, $(k_i)_{i=1}^{n}$, such that the coefficients, $(b_i)_{i=0}^{n+3}$, can be estimated using a regression. The general formula for a cubic spline describing Y as a function of X is

$$Y = b_0 + b_1 \times X + b_2 \times X^2 + b_3 \times X^3 + \sum_{i=1}^{n} b_{i+3} \times \max(0, X - k_i)^3$$

There is typically a problem with cubic splines at the tails of the curve. It is unknown how the curve should behave when extrapolating the curve beyond the range of X.

To avoid inheriting false curvature from the center of the curve, one can restrict the splines to be linear beyond the range of the knots, hence restricted cubic splines.

To achieve linearity at the left side of the range of X for a restricted cubic spline, one removes the quadratic $(b_2 \times X^2)$ and cubic parts $(b_3 \times X^3)$ outside the summation. A more complex transformation of X is necessary to secure linearity to the right of the range of the knots (see [R] **mkspline**). One advantage of this transformation is that the first generated variable is the generating variable (X) itself.

There are guidelines (Harrell 2015, 26) covering most datasets for choosing the knots and their numbers. In summary, they are the following:

- use fixed percentiles for the marginal distribution of the effect modifier to ensure enough points in each interval and guarding against outliers overly influencing knot placement;
- choose a number of knots between 3 (small datasets, number of observations around 100) and 7;
- if the number of observations is less than 100, choose the fifth-smallest and the fifth-largest observation as outer knots;
- choosing more than five knots is seldom necessary;
- often, choosing four knots is a good choice; and
- Akaike's information criterion (AIC) can be used for data-driven comparisons of the number of knots and where to place them.

4 The example data

The example dataset is the Whitehall 1 dataset, which consists of data on 18,403 male British Civil Servants employed in London (StataCorp 2017; Royston and Sauerbrei 2008):

```
. use all10 cigs map using http://www.stata-press.com/data/r15/smoking.dta
(Smoking and mortality data)
```

The outcome is 10 years all-cause mortality (variable `all10`) with status alive (`0`) or dead (`1`).

The exposure variable `smoker` is a recoding of `cigs` (number of cigarettes) such that `smoker` is zero (`No`) if `cigs` is zero and one (`Yes`) otherwise. The commands below create the exposure variable `smoker`.

```
. generate smoker = (cigs > 0) if !missing(cigs)
. label variable smoker "Smoking at baseline"
. label define smoker 0 "No" 1 "Yes"
. label values smoker smoker
```

The effect modifier is `map`, mean arterial pressure (mm Hg), with range $[50; 210]$.

5 Restricted cubic splines in Stata

In Stata, to generate a set of restricted cubic spline variables for regressions, use the command `mkspline` with option `cubic`. The option `nknots()` specifies the number of knots to use. The knots can be chosen according to Harrell (2015).

The package `postrcspline` (Buis 2008, 2009) extends `mkspline` with `mkspline2` and further introduces commands `adjustrcspline` and `mfxrcspline` to get adjusted predictions and marginal effects, respectively.

Orsini and Greenland (2011) extend the package `postrcspline` into a single command, `xblc`, that is not limited to restricted cubic splines but also handles, for example, fractional polynomials with the `fp` command.

The goal for both Buis (2008) and Orsini and Greenland (2011) is to model the dependency between the outcome and a continuous exposure. The first uses restricted cubic splines, while the latter is more flexible.

My approach in this article is to estimate a measure of association (a contrast or a function of one) as a function of a selected set of values of an effect modifier by estimating each part of the contrast as a function separately and then estimating the contrast by the difference between the two functions for each of the selected values.

Using four knots, we can estimate the contrast function (from section 2) of the coefficients by the following four steps for the example dataset:

1. Estimate restricted cubic splines for nonsmokers. Spline values are missing for smokers:

```
. mkspline _map0 = map if !smoker & !missing(smoker, map), cubic nknots(4)
```

2. Estimate restricted cubic splines for smokers. Spline values are missing for nonsmokers:

```
. mkspline _map1 = map if smoker & !missing(smoker, map), cubic nknots(4)
```

3. Set missing values for the splines to be zero. This way, the estimates on splines for nonsmokers are independent of the estimates on splines for smokers:

```
. mvencode _map??, mv(0) override
```
(*output omitted*)

4. Do the logit regression:

```
. logit all10 i.smoker _map0? _map1?, nolog
Logistic regression                             Number of obs     =     17,260
                                                LR chi2(7)        =     574.95
                                                Prob > chi2       =     0.0000
Log likelihood = -5199.3929                     Pseudo R2         =     0.0524
```

all10	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
smoker						
Yes	-.1696929	1.479339	-0.11	0.909	-3.069144	2.729758
_map01	-.0207781	.0124838	-1.66	0.096	-.045246	.0036897
_map02	.1518839	.0476537	3.19	0.001	.0584843	.2452834
_map03	-.3791315	.1389264	-2.73	0.006	-.6514223	-.1068407
_map11	-.0093646	.0114195	-0.82	0.412	-.0317464	.0130172
_map12	.1138028	.0438132	2.60	0.009	.0279305	.1996751
_map13	-.2997912	.1291569	-2.32	0.020	-.552934	-.0466484
_cons	-1.264849	1.103351	-1.15	0.252	-3.427378	.8976792

The variables `_map0?` and `_map1?` are the generated restricted cubic spline variables for nonsmokers and smokers. The first variable for nonsmokers, `_map01`, is equal to the values of `map` for nonsmokers and zero for smokers. Estimates of the `_cons` and `_map0?` are for nonsmokers, and `_cons`, `smoker`, and `_map1?` are for smokers. When contrasting nonsmokers with smokers, `_cons` levels out. Possible linear adjustments in the estimating regression also level out.

To predict the contrasts at selected values of the effect modifier is more complex.

The independent variables from restricted cubic splines are transformations of the effect modifier, and these transformations cannot be handled by, for example, `margins`.

To use `margins` to estimate the contrasts, we must make the underlying model an overall polynomial, for example, a linear or quadratic approximation. `emc` implements the presented approach to estimate and visualize nonlinear dependence between an effect measure or contrast and an effect modifier.

6 The emc command

6.1 Syntax

`emc, at(`*numlist*`)` [`pctknots(`*numlist*`) nknots(`#`) eform keepcubicsplines emcnames(`*namelist*`) cilimits(`*real*`) graph` *twoway_options*]:
regression_command

6.2 Options

`at(`*numlist*`)` specifies the values of the effect modifier at which to estimate the effect measure. `at()` is required.

`pctknots(`*numlist*`)` specifies the percentages of the percentiles that are used for calculating the restricted cubic splines. *numlist* is a list of values between 0 and 100. The length of the *numlist* must be between 3 and 10.

`nknots(`#`)` specifies the recommended standard set of percentages as described in `nknots()` of [R] **mkspline**; see *Methods and formulas* in [R] **mkspline**. # must be an integer value between 3 and 7. The default is `nknots(4)`. If option `pctknots()` is set, option `nknots()` is ignored.

`eform` exponentiates the estimated effect measures.

`keepcubicsplines` keeps the generated cubic spline regressors for detailed analysis.

`emcnames(`*namelist*`)` renames the generated variables with the requested values of the effect modifier, the estimated effect measures, and the 95% confidence interval for the estimated effect measure. *namelist* must have length 4. The default names are *__third_variable_name*, *__third_variable_name*`_contrast`, *__third_variable_name*`_lb`, and *__third_variable_name*`_ub`.

`cilimits(`*real*`)` changes the percentage for the confidence intervals from the default `cilimits(95)`. *real* must be between 0 and 100.

`graph` generates a default graph.

twoway_options change the default graph. `graph` is implied if *twoway_options* are specified.

6.3 Description

The prefix command `emc` takes a regression command as an argument. From the regression command argument, `emc` uses the first variable as an outcome variable, the second variable as a binary contrast variable, and the third variable as an effect modifier transformed into a function using cubic splines.

For each value in *numlist* specified in the `at()` option, the estimated contrast and the confidence interval limits (normal approximation) are saved in four variables. For reproducibility, detailed results are stored in the returned values.

`emc` with the `graph` option generates a simple default graph of the estimates and the 95% confidence interval. It is deliberately simple but easily modifiable by adding any `twoway` option. The option `graph` is not necessary when a `twoway` option is specified.

The calculations in `emc` are based on returned matrices `e(b)` and `e(V)` and are therefore independent of the type of regression performed. Interesting contrasts (directly or exponentiated) that may be studied with this approach include the following:

- difference in means using `regress` or `mixed`
- odds ratios using `logit` or `logistic`
- odds ratios in a matched study using `clogit`
- risk differences using `binreg`
- relative risks using `binreg`
- hazard ratios using `stcox` (here the first variable is the contrast, and the second, the effect modifier)
- incidence-rate ratios using `poisson` or `nbreg`

One can also analyze contrasts in `glm`.

I developed `emc` to estimate and visualize effect modification on a contrast or its log. However, one can also use `emc` to visualize gap developments by a continuous variable, for example, visualizing the income gap over time between the two genders.

`emc` was tested in Stata/IC for Windows, versions 12 through 15.

6.4 Stored results

`emc` stores the following in `r()`:

Macros

`r(graph_cmd)`	`twoway` graph command generating the graph
`r(emnames)`	names of the variables generated
`r(command)`	regression command generating the estimates

Matrices

`r(predictions)`	estimated contrasts and confidence intervals
`r(regressors)`	regressors used for the predictions
`r(knots)`	knots used for the unexposed and exposed parts of the effect modifier

7 Comparing emc prefix command with other approaches

This section compares an application of the `emc` prefix command with three other approaches by looking at how the odds ratio (effect measure) of smoking (`smoker`) on all-cause mortality (`all10`) is modified by mean arterial pressure (`map`). Figure 1 shows how the contrast (the log odds-ratio of smoking on all-cause mortality) is modified by the mean arterial pressure for four approaches. By exponentiation, figure 2 shows how the effect measure, or the odds ratio of smoking on all-cause mortality, is modified by the mean arterial pressure for four approaches. The results show the presence of a nonlinear dependence between the odds ratios of smoking on all-cause mortality and mean arterial pressure. Smoking affects all-cause mortality the most when mean arterial pressure is around 100.

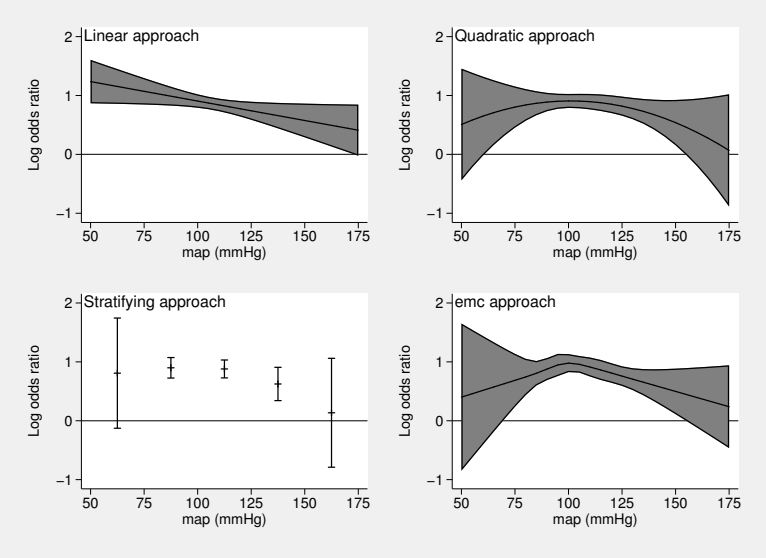

Figure 1. Four approaches (linear, quadratic, stratified, and `emc`) showing how the mean arterial pressure in the range [50; 175] modifies the log odds-ratios of smoking on all-cause mortality. A horizontal line at zero for no effect is shown.

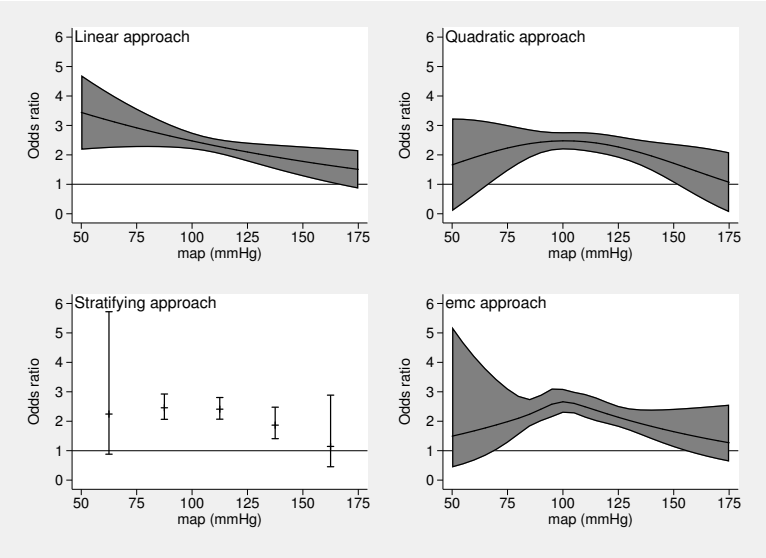

Figure 2. Four approaches (linear, quadratic, stratified, and `emc`) showing how the mean arterial pressure in the range [50; 175] modifies the odds ratios of smoking on all-cause mortality. A horizontal line at one for no effect is shown.

The first approach reported in figures 1 and 2 is linear. Here the log odds are modeled in a logit model with the main effects of `smoker` and `map` and an interaction term, `smoker#map`.

```
. estimates clear
. logit all10 i.smoker##c.map, nolog
Logistic regression                             Number of obs     =     17,260
                                                LR chi2(3)        =     545.66
                                                Prob > chi2       =     0.0000
Log likelihood = -5214.0391                     Pseudo R2         =     0.0497

------------------------------------------------------------------------------
       all10 |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      smoker |
        Yes |   1.566194   .3427253     4.57   0.000     .8944645    2.237923
        map |   .0323882   .0022805    14.20   0.000     .0279185    .0368579
            |
smoker#c.map |
        Yes |  -.0066035   .0031632    -2.09   0.037    -.0128033   -.0004036
            |
      _cons |  -6.046836   .2528069   -23.92   0.000    -6.542328   -5.551343
------------------------------------------------------------------------------
. estimates store Linear
```

First, we estimate the log odds-ratios using `margins`:

```
. margins, at(map=(50(5)175)) expression(_b[1.smoker]+_b[1.smoker#map]*map)
```
(*output omitted*)

We save estimates and confidence intervals in matrices for graphing:

```
. matrix tmp = r(at), r(table)´
. matrix Linear = tmp[1...,"map"], tmp[1...,"b"], tmp[1...,"ll"], tmp[1...,"ul"]
```
(*output omitted*)

Second, we estimate the odds ratios

```
. margins, at(map=(50(5)175)) expression(exp(_b[1.smoker]+_b[1.smoker#map]*map))
```
(*output omitted*)

We save estimates and confidence intervals in matrices for graphing:

```
. matrix tmp = r(at), r(table)´
. matrix Linear_exp = tmp[1...,"map"], tmp[1...,"b"], tmp[1...,"ll"],
> tmp[1...,"ul"]
```

The second model is quadratic; that is, there is a quadratic term (`map#map`) and an interaction term between `smoker`, and the quadratic term (`smoker#map#map`) is added to the linear model.

```
. logit all10 i.smoker##(c.map c.map#c.map), nolog
Logistic regression                             Number of obs     =     17,260
                                                LR chi2(5)        =     559.04
                                                Prob > chi2       =     0.0000
Log likelihood = -5207.3503                     Pseudo R2         =     0.0509

------------------------------------------------------------------------------
       all10 |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      smoker |
        Yes |  -.6605595   1.562096    -0.42   0.672    -3.722212    2.401093
        map |   -.032226   .0191726    -1.68   0.093    -.0698036    .0053516
            |
  c.map#c.map |   .0002759   .0000818     3.37   0.001     .0001157    .0004361
            |
smoker#c.map |
        Yes |   .0310415   .0278588     1.11   0.265    -.0235607    .0856437
            |
smoker#c.map#|
      c.map |
        Yes |  -.0001536   .0001225    -1.25   0.210    -.0003937    .0000864
            |
      _cons |  -2.368635   1.104982    -2.14   0.032    -4.534361   -.2029094
------------------------------------------------------------------------------
. estimates store Quadratic
```

Similarly to the linear case, we estimate the log odds-ratios by using `margins` and save them in matrices:

```
. margins, at(map=(50(5)175))
> expression(_b[1.smoker]+_b[1.smoker#map]*map+_b[1.smoker#map#map]*map*map)
  (output omitted)
. matrix tmp = r(at), r(table)´
. matrix Quadratic = tmp[1...,"map"], tmp[1...,"b"], tmp[1...,"ll"],
> tmp[1...,"ul"]
```

Likewise, we estimate the odds ratios by using `margins` and save them:

```
. margins, at(map=(50(5)175))
> expression(exp(_b[1.smoker]+_b[1.smoker#map]*map+_b[1.smoker#map#map]*map*map))
  (output omitted)
. matrix tmp = r(at), r(table)´
. matrix Quadratic_exp = tmp[1...,"map"], tmp[1...,"b"], tmp[1...,"ll"],
> tmp[1...,"ul"]
```

The third approach is stratifying the mean arterial pressure (`map`) into intervals of widths of 25.

We create the stratifying variable (`map_grp`) with the `egen` command `cut()`:

```
. egen map_grp = cut(map), at(50(25)175)
(9 missing values generated)
```

We create the interval midpoints and saved them in a matrix named `map` for later:

```
. mata: st_matrix("map", (2::6) :* 25 :+ 12.5)
. matrix colnames map = map
```

The main effects of `smoker` and the stratifying variable `map_grp` and their interaction terms are used in this model.

```
. logit all10 i.smoker##i.map_grp, nolog
Logistic regression                             Number of obs     =     17,251
                                                LR chi2(9)        =     513.10
                                                Prob > chi2       =     0.0000
Log likelihood = -5211.5158                     Pseudo R2         =     0.0469

------------------------------------------------------------------------------
       all10 |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      smoker |
        Yes |   .8082312   .4774587     1.69   0.090    -.1275706    1.744033
             |
      map_grp |
         75 |  -.2895231   .3960767    -0.73   0.465    -1.065819    .4867729
        100 |   .1882785   .3939385     0.48   0.633    -.5838267    .9603838
        125 |   1.255965   .4021398     3.12   0.002     .4677854    2.044144
        150 |   1.647616   .4708918     3.50   0.000      .724685    2.570547
             |
smoker#map_grp |
    Yes# 75 |   .0908806   .4856401     0.19   0.852    -.8609565    1.042718
    Yes#100 |   .0713478   .4837316     0.15   0.883    -.8767488    1.019444
    Yes#125 |  -.1837667   .4987726    -0.37   0.713    -1.161343    .7938096
    Yes#150 |  -.6730564   .6712805    -1.00   0.316    -1.988742    .6426292
             |
       _cons |  -2.76362   .3897001    -7.09   0.000    -3.527418   -1.999822
------------------------------------------------------------------------------
. estimates store Stratifying
```

We again use `margins` to estimate the log odds-ratios:

```
. margins r.smoker, over(map_grp) predict(xb)
  (output omitted)
. matrix tmp = r(table)´
. matrix tmp = tmp[1...,"b"], tmp[1...,"ll"], tmp[1...,"ul"]
```

We save interval midpoints and estimates for graphing:

```
. matrix Stratifying = map, tmp
```

A simple way of exponentiation of a matrix in Stata is to use Mata:

```
. mata: st_matrix("tmp", exp(st_matrix("tmp")))
. matrix colnames tmp = b ll ul
```

We save the estimated odds ratios for graphing:

```
. matrix Stratifying_exp = map, tmp
```

Finally, we use the `emc` command to generate the necessary estimates. We use the option `eform` to transform the log odds-ratio estimates from the underlying model into odds ratio estimates. `emc` generates variables for the effect modifier, the exponentiated contrast, and the confidence interval thereof by default.

```
. emc, at(50(5)175) eform: logit all10 smoker map, or
  (output omitted)
. estimates store emc
```

Table 1 compares the underlying logit models (that is, the models for log odds-ratios) using the AIC and Bayesian information criterion (BIC) from stored estimates for the four approaches and the command `estimates stats`. How to use AIC and BIC is discussed in Dziak et al. (2012) and summarized in PennState (2012). AIC and BIC both reward goodness of fit but penalize overfit models; that is, models with fewer parameters are preferred. In short, with AIC there is a risk of choosing an overfit model regardless of n. If n is large, there is little risk of choosing an overfit model based on BIC. BIC has a larger risk than AIC—regardless of n—of choosing an underfit model.

Table 1. AIC and BIC for the four models used for figure 1 and 2

	N	ll(Null)	ll(model)	Degrees of freedom	AIC	BIC
Linear	17260	−5486.87	−5214.04	4	10436	10467
Quadratic	17260	−5486.87	−5207.35	6	10427	10473
Stratifying	17251	−5468.07	−5211.52	10	10443	10521
`emc`	17260	−5486.87	−5199.39	8	10415	10477

For AIC, the `emc` approach is best, but the model in this approach is possibly overfit. For BIC, the linear approach is best, but the underlying model is possibly underfit. The quadratic approach is second best regardless of whether AIC or BIC is used.

For both AIC and BIC, the stratifying approach is the worst. As noted in Royston and Sauerbrei (2008, 2), stratification or classification leads to problems with overparameterization, loss of efficiency, and defining cutpoints. Further, "... a cutpoint model is an unrealistic way to describe a smooth relationship between a predictor and an outcome variable."

Looking at figure 1, we see that the linear approach, compared with the other three approaches, is too simple. The linear model in figure 1 gives poor prediction when the mean arterial pressure is low and hence lesser or wrong insight.

Of course, one can model the curve by adding quadratic or higher terms and using, for example, `margins` to estimate the contrasts, "but the range of curve shapes afforded by conventional low-order polynomials is limited" (Royston and Sauerbrei 2008, 2). Adding a quadratic term leads essentially to the same problem one level up, as with the linear approach—somewhere on the curve, there will be a poor fit.

The emc approach handles the nonlinearity of the contrast in a simple, flexible way by including shapes more complex than low-order polynomials and without the problems of defining the proper polynomial order.

8 Concluding remarks

I presented a simple approach to model and visualize the dependency of a contrast to a continuous variable based on a regression. However, one can also use this approach in more complex applications, for example, to visualize the difference in development between two sets of predicted values, such as the difference in development of unemployment in two countries over time. In this article, I presented a command, emc, that implements this approach using restricted cubic splines.

Other techniques than restricted cubic splines, such as fractional polynomials (Royston and Sauerbrei 2008; StataCorp 2017), are alternatives.

9 Acknowledgment

I thank Professor Henrik Støvring, Department of Public Health, Aarhus University for help and comments. Thanks also to the reviewer for all his help in the review process.

10 Programs and supplemental materials

To install a snapshot of the corresponding software files as they existed at the time of publication of this article, type

```
. net sj 19-3
. net install st0567     (to install program files, if available)
. net get st0567         (to install ancillary files, if available)
```

11 References

Buis, M. L. 2008. postrcspline: Stata module containing post-estimation commands for models using a restricted cubic spline. Statistical Software Components S456928, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s456928.html.

———. 2009. Using and interpreting restricted cubic splines. http://maartenbuis.nl/presentations/bonn09.pdf.

Croxford, R. 2016. Restricted cubic spline regression: A brief introduction. Paper 5621-2016. SAS Global Forum 2016. https://support.sas.com/resources/papers/proceedings16/5621-2016.pdf.

Dziak, J. J., D. L. Coffman, S. T. Lanza, and R. Li. 2012. Sensitivity and specificity of information criteria. Technical Report Series No. 12-119, Pennsylvania State University. https://www.methodology.psu.edu/files/2019/03/12-119-2e90hc6.pdf.

Harrell, F. E., Jr. 2015. *Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*. 2nd ed. Cham, Switzerland: Springer.

Mitchell, M. N. 2012. *Interpreting and Visualizing Regression Models Using Stata*. College Station, TX: Stata Press.

Orsini, N., and S. Greenland. 2011. A procedure to tabulate and plot results after flexible modeling of a quantitative covariate. *Stata Journal* 11: 1–29.

PennState, The Methodology Center. 2012. AIC vs. BIC. https://www.methodology.psu.edu/resources/AIC-vs-BIC/.

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

StataCorp. 2017. *Stata 15 Base Reference Manual*. College Station, TX: Stata Press.

About the author

Niels Henrik Bruun has a master's degree in mathematics and statistics and a bachelor's degree in microeconomics, both from the University of Aarhus. He has taught applied statistics at the undergraduate level and worked as a consultant in industrial statistical process control. He has also worked at one of the largest banks in Denmark as a consultant in financial risk reporting. He now works as a statistical consultant at the Department of Public Health at Aarhus University, where he also teaches biostatistics for medical students at the undergraduate level. He maintains two websites: Stata Hacks (http://www.bruunisejs.dk/StataHacks/) and Python Hacks (http://www.bruunisejs.dk/PythonHacks/).