

The effx function for measuring effects

Michael Hills

October 9, 2007

1 Introduction

Identifying the response variable correctly is the key to analysis. The main types are:

- Metric (a measurement taking many values, usually with units)
- Binary (two values coded 0/1)
- Failure (does the subject fail at end of follow-up, and how long was follow-up)
- Count (aggregated failure data)

The response variable must be numeric.

Variables on which the response may depend are called explanatory variables. They can be factors or numeric. A further important aspect of explanatory variables is the role they will play in the analysis.

- Primary role: exposure
- Secondary role: confounder

The word effect is a general term referring to ways of comparing the values of the response variable at different levels of an explanatory variable. The main measures of effect are:

- Differences in means for a metric response.
- Ratios of odds for a binary response.
- Ratios of rates for a failure or count response.

What other measures of effects might be used?

2 The function `effx`

The function `effx` is intended to introduce the estimation of effects in epidemiology, together with the related ideas of stratification and controlling, without the need for familiarity with statistical modelling.

We shall use the `births` data in the `Epi` package, which can be loaded and inspected with

```
> library(Epi)
> data(births)
> help(births)
```

The variables we shall be interested in are `bweight` (birth weight) and `hyp` (hypertension). An alternative way of characterizing birth weight is shown in `lowbw` which is coded 1 for babies with low birth weight, and 0 otherwise. Other variables of interest are `sex` (of the baby) and `gestwks`, the gestation time. All variables are numeric, so first we need first to do a little housekeeping:

```
> births$hyp <- factor(births$hyp, labels = c("normal", "hyper"))
> births$sex <- factor(births$sex, labels = c("M", "F"))
> births$agegrp <- cut(births$matage, breaks = c(20, 25, 30, 35,
+      40, 45), right = FALSE)
> births$gest4 <- cut(births$gestwks, breaks = c(20, 35, 37, 39,
+      45), right = FALSE)
```

Now try

```
> effx(response = bweight, typ = "metric", exposure = sex, data = births)
```

The effect of `sex` on birth weight, measured as a difference in means, is -197 . The command

```
> stat.table(sex, mean(bweight), data = births)
```

verifies this ($3032.8 - 3229.9 = -197.1$). The p-value refers to the test that there is no effect of `sex` on birth weight. Use `effx` to find the effect of `hyp` on `bweight`.

For another example, consider the effect of `sex` on the binary response `lowbw`.

```
> effx(response = lowbw, typ = "binary", exposure = sex, data = births)
```

The effect of `sex` on `lowbw`, measured as an odds ratio, is 1.43. The command

```
> stat.table(sex, list(odds = ratio(lowbw, 1 - lowbw, 100)), data = births)
```

can be used to verify this ($16.26/11.39 = 1.427$). Use `effx` to find the effect of `hyp` on `lowbw`.

3 Factors on more than two levels

The variable `gest4` is the result of cutting `gestwks` into 4 groups with boundaries [20,35) [35,37) [37,39) [39,45). We shall find the effects of `gest4` on the metric response `bweight`.

```
> effx(response = bweight, typ = "metric", exposure = gest4, data = births)
```

There are now 3 effects

```
[35,37) vs [20,35) 856.6
[37,39) vs [20,35) 1360.0
[39,45) vs [20,35) 1668.0
```

The command

```
> stat.table(gest4, mean(bweight), data = births)
```

verifies that the effect of `agegrp` (level 2 vs level 1) is $2590 - 1733 = 857$, etc. Find the effects of `gest4` on `lowbw`. Use the option `base=4` to change the baseline for `gest4` from 1 to 4.

4 Stratified effects

As an example we shall stratify the effects of `hyp` on `bweight` by `sex` with

```
> effx(bweight, type = "metric", exposure = hyp, strata = sex,
+      data = births)
```

The effects of `hyp` in the different strata defined by `sex` are -496 and -380 .

Use `effx` to stratify the effect of `hyp` on `lowbw` first by `sex` and then by `gest4`.

5 Controlling the effect of `hyp` for `sex`

The effect of `hyp` is controlled for `sex` by first looking at the effects of `hyp` in the two strata defined by `sex`, and then combining these effects if they are similar. In this case the effects were -496 and -380 which look similar (the test for effect modification is a test of whether they differ significantly) so we can combine them, and control for `sex`. The combining is done by declaring `sex` as a control variable:

```
> effx(bweight, type = "metric", exposure = hyp, control = sex,
+      data = births)
```

The effect of `hyp` on `bweight` controlled for `sex` is -448 . Note that it is the name of the control variable which is passed, not the variable itself. There can be more than one control variable, `control=list(sex,agegrp)`.

Many people go straight ahead and control for variables which are likely to confound the effect of exposure without bothering to stratify first, but there are times when it is useful to stratify first.

6 Numeric exposures

If we wished to study the effect of gestation time on the baby's birth weight then `gestwks` is a numeric exposure. Assuming that the relationship of the response with `gestwks` is roughly linear (for a metric response) or log-linear (for a binary response) we can find the linear effect of `gestwks`.

```
> effx(response = bweight, type = "metric", exposure = gestwks,  
+       data = births)
```

The linear effect of `gestwks` is 197 g per extra week of gestation. The linear effect of `gestwks` on `lowbw` can be found similarly

```
> effx(response = lowbw, type = "binary", exposure = gestwks, data = births)
```

The linear effect of `gestwks` on `lowbw` is a reduction by a factor of 0.408 per extra week of gestation, i.e. the odds of a baby having a low birth weight is reduced by a factor of 0.408 per one week increase in gestation.

You cannot stratify by a numeric variable, but you can study the effects of a numeric exposure stratified by (say) `agegrp` with

```
> effx(lowbw, type = "binary", exposure = gestwks, strata = agegrp,  
+       data = births)
```

You can control for a numeric variable by putting it in `control=`.

7 Checking on linearity

At this stage it will be best to make a visual check using `plot`. For example, to check whether `bweight` goes up linearly with `gestwks` try

```
> with(births, plot(gestwks, bweight))
```

Is the relationship roughly linear? It is not possible to check graphically whether log odds of a baby being low birth weight goes down linearly with gestation because the individual odds are either 0 or ∞ . Instead we use the grouped variable `gest4`:

```
> tab <- stat.table(gest4, ratio(lowbw, 1 - lowbw, 100), data = births)
> str(tab)
> odds <- tab[1, 1:4]
> plot(1:4, log(odds), type = "b")
```

The relationship is remarkably linear, but remember this is quite crude because it takes no account of unequal gestation intervals. More about checking for linearity later.

8 Frequency data

Data from very large studies are often summarized in the form of frequency data, which records the frequency of all possible combinations of values of the variables in the study. Such data are sometimes presented in the form of a contingency table, sometimes as a data frame in which one variable is the frequency. As an example, consider the `UCBAdmissions` data, which is one of the standard R data sets, and refers to the outcome of applications to 6 departments by gender. The command

```
> UCBAdmissions
```

shows that the data are in the form of a $2 \times 2 \times 6$ contingency table for the three variables `Admit` (admitted/rejected), `Gender` (male/female), and `Dept` (A/B/C/D/E/F). Thus in department A 512 males were admitted while 312 were rejected, and so on. The question of interest is whether there is any bias against admitting female applicants.

The command

```
> ucb <- as.data.frame(UCBAdmissions)
> head(ucb)
```

coerces the contingency table to a data frame, and shows the first 10 lines. The relationship between the contingency table and the data frame should be clear. The command

```
> ucb$Admit <- as.numeric(ucb$Admit) - 1
```

turns `Admit` into a numeric variable coded 1 for rejection, 0 for admission, so

```
> effx(Admit, type = "binary", exposure = Gender, weights = Freq,
+      data = ucb)
```

shows the odds of rejection for female applicants to be 1.84 times the odds for males (note the use of `weights` to take account of the frequencies). A crude analysis therefore suggests there is a strong bias against admitting females. Continue the analysis by stratifying the crude analysis by department - does this still support a bias against females? What is the effect of gender controlled for department?