Interpreting statistical models using simple graphics

Graeme Hutcheson

Manchester Institute of Education
University of Manchester

This session aims to provide an introduction to interpreting and communicating statistical models using graphics.

This session aims to provide an introduction to interpreting and communicating statistical models using graphics.

Effect displays (accessible through John Fox's **effects** package) are used to illustrate models using tables or graphs which represent terms in a model and are designed to make the task of interpreting them much simpler.

This session aims to provide an introduction to interpreting and communicating statistical models using graphics.

Effect displays (accessible through John Fox's **effects** package) are used to illustrate models using tables or graphs which represent terms in a model and are designed to make the task of interpreting them much simpler.

These displays are quick and easy to produce and manipulate and circumvent many of the problems analysts typically have with interpreting statistical models and with communicating them to wider audiences.

This session aims to provide an introduction to interpreting and communicating statistical models using graphics.

Effect displays (accessible through John Fox's **effects** package) are used to illustrate models using tables or graphs which represent terms in a model and are designed to make the task of interpreting them much simpler.

These displays are quick and easy to produce and manipulate and circumvent many of the problems analysts typically have with interpreting statistical models and with communicating them to wider audiences.

Their ease of use and the intuitive way they illustrate relationships also makes them ideal tools for teaching and learning.

A basic linear regression model

A linear regression model predicting the quality of witness statements...

```
glm(formula = QUALITY ~ MATURITY + LOCATION + GENDER + AGE,
     family = gaussian(identity),
       data = witness)
```

A linear regression model predicting the quality of witness statements...

```
glm(formula = QUALITY ~ MATURITY + LOCATION + GENDER + AGE,
     family = gaussian(identity),
       data = witness)

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         49.1805     3.1627  15.550  < 2e-16 ***
MATURITY             3.1232     1.2820   2.436   0.0177 *
LOCATION[S.Formal]  -2.5990     1.7464  -1.488   0.1417
LOCATION[S.Home]    -0.9194     1.7480  -0.526   0.6008
LOCATION[S.School]  -1.4372     1.8311  -0.785   0.4355
GENDERmale          -2.5332     2.1389  -1.184   0.2407
AGE8-9 years        10.7861     2.1535   5.009  4.7e-06 ***
```

A linear regression model predicting the quality of witness statements...

```
glm(formula = QUALITY ~ MATURITY + LOCATION + GENDER + AGE,
     family = gaussian(identity),
       data = witness)

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         49.1805     3.1627  15.550  < 2e-16 ***
MATURITY             3.1232     1.2820   2.436   0.0177 *
LOCATION[S.Formal]  -2.5990     1.7464  -1.488   0.1417
LOCATION[S.Home]    -0.9194     1.7480  -0.526   0.6008
LOCATION[S.School]  -1.4372     1.8311  -0.785   0.4355
GENDERmale          -2.5332     2.1389  -1.184   0.2407
AGE8-9 years        10.7861     2.1535   5.009  4.7e-06 ***
```

It can be difficult to appreciate the important relationships in the model (particularly with categorical explanatory variables) and communicate these results to non-specialists.

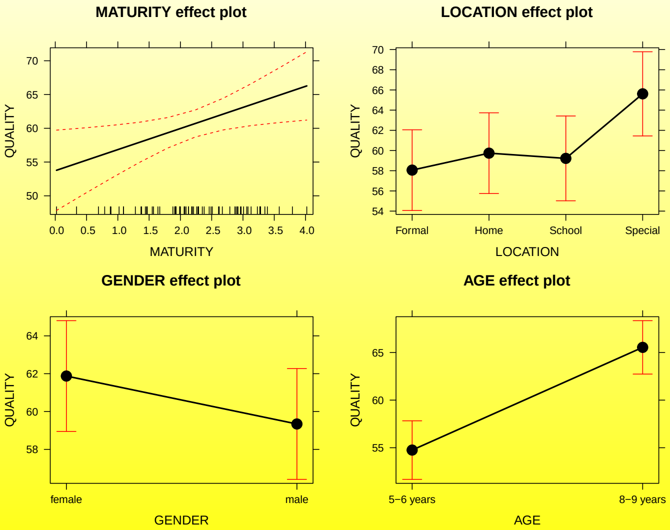
The **effects** package allows the information shown in the regression output above to be displayed using graphics...

The **effects** package allows the information shown in the regression output above to be displayed using graphics...

```
library(effects)

plot(allEffects(witness.01))
```

Effect Display: QUALITY ∼ MATURITY + LOCATION + GENDER + AGE



The effect display gives the same information as the regression output, but provides it in a more intuitive way and also provides information that is hidden in the original output...

The effect display gives the same information as the regression output, but provides it in a more intuitive way and also provides information that is hidden in the original output...

In particular, the default reference category (the special interview room) may not be an ideal choice for this analysis (treatment contrasts may be more informative).

A linear regression model with a three-way interaction

A linear regression model predicting the quantity of ice cream sold...

```
glm(formula = Consumption ~ Price * Temperature * Income,
     family = gaussian(identity),
       data = iceCREAM)
```

A linear regression model predicting the quantity of ice cream sold...

```
glm(formula = Consumption ~ Price * Temperature * Income,
     family = gaussian(identity),
       data = iceCREAM)

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              20.665656   7.970792   2.593 0.016613 *
Price                   -76.452035  29.171664  -2.621 0.015610 *
Temperature              -0.757488   0.180361  -4.200 0.000370 ***
Income                   -0.250381   0.092277  -2.713 0.012692 *
Price:Temperature         2.818488   0.664467   4.242 0.000334 ***
Price:Income              0.935945   0.337905   2.770 0.011174 *
Temperature:Income        0.009276   0.002130   4.355 0.000253 ***
Price:Temperature:Income -0.034391   0.007858  -4.377 0.000240 ***
```

A linear regression model predicting the quantity of ice cream sold...

```
glm(formula = Consumption ~ Price * Temperature * Income,
     family = gaussian(identity),
       data = iceCREAM)

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              20.665656   7.970792   2.593 0.016613 *
Price                   -76.452035  29.171664  -2.621 0.015610 *
Temperature              -0.757488   0.180361  -4.200 0.000370 ***
Income                   -0.250381   0.092277  -2.713 0.012692 *
Price:Temperature         2.818488   0.664467   4.242 0.000334 ***
Price:Income              0.935945   0.337905   2.770 0.011174 *
Temperature:Income        0.009276   0.002130   4.355 0.000253 ***
Price:Temperature:Income -0.034391   0.007858  -4.377 0.000240 ***
```

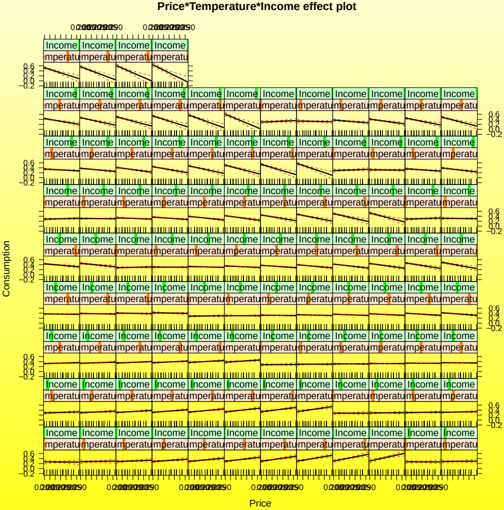
It challenging to interpret the three-way interaction using this output...

Effect displays are particularly useful for interpreting interactions and can be easily plotted using the pull-down menu in the Rcmdr or by using the command...

Effect displays are particularly useful for interpreting interactions and can be easily plotted using the pull-down menu in the Rcmdr or by using the command...

```
plot(allEffects(iceCREAM.01))
```

Effect Display: consumption ∼ price*income*temperature

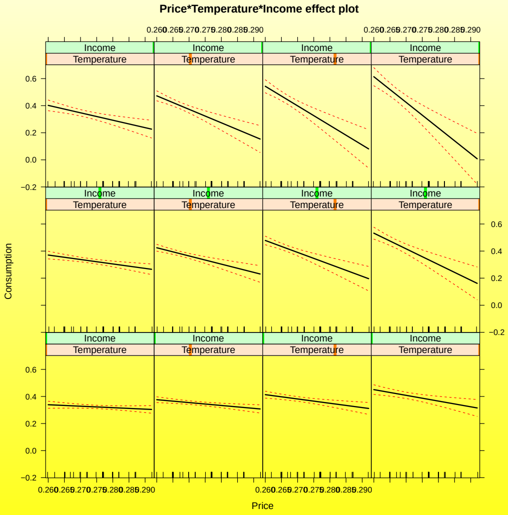


The effect display is difficult to interpret as there are too many panels. This can be easily remedied by defining the number of panels using the `"xlevels="` function. The following defines 4 panels for temperature (40, 50, 60 and 70) and 3 panels for income (85, 90 and 95).

The effect display is difficult to interpret as there are too many panels. This can be easily remedied by defining the number of panels using the "xlevels=" function. The following defines 4 panels for temperature (40, 50, 60 and 70) and 3 panels for income (85, 90 and 95).

```
plot(allEffects(iceCREAM.01,
            xlevels=list(
              Temperature=seq(40,70,10),
              Income=seq(85,95,5))
     ))
```

Effect Display: consumption ∼ price*income*temperature



Interpreting the three-way interaction is now much easier.

Interpreting the three-way interaction is now much easier.

At high temperatures, those with high incomes are able to exercise a choice about whether they buy ice cream. This choice is based, partly, on the price.

Interpreting the three-way interaction is now much easier.

At high temperatures, those with high incomes are able to exercise a choice about whether they buy ice cream. This choice is based, partly, on the price.

This relationship is very hard to identify using the original model output.

HINT...

HINT...

The layout of the graphs (i.e., whether they are plotted in a 3-by-4 matrix, or a 1-by-12 matrix) is dictated by a number of rules which can be defined by the user.

HINT...

The layout of the graphs (i.e., whether they are plotted in a 3-by-4 matrix, or a 1-by-12 matrix) is dictated by a number of rules which can be defined by the user.

The layout can, however, be easily manipulated by changing the dimensions of the R-studio output window (the plot window).

HINT...

The layout of the graphs (i.e., whether they are plotted in a 3-by-4 matrix, or a 1-by-12 matrix) is dictated by a number of rules which can be defined by the user.

The layout can, however, be easily manipulated by changing the dimensions of the R-studio output window (the plot window).

For example, to plot just the top 3 panels (when `income=95`), all side by side, change the dimensions of the plot window so that it is short and wide and then run the command...

HINT...

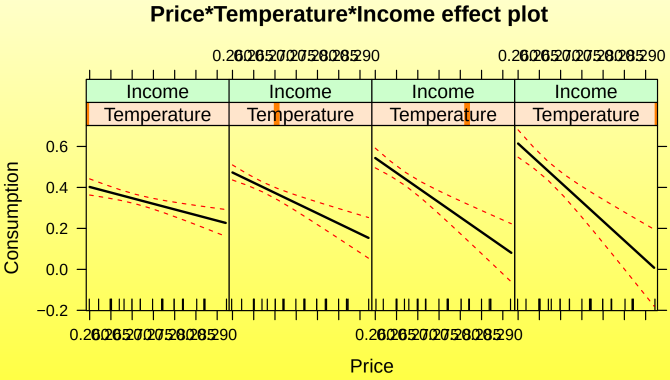
The layout of the graphs (i.e., whether they are plotted in a 3-by-4 matrix, or a 1-by-12 matrix) is dictated by a number of rules which can be defined by the user.

The layout can, however, be easily manipulated by changing the dimensions of the R-studio output window (the plot window).

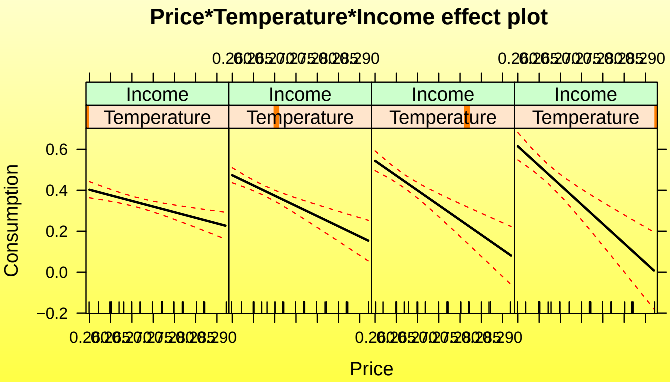
For example, to plot just the top 3 panels (when `income=95`), all side by side, change the dimensions of the plot window so that it is short and wide and then run the command...

```
plot(allEffects(iceCREAM.01,
      xlevels=list(
       Temperature=seq(40,70,10),
       Income=seq(95,95,0))
    ))
```

Effect Display: consumption ∼ price*income*temperature



Effect Display: consumption $\sim$ price*income*temperature



Note: it is easy to edit the graphic to improve the presentation (consult the **effects** documentation or look at the **tikzDevice** package).

A generalised linear logit regression model

A logistic regression model predicting the probability of being released using the `Arrests` dataset from the **effects** package (`data(Arrests)`).

```
glm(formula = released ~ checks + colour + sex + yearCAT,
     family = binomial(logit),
       data = Arrests)
```

A logistic regression model predicting the probability of being released using the `Arrests` dataset from the **effects** package (`data(Arrests)`).

```
glm(formula = released ~ checks + colour + sex + yearCAT,
     family = binomial(logit),
       data = Arrests)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)      1.55599    0.19659   7.915 2.48e-15 ***
checks          -0.40367    0.02516 -16.044  < 2e-16 ***
colour[T.White]  0.54187    0.08183   6.622 3.54e-11 ***
sex[T.Male]      0.09156    0.14711   0.622   0.5337
yearCAT[T.1998]  0.34079    0.14471   2.355   0.0185 *
yearCAT[T.1999]  0.36675    0.13958   2.627   0.0086 **
yearCAT[T.2000]  0.57144    0.13926   4.103 4.07e-05 ***
yearCAT[T.2001]  0.33515    0.13688   2.448   0.0143 *
yearCAT[T.2002]  0.17366    0.19278   0.901   0.3677
```

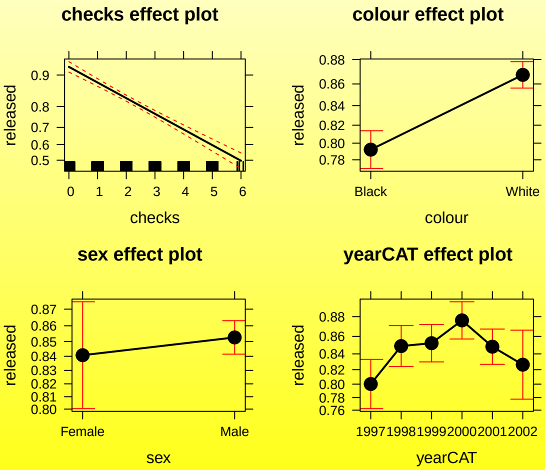
A logistic regression model predicting the probability of being released using the Arrests dataset from the **effects** package (data(Arrests)).

```
glm(formula = released ~ checks + colour + sex + yearCAT,
     family = binomial(logit),
       data = Arrests)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)      1.55599    0.19659   7.915 2.48e-15 ***
checks          -0.40367    0.02516 -16.044  < 2e-16 ***
colour[T.White]  0.54187    0.08183   6.622 3.54e-11 ***
sex[T.Male]      0.09156    0.14711   0.622   0.5337
yearCAT[T.1998]  0.34079    0.14471   2.355   0.0185 *
yearCAT[T.1999]  0.36675    0.13958   2.627   0.0086 **
yearCAT[T.2000]  0.57144    0.13926   4.103 4.07e-05 ***
yearCAT[T.2001]  0.33515    0.13688   2.448   0.0143 *
yearCAT[T.2002]  0.17366    0.19278   0.901   0.3677
```

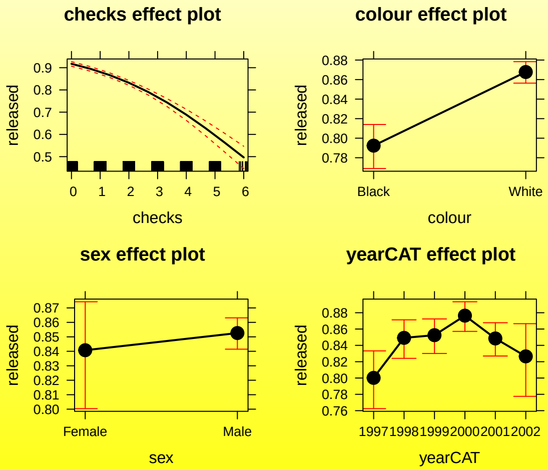
This is not easy to interpret, particularly as the estimates are provided in logits.

```
plot(allEffects(arrests.01))
```



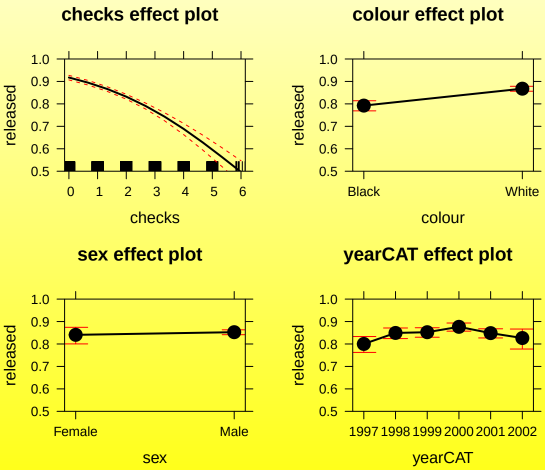
Effect Display: released ∼ checks + colour + gender + year

```
plot(allEffects(arrests.01), rescale.axis=FALSE)
```



Effect Display: released ∼ checks + colour + gender + year

```
plot(allEffects(arrests.01), rescale.axis=FALSE, ylim=c(0.5,1))
```



A particularly useful feature of the **effects** package is being able to define values of the explanatory variables.

A particularly useful feature of the **effects** package is being able to define values of the explanatory variables.

With regards to the variable `colour`, the effect displays default is to provide predictions that represent the average mix of black and white people in the sample (a factor is fixed "at it's proportional distribution in the data"; Fox, 2003).

A particularly useful feature of the **effects** package is being able to define values of the explanatory variables.

With regards to the variable `colour`, the effect displays default is to provide predictions that represent the average mix of black and white people in the sample (a factor is fixed "at it's proportional distribution in the data"; Fox, 2003).

Whist this is useful for describing the model, it can also be useful to derive predictions for black and white people individually.

A particularly useful feature of the **effects** package is being able to define values of the explanatory variables.

With regards to the variable `colour`, the effect displays default is to provide predictions that represent the average mix of black and white people in the sample (a factor is fixed "at it's proportional distribution in the data"; Fox, 2003).

Whist this is useful for describing the model, it can also be useful to derive predictions for black and white people individually.

This can be achieved using the `"given.values = "` option. To obtain predictions for white people, set the proportion of white people to 1. To obtain predictions for black people, set the proportion of white people to 0.

A particularly useful feature of the **effects** package is being able to define values of the explanatory variables.

With regards to the variable `colour`, the effect displays default is to provide predictions that represent the average mix of black and white people in the sample (a factor is fixed "at it's proportional distribution in the data"; Fox, 2003).

Whist this is useful for describing the model, it can also be useful to derive predictions for black and white people individually.

This can be achieved using the `"given.values = "` option. To obtain predictions for white people, set the proportion of white people to 1. To obtain predictions for black people, set the proportion of white people to 0.

Note: the defined category cannot be the reference category.

To get effect displays for white people...

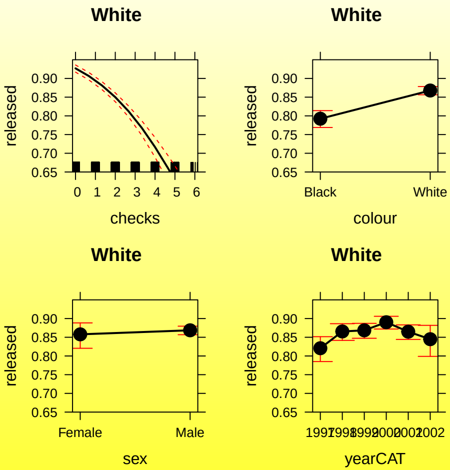
```
plot(allEffects(arrests.01,
      given.values = c(colourWhite = 1)),
      rescale.axis = FALSE,
              ylim = c(0.65,0.95),
              main = "White")
```

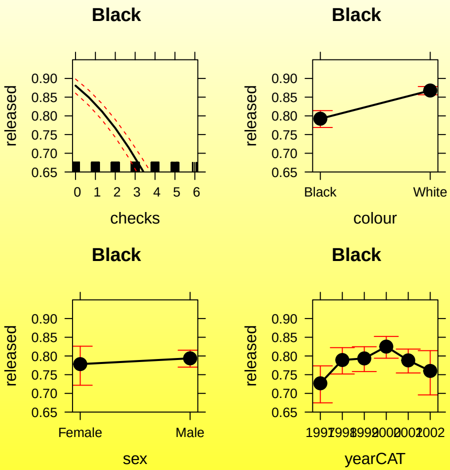
To get effect displays for white people...

```
plot(allEffects(arrests.01,
      given.values = c(colourWhite = 1)),
      rescale.axis = FALSE,
              ylim = c(0.65,0.95),
              main = "White")
```

and for black people...

```
plot(allEffects(arrests.01,
      given.values = c(colourWhite = 0)),
      rescale.axis = FALSE,
              ylim = c(0.65,0.95),
              main = "Black")
```


White
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
White
released
Black
White
colour
White
released
Female
Male
sex
White
released
yearCAT


Black
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
Black
released
Black
White
colour
Black
released
Female
Male
sex
Black
released
1997 1998 1999 2000 2001 2002
yearCAT

The ability to define categories enables animations to be constructed. For example, to animate what happens across a number of years, we simply need to define the years.

The ability to define categories enables animations to be constructed. For example, to animate what happens across a number of years, we simply need to define the years.

For 1997...

```
plot(allEffects(arrests.01,
      given.values = c(yearCAT1998 = 0, yearCAT1999 = 0,
                       yearCAT2000 = 0, yearCAT2001 = 0,
                       yearCAT2002 = 0)),
      rescale.axis = FALSE, ylim = c(0.65,0.95), main = "1997")
```

The ability to define categories enables animations to be constructed. For example, to animate what happens across a number of years, we simply need to define the years.

For 1997...

```
plot(allEffects(arrests.01,
      given.values = c(yearCAT1998 = 0, yearCAT1999 = 0,
                       yearCAT2000 = 0, yearCAT2001 = 0,
                       yearCAT2002 = 0)),
      rescale.axis = FALSE, ylim = c(0.65,0.95), main = "1997")
```

For 1998...

```
plot(allEffects(arrests.01,
      given.values = c(yearCAT1998 = 1)),
      rescale.axis = FALSE, ylim = c(0.65,0.95), main = "1998")
```

The ability to define categories enables animations to be constructed. For example, to animate what happens across a number of years, we simply need to define the years.

For 1997...

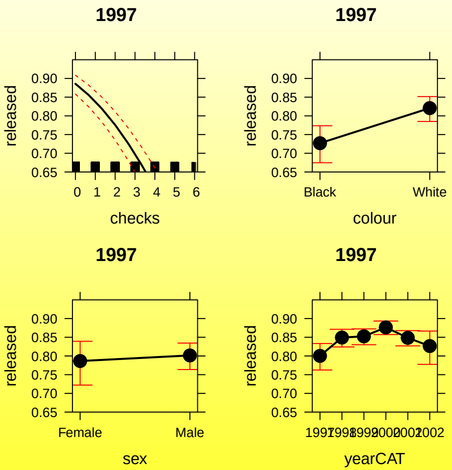
```
plot(allEffects(arrests.01,
      given.values = c(yearCAT1998 = 0, yearCAT1999 = 0,
                       yearCAT2000 = 0, yearCAT2001 = 0,
                       yearCAT2002 = 0)),
      rescale.axis = FALSE, ylim = c(0.65,0.95), main = "1997")
```

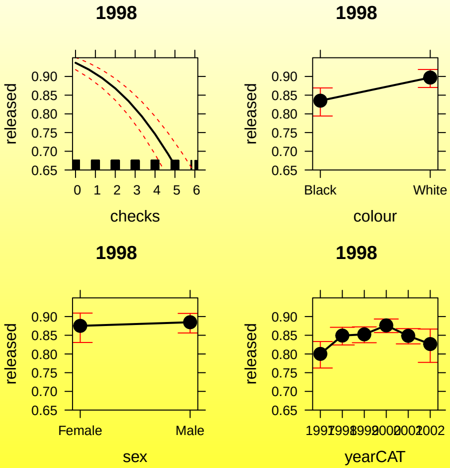
For 1998...

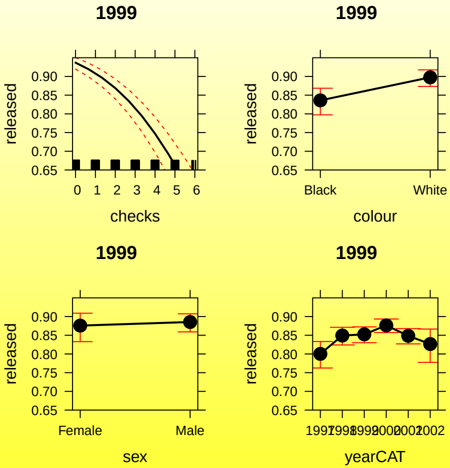
```
plot(allEffects(arrests.01,
      given.values = c(yearCAT1998 = 1)),
      rescale.axis = FALSE, ylim = c(0.65,0.95), main = "1998")
```

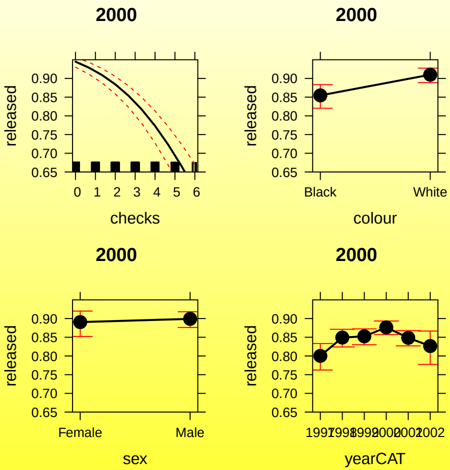
For 1999...

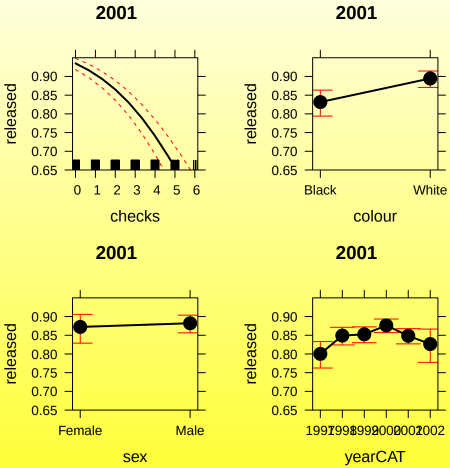
```
plot(allEffects(arrests.01,
      given.values = c(yearCAT1999 = 1)),
      rescale.axis = FALSE, ylim = c(0.65,0.95), main = "1999")
```

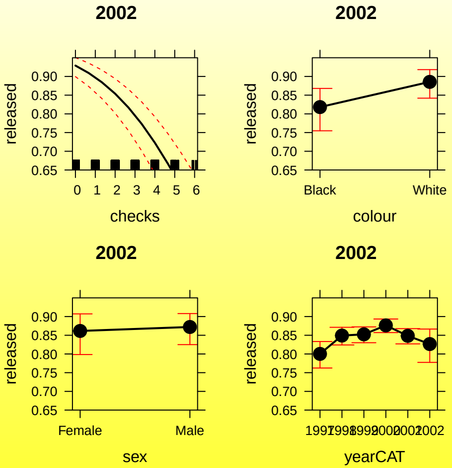

1997
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
1997
released
Black
White
colour
1997
released
Female
Male
sex
1997
released
1997 1998 1999 2000 2001 2002
yearCAT


1998
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
1998
released
Black
White
colour
1998
released
Female
Male
sex
1998
released
yearCAT


1999
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
1999
released
Black
White
colour
1999
released
Female
Male
sex
1999
released
1997 1998 1999 2000 2001 2002
yearCAT


2000
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
2000
released
Black
White
colour
2000
released
Female
Male
sex
2000
released
1997 1998 1999 2000 2001 2002
yearCAT


2001
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
2001
released
Black
White
colour
2001
released
Female
Male
sex
2001
released
1997 1998 1999 2000 2001 2002
yearCAT


2002
released
0.90
0.85
0.80
0.75
0.70
0.65
0 1 2 3 4 5 6
checks
2002
released
Black
White
colour
2002
released
Female
Male
sex
2002
released
1997 1998 1999 2000 2001 2002
yearCAT

The main difference highlighted in the animation appears to be between 1997 and 1998. This is something that is quite hidden in the standard output...

A Poisson model with interactions

`checks` is an interesting variable to model.

`checks` is an interesting variable to model.

`checks` is a count variable and can be modelled using a Poisson regression model (log-linear).

`checks` is an interesting variable to model.

`checks` is a count variable and can be modelled using a Poisson regression model (log-linear).

The model shown below includes the variables `age`, `colour`, `yearCAT` and `sex`, including interactions between `age` and `colour` and between `colour` and `yearCAT`.

`checks` is an interesting variable to model.

`checks` is a count variable and can be modelled using a Poisson regression model (log-linear).

The model shown below includes the variables `age`, `colour`, `yearCAT` and `sex`, including interactions between `age` and `colour` and between `colour` and `yearCAT`.

The standard output is difficult to interpret and provides limited information about the relationships in the model. The effect displays, on the other hand, are easy to understand and provide a greatly enhanced picture of the data. They even suggest an interesting interaction between `colour` and `yearCAT` that is not evident in the standard output.

Poisson Model: checks ∼ age*colour + colour*year + gender

	Estimate	Std. Error	z value	Pr(>|z|)	
(Intercept)	2.909e-01	9.448e-02	3.079	0.00208	**
age	3.919e-03	2.180e-03	1.798	0.07222	.
colour[T.White]	-5.351e-01	9.882e-02	-5.415	6.13e-08	***
yearCAT[T.1998]	-5.264e-02	7.619e-02	-0.691	0.48966	
yearCAT[T.1999]	7.005e-03	7.479e-02	0.094	0.92538	
yearCAT[T.2000]	-7.169e-05	7.347e-02	-0.001	0.99922	
yearCAT[T.2001]	-6.767e-02	7.311e-02	-0.926	0.35462	
yearCAT[T.2002]	-3.251e-02	9.701e-02	-0.335	0.73751	
sex[T.Male]	3.977e-01	4.692e-02	8.478	< 2e-16	***
age:colour[T.White]	1.328e-02	2.621e-03	5.069	4.00e-07	***
colour[T.White]:yearCAT[T.1998]	-4.342e-02	9.165e-02	-0.474	0.63571	
colour[T.White]:yearCAT[T.1999]	-1.704e-01	8.927e-02	-1.909	0.05629	.
colour[T.White]:yearCAT[T.2000]	-1.586e-01	8.756e-02	-1.811	0.07010	.
colour[T.White]:yearCAT[T.2001]	-1.210e-01	8.770e-02	-1.380	0.16770	
colour[T.White]:yearCAT[T.2002]	-1.665e-01	1.212e-01	-1.374	0.16947	

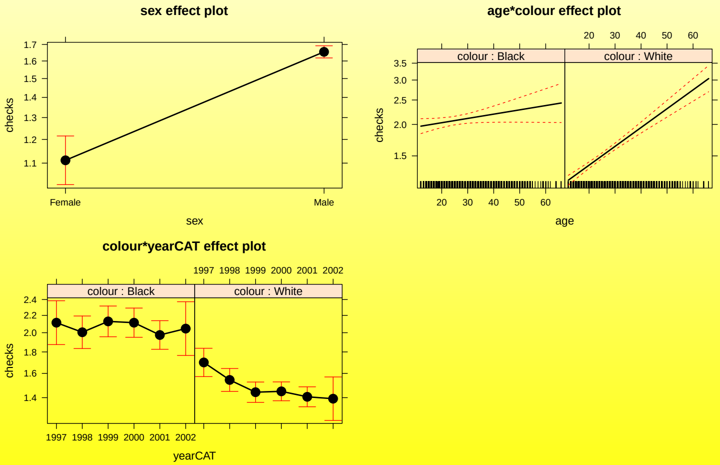
Poisson Model: checks $\sim$ age*colour + colour*year + gender

```
                               Estimate Std. Error z value Pr(>|z|)
(Intercept)                   2.909e-01  9.448e-02   3.079  0.00208 **
age                           3.919e-03  2.180e-03   1.798  0.07222 .
colour[T.White]              -5.351e-01  9.882e-02  -5.415 6.13e-08 ***
yearCAT[T.1998]              -5.264e-02  7.619e-02  -0.691  0.48966
yearCAT[T.1999]               7.005e-03  7.479e-02   0.094  0.92538
yearCAT[T.2000]              -7.169e-05  7.347e-02  -0.001  0.99922
yearCAT[T.2001]              -6.767e-02  7.311e-02  -0.926  0.35462
yearCAT[T.2002]              -3.251e-02  9.701e-02  -0.335  0.73751
sex[T.Male]                   3.977e-01  4.692e-02   8.478  < 2e-16 ***
age:colour[T.White]           1.328e-02  2.621e-03   5.069 4.00e-07 ***
colour[T.White]:yearCAT[T.1998] -4.342e-02  9.165e-02  -0.474  0.63571
colour[T.White]:yearCAT[T.1999] -1.704e-01  8.927e-02  -1.909  0.05629 .
colour[T.White]:yearCAT[T.2000] -1.586e-01  8.756e-02  -1.811  0.07010 .
colour[T.White]:yearCAT[T.2001] -1.210e-01  8.770e-02  -1.380  0.16770
colour[T.White]:yearCAT[T.2002] -1.665e-01  1.212e-01  -1.374  0.16947
```

```
Type II tests
Response: checks
              LR Chisq Df Pr(>Chisq)
age            108.406  1  < 2.2e-16 ***
colour         175.067  1  < 2.2e-16 ***
yearCAT         15.176  5   0.009637 **
sex             80.859  1  < 2.2e-16 ***
age:colour      26.308  1  2.911e-07 ***
colour:yearCAT   6.445  5   0.265274
```

Effect Display: checks $\sim$ age*colour + colour*year + gender

`plot(allEffects(arrests.02))`



Conclusions

- Effect displays are easy to produce and often provide much clearer and more detailed information than the standard output.

- Effect displays are easy to produce and often provide much clearer and more detailed information than the standard output.

- Models can be interpreted without reference to logits, contrast coding, reference categories, type II or III ANOVA tests, lower-order interactions, or model-fit statistics.

- Effect displays are easy to produce and often provide much clearer and more detailed information than the standard output.
- Models can be interpreted without reference to logits, contrast coding, reference categories, type II or III ANOVA tests, lower-order interactions, or model-fit statistics.
- Effect displays can be applied to all generalized linear models; these include the 'standard' t-test, ANOVA, ANCOVA, Mann-Whitney, Friedman, chi-square, log-linear, proportional-odds, multinomial logit and mixed-effect models.

- Effect displays are easy to produce and often provide much clearer and more detailed information than the standard output.
- Models can be interpreted without reference to logits, contrast coding, reference categories, type II or III ANOVA tests, lower-order interactions, or model-fit statistics.
- Effect displays can be applied to all generalized linear models; these include the 'standard' t-test, ANOVA, ANCOVA, Mann-Whitney, Friedman, chi-square, log-linear, proportional-odds, multinomial logit and mixed-effect models.
- Effect displays enable models to be communicated to non-specialists and also encourage dialogue about the 'meaning' of models.