Friday, March 28

Discrete Marginal Effects

Consider a regression model with (without loss of generality) two explanatory variables, X_1 and X_2. A *discrete marginal effect* is the change in the expected response when we change an explanatory variable.

For example, if we have a regression model where $E(Y)$ is a function of X_1 and X_2, the discrete marginal effect of changing X_1 from x_b to x_a is

$$E(Y|X_1 = x_a, X_2 = x_2) - E(Y|X_1 = x_b, X_2 = x_2).$$

That is, the change in the expected response when X_1 is changed from x_b to x_a. (Note: When we talk about a change in the expected response or the "effect" of a change in an explanatory variable, we do not necessarily mean that this is a *causal* relationship.)

In a linear model a discrete marginal effect is basically what is done by `contrast`.

Example: Recall our model for the `whiteside` data. The function `margeff` in the **trtools** package will estimate a discrete marginal effect.

```
m <- lm(Gas ~ Insul + Temp + Insul:Temp, data = MASS::whiteside)
summary(m)$coefficients
```

```
                Estimate Std. Error t value Pr(>|t|)
(Intercept)        6.854     0.1360   50.41 8.00e-46
InsulAfter        -2.130     0.1801  -11.83 2.32e-16
Temp              -0.393     0.0225  -17.49 1.98e-23
InsulAfter:Temp    0.115     0.0321    3.59 7.31e-04
```

The model is

$$E(Y_i) = \beta_0 + \beta_1 a_i + \beta_2 t + \beta_3 a_i t_i,$$

where Y_i is gass consumption,

$$a_i = \begin{cases} 1, & \text{if the } i\text{-th observation is after insulation,} \\ 0, & \text{otherwise.} \end{cases}$$

So the marginal effect of increasing temperature from $t_b = 2$ to $t_a = 7$ after insulation is

$$E(Y|a = 1, t = 7) - E(Y|a = 1, t = 2) = 5(\beta_2 + \beta_3).$$

Before insulation it is

$$E(Y|a = 0, t = 7) - E(Y|a = 0, t = 2) = 5\beta_2.$$

We can estimate this using the `lincon` or `contrast` functions.

```
library(trtools)
lincon(m, a = c(0,0,5,5)) # marginal effect after insulation
```

```
            estimate    se lower upper tvalue df   pvalue
(0,0,5,5),0    -1.39 0.115 -1.62 -1.16  -12.1 52 8.94e-17
```

```
lincon(m, a = c(0,0,5,0)) # marginal effect after insulation
```

```
             estimate    se lower upper tvalue df   pvalue
(0,0,5,0),0     -1.97 0.112 -2.19 -1.74  -17.5 52 1.98e-23
```

```
contrast(m, cnames = c("Before","After"),
  a = list(Temp = 7, Insul = c("Before","After")),
  b = list(Temp = 2, Insul = c("Before","After")))
```

```
       estimate    se lower upper tvalue df   pvalue
Before    -1.97 0.112 -2.19 -1.74  -17.5 52 1.98e-23
After     -1.39 0.115 -1.62 -1.16  -12.1 52 8.94e-17
```

The function `margeff` (also from the **trtools** package) is specifically designed to estimate marginal effects (and other things) and works similarly to `contrast`.

```
margeff(m, cnames = c("Before","After"),
  a = list(Temp = 7, Insul = c("Before","After")),
  b = list(Temp = 2, Insul = c("Before","After")))
```

```
       estimate    se lower upper tvalue df   pvalue
Before    -1.97 0.112 -2.19 -1.74  -17.5 52 1.98e-23
After     -1.39 0.115 -1.62 -1.16  -12.1 52 8.94e-17
```

We can also estimate the discrete marginal effect of adding insulation at different temperatures.

```
contrast(m, cnames = c("0C","5C","10C"),
  a = list(Temp = c(0,5,10), Insul = "After"),
  b = list(Temp = c(0,5,10), Insul = "Before"))
```

```
    estimate     se lower  upper tvalue df   pvalue
0C    -2.130 0.1801 -2.49 -1.769 -11.83 52 2.32e-16
5C    -1.553 0.0878 -1.73 -1.377 -17.70 52 1.15e-23
10C   -0.977 0.1858 -1.35 -0.604  -5.26 52 2.78e-06
```

```
margeff(m, cnames = c("0C","5C","10C"),
  a = list(Temp = c(0,5,10), Insul = "After"),
  b = list(Temp = c(0,5,10), Insul = "Before"))
```

```
    estimate     se lower  upper tvalue df   pvalue
0C    -2.130 0.1801 -2.49 -1.769 -11.83 52 2.32e-16
5C    -1.553 0.0878 -1.73 -1.377 -17.70 52 1.15e-23
10C   -0.977 0.1858 -1.35 -0.604  -5.26 52 2.78e-06
```

So what's the use of `margeff`? The `contrast` and `lincon` functions can only handle *linear* functions of the model parameters. But in some cases the marginal effect is not a linear function of the model parameters. This is where the `margeff` function is useful.

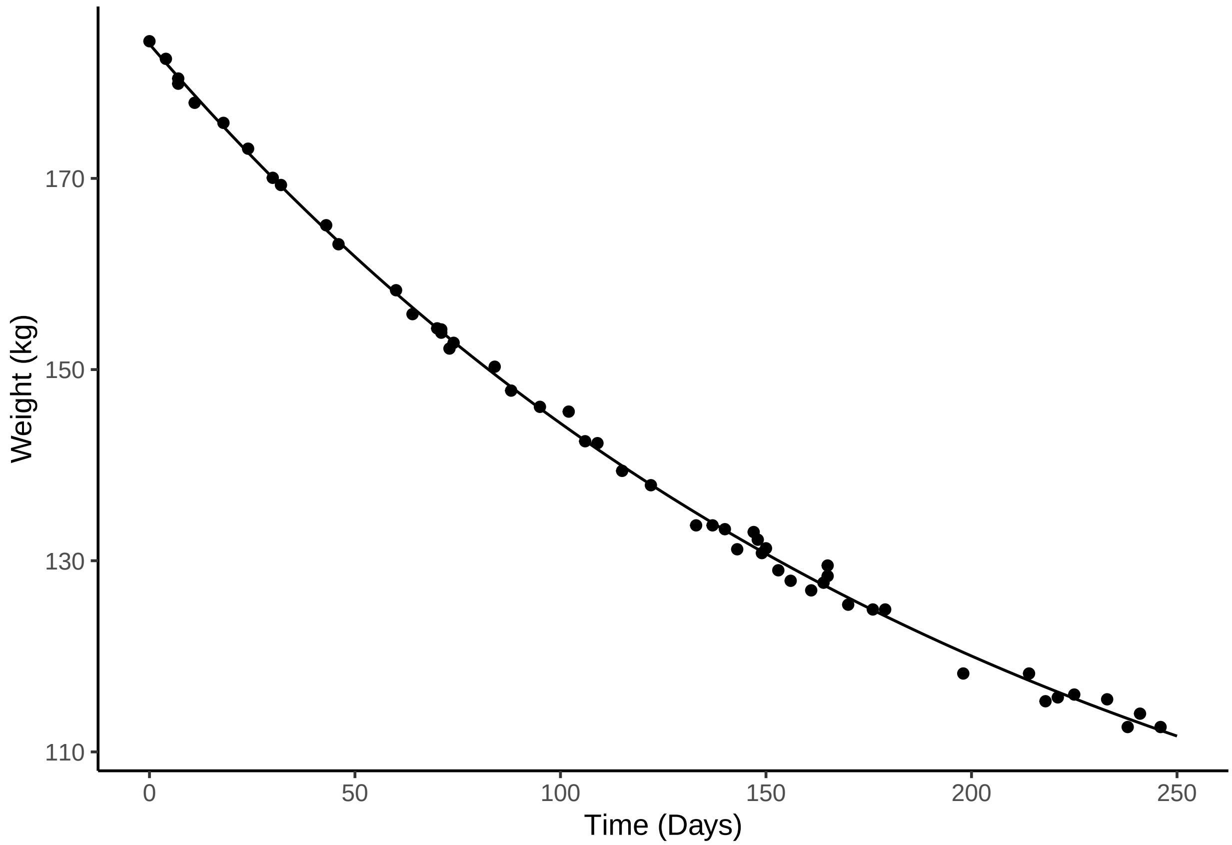
Example: Consider the following nonlinear model for the change in expected weight over time.

```
m <- nls(Weight ~ t1 + t2*2^(-Days/t3), data = MASS::wtloss,
  start = list(t1 = 90, t2 = 95, t3 = 120))

d <- data.frame(Days = seq(0, 250, by = 1))
d$yhat <- predict(m, newdata = d)

p <- ggplot(MASS::wtloss, aes(x = Days, y = Weight)) +
  geom_point() + theme_classic() +
  labs(y = "Weight (kg)", x = "Time (Days)") +
```

```
  geom_line(aes(y = yhat), data = d)
plot(p)
```



The model is

$$E(Y) = \theta_1 + \theta_2 2^{-d/\theta_3},$$

where Y is weight and d is days. The discrete marginal effect of going from 50 to 100 days is

$$\underbrace{\theta_1 + \theta_2 2^{-100/\theta_3}}_{E(Y|d=100)} - (\underbrace{\theta_1 + \theta_2 2^{-50/\theta_3}}_{E(Y|d=50)}) = \theta_2(2^{-100/\theta_3} - 2^{-50/\theta_3}).$$

This is *not* a linear function of the model parameters, so we cannot use the usual methods like `contrast` or `lincon`. But we can make (approximate) inferences using the *delta method* (more on that later). The `margeff` function makes implementing this method relatively straight forward.

```
margeff(m, a = list(Days = 100), b = list(Days = 50))
```

```
 estimate    se lower upper tvalue df   pvalue
    -17.4 0.129 -17.7 -17.2   -135 49 1.18e-64
```

```
margeff(m,
  a = list(Days = c(50,100,150,200)),
  b = list(Days = c(0,50,100,150)),
  cnames = c("0->50", "50->100", "100->150", "150->200"))
```

```
         estimate    se lower upper tvalue df   pvalue
0->50       -22.3 0.329 -22.9 -21.6  -67.6 49 4.84e-50
50->100     -17.4 0.129 -17.7 -17.2 -134.9 49 1.18e-64
100->150    -13.7 0.103 -13.9 -13.4 -132.2 49 3.11e-64
```

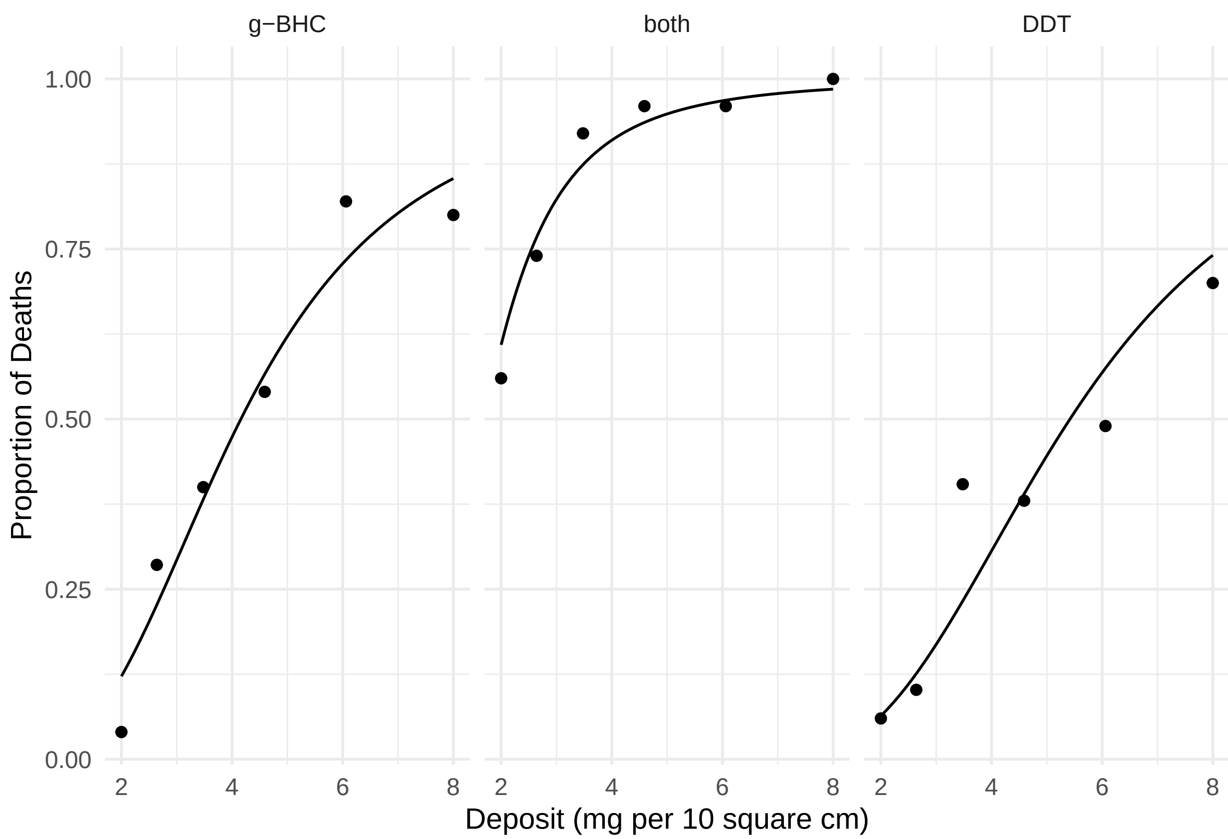
```
150->200    -10.7 0.161 -11.0 -10.4  -66.6 49 1.00e-49
```

Example: Consider the following model for the `insecticide` data.

```
m <- glm(cbind(deaths, total-deaths) ~ log(deposit) + insecticide,
  family = binomial, data = insecticide)

d <- expand.grid(deposit = seq(2, 8, length = 100),
  insecticide = unique(insecticide$insecticide))
d$phat <- predict(m, newdata = d, type = "response")

p <- ggplot(insecticide, aes(x = deposit, y = deaths/total)) +
  geom_point() + facet_wrap(~ insecticide) +
  geom_line(aes(y = phat), data = d) + theme_minimal() +
  labs(x = "Deposit (mg per 10 square cm)",
    y = "Proportion of Deaths")
plot(p)
```



We know how to interpret the effects using *odds ratios*. Here are the odds ratios for the effect of doubling the deposit from 2 to 4 units.

```
contrast(m, tf = exp,
  a = list(deposit = 4, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 2, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

```
      estimate lower upper
g-BHC     6.48  4.83  8.68
```

```
both      6.48  4.83  8.68
DDT       6.48  4.83  8.68
```

And here are the odds ratios for the effect of insecticide (g-BHC versus DDT).

```
contrast(m, tf = exp,
  a = list(deposit = c(2,4,6,8), insecticide = "g-BHC"),
  b = list(deposit = c(2,4,6,8), insecticide = "DDT"),
  cnames = c("2","4","6","8"))
```

```
  estimate lower upper
2     2.04  1.38  3.01
4     2.04  1.38  3.01
6     2.04  1.38  3.01
8     2.04  1.38  3.01
```

But with odds ratios we have to interpret effects in terms of *odds*. What if we want to interpret the effect on the *probability*? The discrete marginal effect is in terms of the *expected response* (here the expected proportion or, equivalently, the probability of death).

```
margeff(m,
  a = list(deposit = 4, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 2, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

```
      estimate     se lower upper tvalue  df   pvalue
g-BHC    0.352 0.0247 0.303 0.400  14.24 Inf 5.18e-46
both     0.301 0.0365 0.229 0.372   8.24 Inf 1.74e-16
DDT      0.242 0.0219 0.200 0.285  11.08 Inf 1.63e-28
```

Here are some discrete marginal effects of insecticide (g-BHC versus DDT).

```
margeff(m,
  a = list(deposit = c(2,4,6,8), insecticide = "g-BHC"),
  b = list(deposit = c(2,4,6,8), insecticide = "DDT"),
  cnames = c("2","4","6","8"))
```

```
  estimate     se  lower upper tvalue  df   pvalue
2   0.0582 0.0177 0.0235 0.093   3.28 Inf 0.001026
4   0.1675 0.0456 0.0781 0.257   3.67 Inf 0.000243
6   0.1603 0.0439 0.0742 0.246   3.65 Inf 0.000264
8   0.1128 0.0322 0.0495 0.176   3.50 Inf 0.000472
```

The appeal of the marginal effect here is that for many people probabilities are more intuitive than odds.

Example: Consider the following model for data from a study of the effect of blood plasma concentration/dilution on clotting time.

```
clotting <- data.frame(
  conc = rep(c(5,10,15,20,30,40,60,80,100), 2),
  time = c(118,58,42,35,27,25,21,19,18,69,35,26,21,18,16,13,12,12),
  lot = rep(c("L1","L2"), each = 9)
)
head(clotting)
```

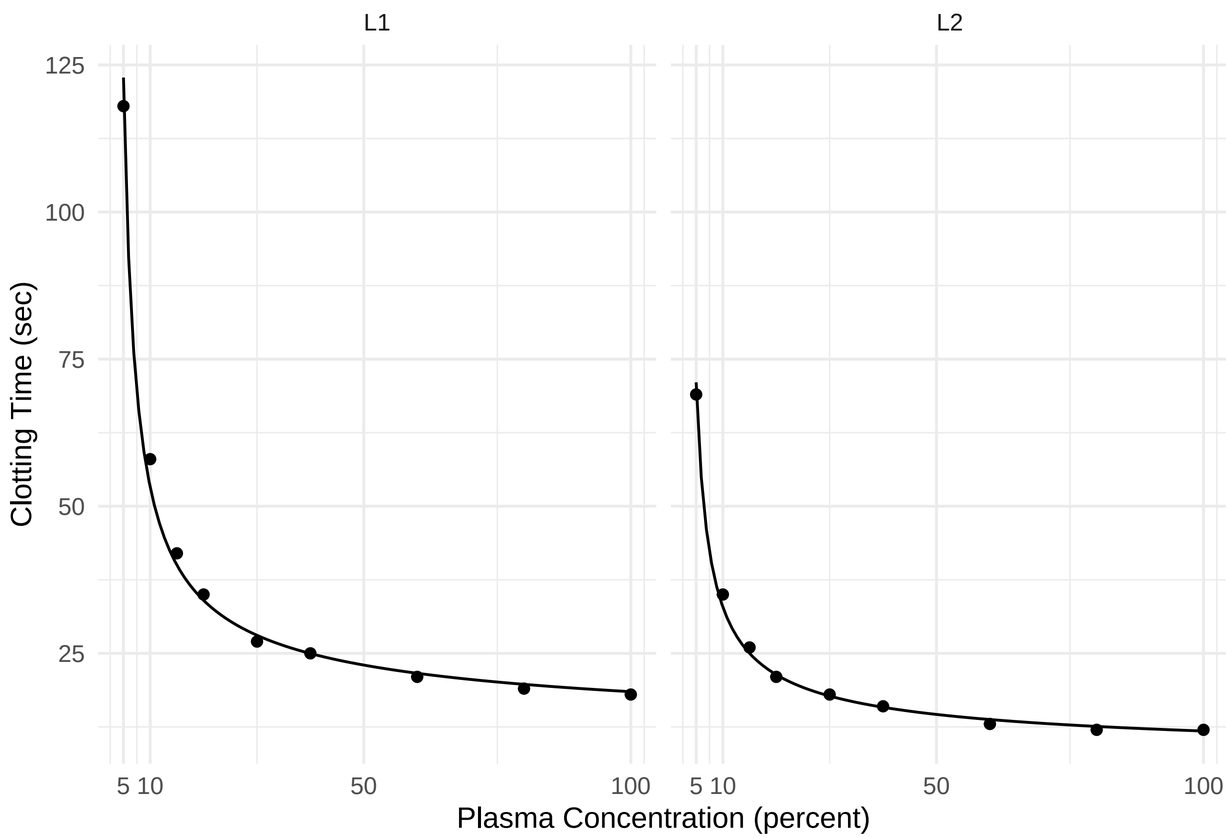
```
  conc time lot
1    5  118  L1
2   10   58  L1
3   15   42  L1
```

```
4   20   35  L1
5   30   27  L1
6   40   25  L1
```

```
m <- glm(time ~ lot + log(conc) + lot:log(conc),
  family = Gamma(link = inverse), data = clotting)

d <- expand.grid(conc = seq(5, 100, length = 100), lot = c("L1","L2"))
d$yhat <- predict(m, newdata = d, type = "response")

p <- ggplot(clotting, aes(x = conc, y = time)) + theme_minimal() +
  geom_point() + facet_wrap(~ lot) + facet_wrap(~ lot) +
  labs(x = "Plasma Concentration (percent)", y = "Clotting Time (sec)") +
  scale_x_continuous(breaks = c(5,10,50,100)) +
  geom_line(aes(y = yhat), data = d)
plot(p)
```




```
summary(m)$coefficients
```

```
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      -0.01655   0.000865  -19.13 1.97e-11
lotL2            -0.00735   0.001678   -4.38 6.25e-04
log(conc)         0.01534   0.000387   39.63 8.85e-16
lotL2:log(conc)  0.00826   0.000735   11.23 2.18e-08
```

This generalized linear model can be written as

$$\frac{1}{E(T_i)} = \beta_0 + \beta_1 l_i + \beta_2 \log_2 c_i + \beta_3 l_i \log_2 c_i,$$

or, equivalently,

$$E(T_i) = \frac{1}{\beta_0 + \beta_1 l_i + \beta_2 \log_2 c_i + \beta_3 l_i \log_2 c_i},$$

where T_i is clotting time, c_i is plasma concentration, and l_i is an indicator variable such that $l_i = 1$ if the i-th observation is from the second lot, and $l_i = 0$ otherwise.

Marginal effects of increasing the plasma concentration from 5 to 10 in each lot.

```r
margeff(m,
  a = list(conc = 10, lot = c("L1","L2")),
  b = list(conc =  5, lot = c("L1","L2")),
  cnames = c("L1,5->10","L2,5->10"))
```

```
         estimate   se lower upper tvalue df   pvalue
L1,5->10    -69.6 4.81 -79.9 -59.3  -14.5 14 8.15e-10
L2,5->10    -38.2 2.71 -44.0 -32.4  -14.1 14 1.16e-09
```

Marginal effects of increasing from 5 to 10, 10 to 50, and 50 to 100 in the first lot.

```r
margeff(m,
  a = list(conc = c(10,50,100), lot = "L1"),
  b = list(conc = c(5,10,50), lot = "L1"),
  cnames = c("L1,5->10","L1,10->50","L1,50->100"))
```

```
           estimate     se  lower  upper tvalue df   pvalue
L1,5->10     -69.60 4.8081 -79.91 -59.28  -14.5 14 8.15e-10
L1,10->50    -30.26 0.7124 -31.79 -28.73  -42.5 14 3.38e-16
L1,50->100    -4.52 0.0696  -4.67  -4.37  -65.0 14 9.06e-19
```

Marginal effects for plasma concentration for the *second* lot.

```r
margeff(m,
  a = list(conc = c(10,50,100), lot = "L2"),
  b = list(conc = c(5,10,50), lot = "L2"),
  cnames = c("L2,5->10","L2,10->50","L2,50->100"))
```

```
           estimate     se  lower  upper tvalue df   pvalue
L2,5->10     -38.20 2.7107 -44.01 -32.38  -14.1 14 1.16e-09
L2,10->50    -18.24 0.4595 -19.23 -17.26  -39.7 14 8.61e-16
L2,50->100    -2.82 0.0436  -2.91  -2.73  -64.7 14 9.61e-19
```

Marginal effects for lot at three plasma concentrations.

```r
margeff(m,
  a = list(conc = c(25,50,75), lot = c("L1")),
  b = list(conc = c(25,50,75), lot = c("L2")),
  cnames = c("25","50","75"))
```

```
   estimate    se lower upper tvalue df   pvalue
25    11.25 0.581 10.00 12.49   19.4 14 1.67e-11
50     8.39 0.481  7.36  9.42   17.4 14 6.84e-11
75     7.30 0.439  6.36  8.24   16.6 14 1.30e-10
```

"Instantaneous" Marginal Effects

Consider a regression model with (without loss of generality) two explanatory variables, X_1 and X_2. Assuming that X_1 is *continuous*, the "instantaneous" marginal effect of X_1 at x_1 when $X_2 = x_2$ is

$$\lim_{\delta \to 0} \frac{E(Y|X_1 = x_1 + \delta, X_2 = x_2) - E(Y|X_1 = x_1, X_2 = x_2)}{\delta}.$$

This can also be written as

$$\left.\frac{\partial E(Y|X_1 = z, X_2 = x_2)}{\partial z}\right|_{z=x_1}$$

i.e., the partial derivative of $E(Y|X_1 = x_1, X_2 = x_2)$ with respect to and evaluated at x_1.

Intuitively, this is the rate of change in the expected response at a specific value of the explanatory variable — i.e., the slope of the function at a specific point.

To compute this marginal effect we can either find the partial derivative analytically or approximate it numerically using

$$\frac{E(Y|X_1 = x_1 + \delta, X_2 = x_2) - E(Y|X_1 = x_1, X_2 = x_2)}{\delta}$$

where δ set to a small value relative to x_1 (this is called *numerical differentiation*).

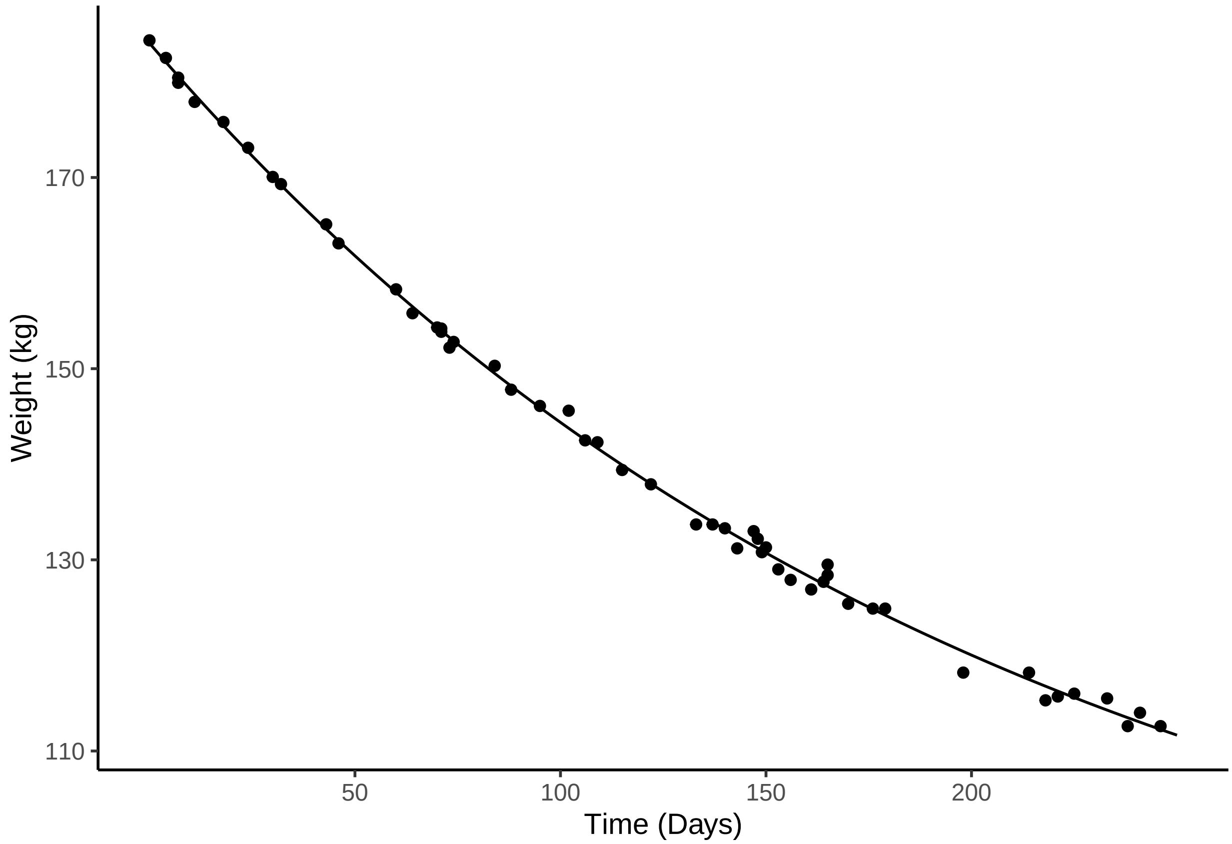
Note that instantaneous marginal effects are only defined for *continuous quantitative variables*.

Example: Consider again the nonlinear regression model for expected weight as a function of days.

```
m <- nls(Weight ~ t1 + t2*2^(-Days/t3), data = MASS::wtloss,
  start = list(t1 = 90, t2 = 95, t3 = 120))

d <- data.frame(Days = seq(0, 250, by = 1))
d$yhat <- predict(m, newdata = d)

p <- ggplot(MASS::wtloss, aes(x = Days, y = Weight)) +
  geom_point() + theme_classic() +
  labs(y = "Weight (kg)", x = "Time (Days)") +
  geom_line(aes(y = yhat), data = d) +
  scale_x_continuous(breaks = c(50,100,150,200))
plot(p)
```



We can estimate the instantaneous marginal effects at 50, 100, 150, and 200 days.

```
margeff(m, delta = 0.001,
  a = list(Days = c(50,100,150,200) + 0.001),
  b = list(Days = c(50,100,150,200)),
  cnames = c("@50", "@100", "@150", "@200"))
```

```
     estimate      se  lower  upper tvalue df   pvalue
@50    -0.393 0.00417 -0.401 -0.384  -94.1 49 4.93e-57
@100   -0.308 0.00183 -0.311 -0.304 -168.0 49 2.53e-69
@150   -0.241 0.00268 -0.246 -0.236  -89.8 49 4.94e-56
@200   -0.189 0.00367 -0.196 -0.181  -51.4 49 2.76e-44
```

Note: To estimate an instantaneous marginal effect, add a relatively small value of δ to the `a` variable, and also specify this amount to the `delta` argument.

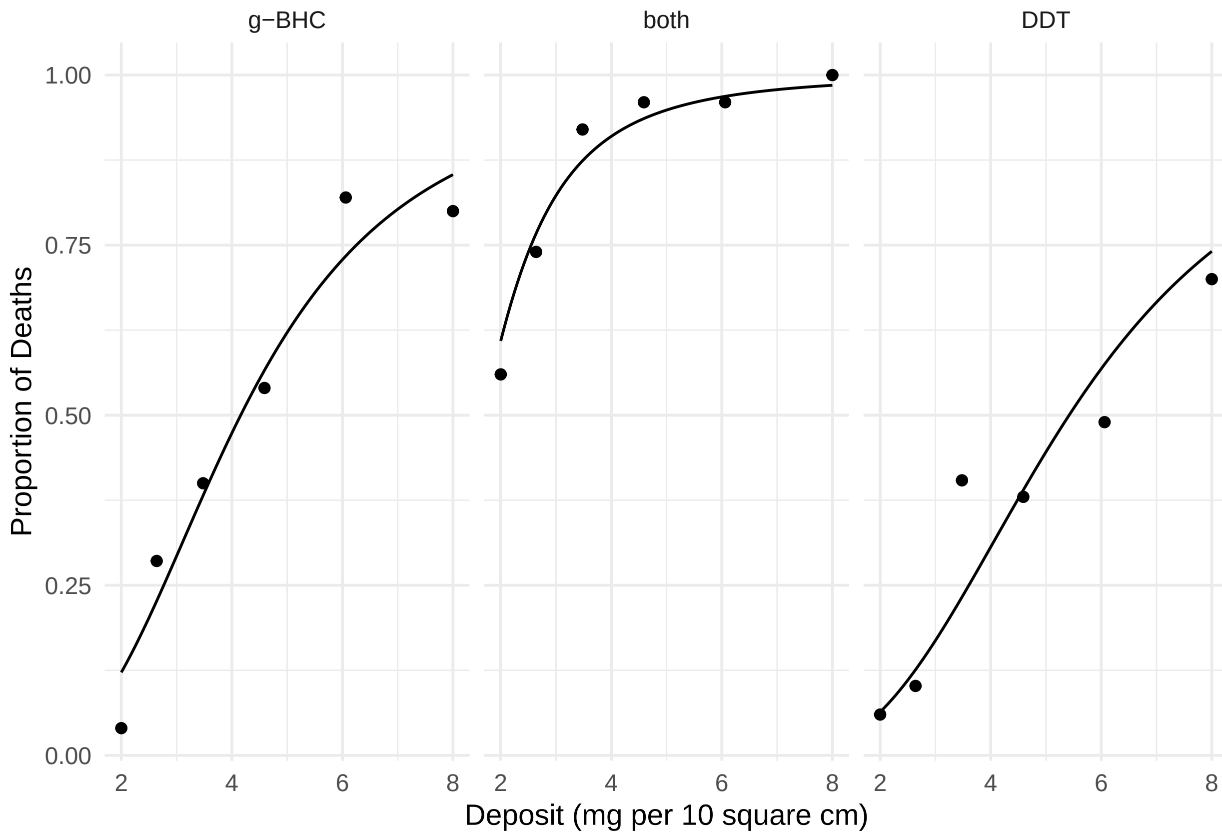
Example: Consider again the model for the `insecticide` data.

```
m <- glm(cbind(deaths, total-deaths) ~ log(deposit)
 + insecticide, family = binomial, data = insecticide)

d <- expand.grid(deposit = seq(2, 8, length = 100),
  insecticide = unique(insecticide$insecticide))
d$phat <- predict(m, newdata = d, type = "response")

p <- ggplot(insecticide, aes(x = deposit, y = deaths/total)) +
  geom_point() + facet_wrap(~ insecticide) +
  geom_line(aes(y = phat), data = d) + theme_minimal() +
```

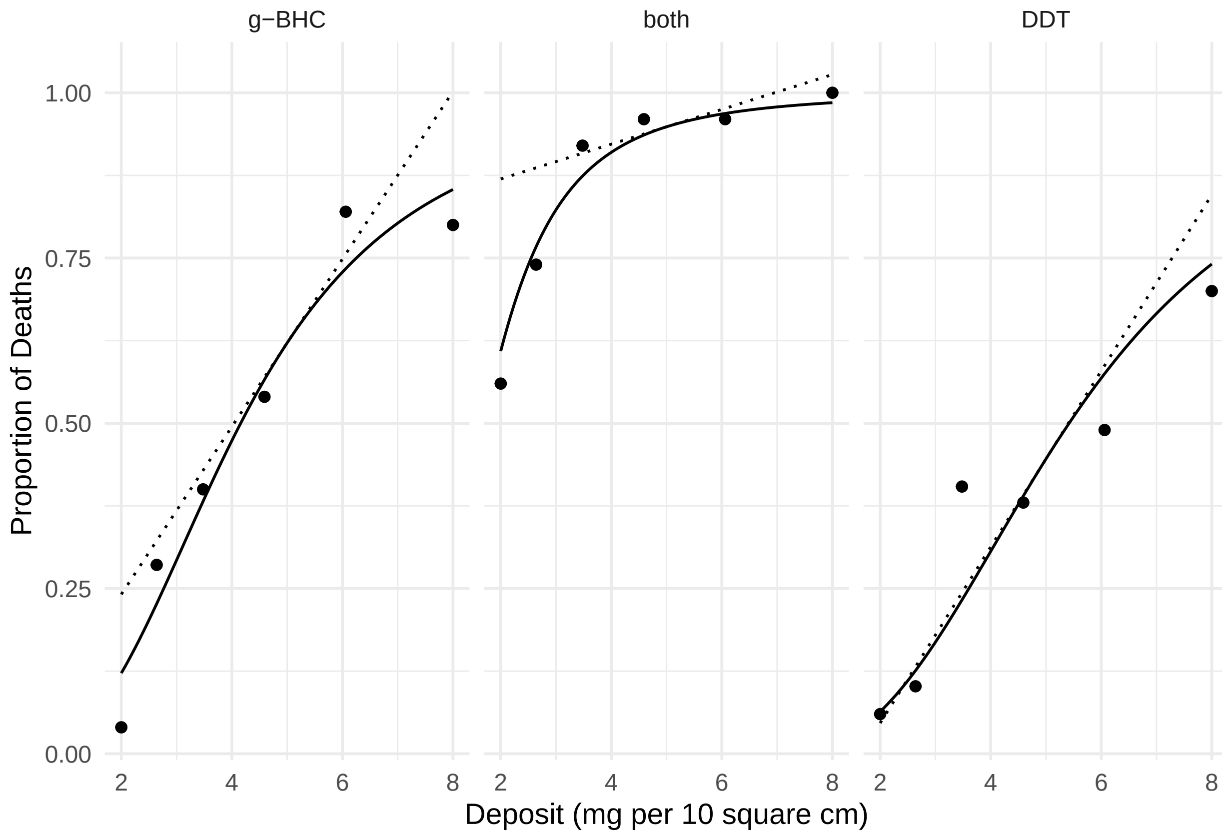
```r
  labs(x = "Deposit (mg per 10 square cm)", y = "Proportion of Deaths")
plot(p)
```



We can estimate the instantaneous marginal effect of deposit at a given amount of deposit, say 5 mg per 10 square cm.

```r
margeff(m, delta = 0.001,
  a = list(deposit = 5 + 0.001, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 5, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

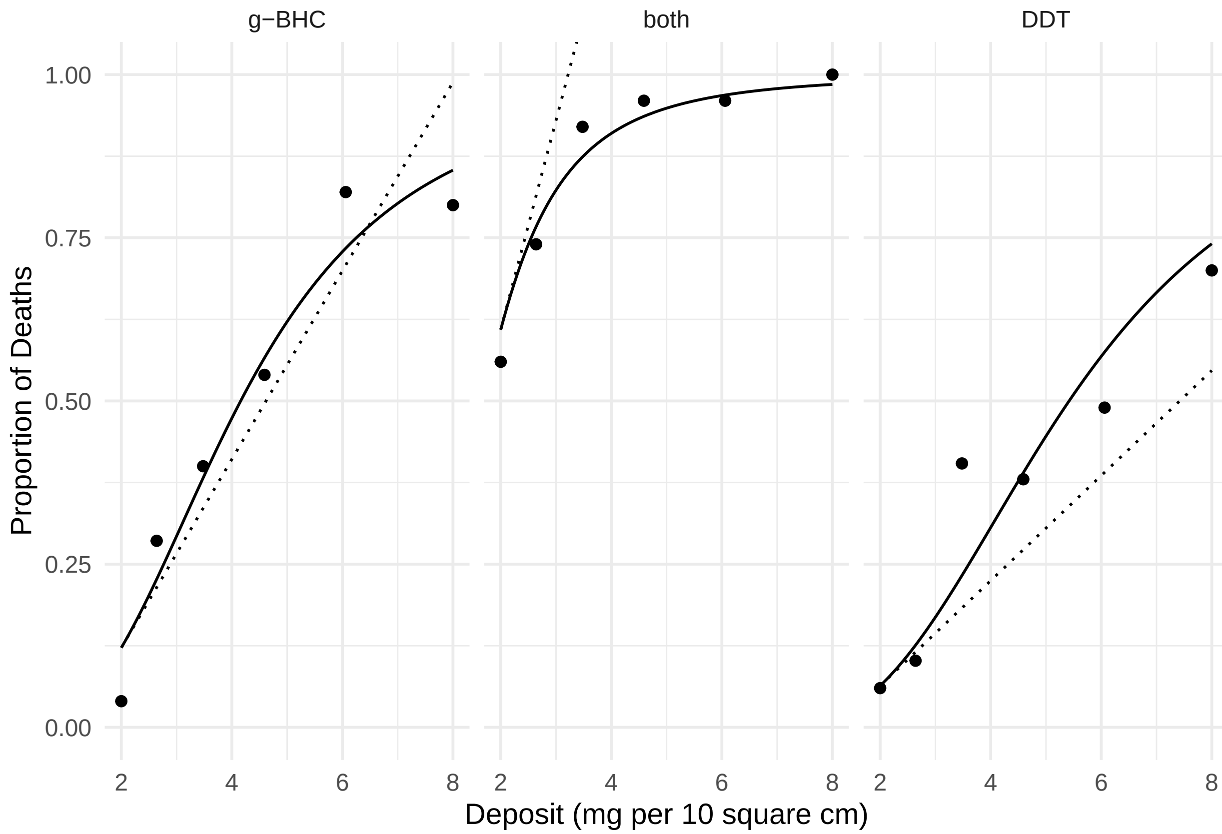
```
      estimate      se  lower  upper tvalue  df   pvalue
g-BHC   0.1268 0.00975 0.1077 0.1459  13.00 Inf 1.15e-38
both    0.0263 0.00428 0.0179 0.0347   6.15 Inf 7.94e-10
DDT     0.1332 0.01106 0.1115 0.1549  12.05 Inf 1.98e-33
```



Note that the instantaneous effect of deposit *depends on the deposit* because the probability is not a linear function of deposit.

```
margeff(m, delta = 0.001,
  a = list(deposit = 2 + 0.001, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 2, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

```
      estimate     se  lower upper tvalue  df   pvalue
g-BHC   0.1444 0.0157 0.1135 0.175   9.17 Inf 4.84e-20
both    0.3208 0.0332 0.2557 0.386   9.65 Inf 4.75e-22
DDT     0.0805 0.0118 0.0574 0.104   6.82 Inf 8.88e-12
```



Instantaneous Marginal Effects for Generalized Linear Models

Recall that in a GLM that $E(Y) = g^{-1}(\eta)$ where $\eta = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$. Consider a GLM where $\eta_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}$. The instantaneous marginal effect of X_1 at x_1 is

$$\frac{\partial E(Y|X_1 = x_1, X_2 = x_2)}{\partial x_1} = \frac{\partial g^{-1}(\eta)}{\partial x_1} = \frac{\partial g^{-1}(\eta)}{\partial \eta}\frac{\partial \eta}{\partial x_1} = \frac{\partial g^{-1}(\eta)}{\partial \eta}\beta_1$$

by the "chain rule" for (partial) derivatives.

Suppose that $E(Y) = e^{\eta}$ (i.e., log link function) where $\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2$. Then

$$\frac{\partial g^{-1}(\eta)}{\partial \eta}\beta_1 = \frac{\partial e^{\eta}}{\partial \eta}\beta_1 = e^{\eta}\beta_1 = E(Y)\beta_1.$$

Suppose now that $E(Y) = e^{\eta}/(1 + e^{\eta})$ (i.e., logit link function). Then

$$\frac{\partial g^{-1}(\eta)}{\partial \eta}\beta_1 = \frac{\partial e^{\eta}/(1 + e^{\eta})}{\partial \eta}\beta_1 = \frac{e^{\eta}}{(1 + e^{\eta})^2}\beta_1 = E(Y)[1 - E(Y)]\beta_1.$$

Suppose now that $E(Y) = \eta$ (e.g., identity link function). Then

$$\frac{\partial g^{-1}(\eta)}{\partial \eta}\beta_1 = \frac{\partial \eta}{\partial \eta}\beta_1 = \beta_1.$$

Things get a little more complicated if X_1 is a *transformed* explanatory variable or represents an interaction.

Suppose $E(Y) = \beta_0 + \beta_1 \log(x_1) + \beta_2 x_2$. Then

$$\frac{\partial g^{-1}(\eta)}{\partial \eta}\frac{\partial \eta}{\partial x_1} = \frac{\partial \eta}{\partial x_1} = \beta_1/x_1.$$

Suppose $E(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2$. Then

$$\frac{\partial g^{-1}(\eta)}{\partial \eta}\frac{\partial \eta}{\partial x_1} = \frac{\partial \eta}{\partial x_1} = \beta_1 + 2\beta_2 x_1.$$

Suppose $E(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$. Then

$$\frac{\partial g^{-1}(\eta)}{\partial \eta}\frac{\partial \eta}{\partial x_1} = \frac{\partial \eta}{\partial x_1} = \beta_1 + \beta_3 x_2.$$

Fortunately, `margeff` does the calculus!

Discrete Marginal Effects as Percent Change

Consider a regression model with (without loss of generality) two explanatory variables, X_1 and X_2. The *percent change* in the expected response when changing X_1 from x_b to x_a when $X_2 = x_2$ is

$$\frac{E(Y|X_1 = x_a, X_2 = x_2) - E(Y|X_1 = x_b, X_2 = x_2)}{E(Y|X_1 = x_b, X_2 = x_2)} \times 100\%.$$

or, equivalently,

$$\left[\frac{E(Y|X_1 = x_a, X_2 = x_2)}{E(Y|X_1 = x_b, X_2 = x_2)} - 1\right] \times 100\%.$$

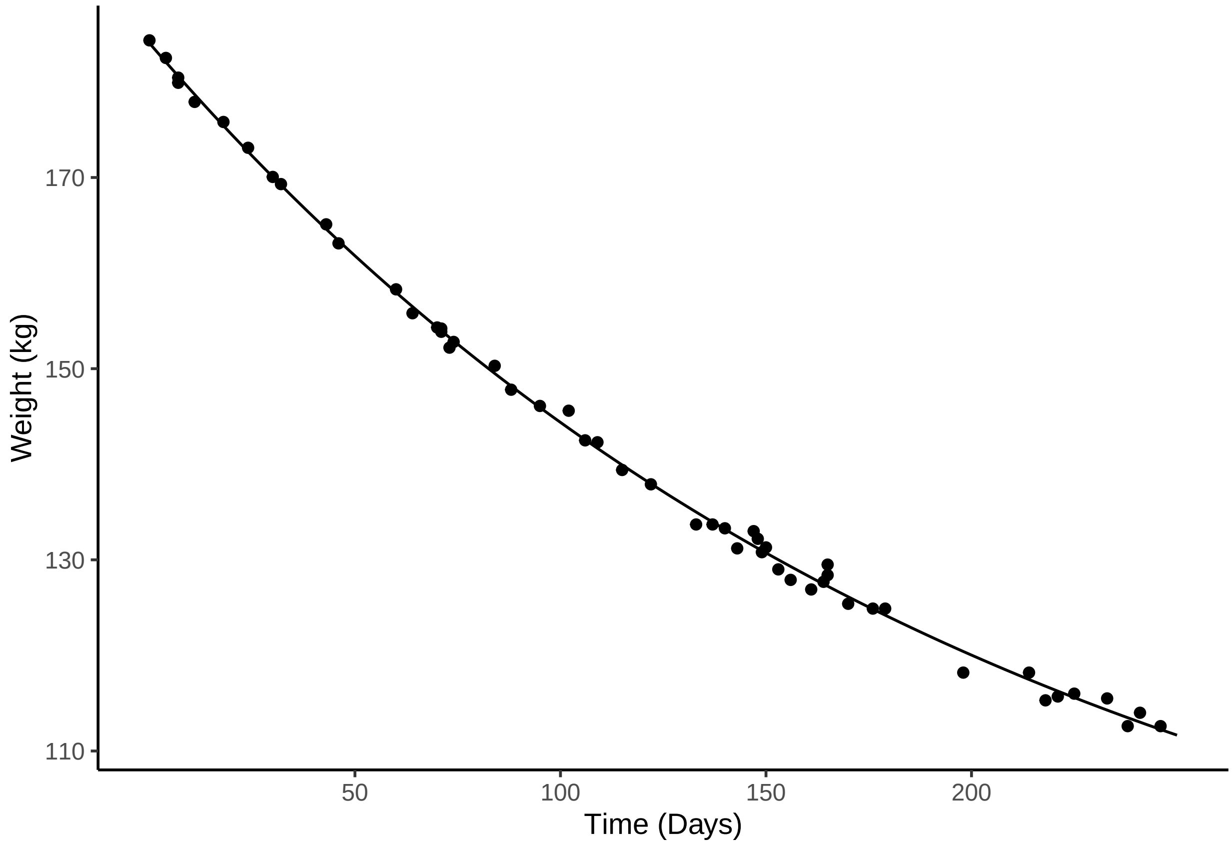
Note that the *sign* indicates if it is a percent increase or decrease.

Example: Consider again the weight loss model.

```
m <- nls(Weight ~ t1 + t2*2^(-Days/t3), data = MASS::wtloss,
  start = list(t1 = 90, t2 = 95, t3 = 120))

d <- data.frame(Days = seq(0, 250, by = 1))
d$yhat <- predict(m, newdata = d)

p <- ggplot(MASS::wtloss, aes(x = Days, y = Weight)) +
  geom_point() + theme_classic() +
  labs(y = "Weight (kg)", x = "Time (Days)") +
  geom_line(aes(y = yhat), data = d) +
  scale_x_continuous(breaks = c(50,100,150,200))
plot(p)
```



Consider the percent change in expected weight from 50 to 100 days. This is

$$\frac{\theta_1 + \theta_2 2^{-100/\theta_3} - \theta_1 - \theta_2 2^{-50/\theta_3}}{\theta_1 + \theta_2 2^{-50/\theta_3}} = \frac{\theta_2 2^{-100/\theta_3} - \theta_2 2^{-50/\theta_3}}{\theta_1 + \theta_2 2^{-50/\theta_3}}.$$

We can estimate the percent change in expected weight from 50 to 100 days as follows.

```
margeff(m, a = list(Days = 100), b = list(Days = 50), type = "percent")
```

```
 estimate     se lower upper tvalue df   pvalue
    -10.8 0.0767 -10.9 -10.6   -140 49 1.67e-65
```

We can do it for several 50 day increments as well.

```
margeff(m, type = "percent",
  a = list(Days = c(50,100,150,200)),
  b = list(Days = c(0,50,100,150)),
  cnames = c("0->50", "50->100", "100->150", "150->200"))
```

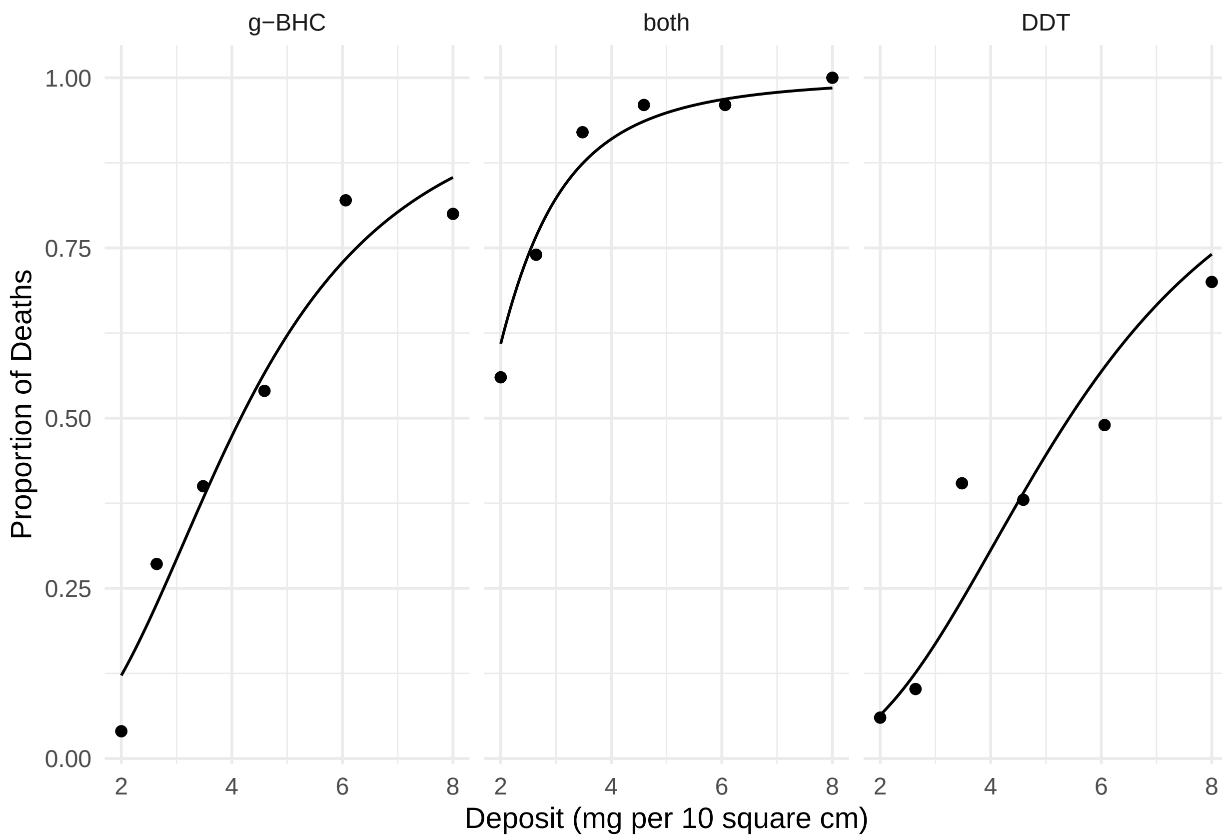
```
         estimate     se  lower  upper tvalue df   pvalue
0->50      -12.09 0.1579 -12.41 -11.77  -76.5 49 1.17e-52
50->100    -10.77 0.0767 -10.93 -10.62 -140.4 49 1.67e-65
100->150    -9.46 0.0663  -9.59  -9.32 -142.7 49 7.49e-66
150->200    -8.18 0.1209  -8.42  -7.94  -67.7 49 4.66e-50
```

Example: Consider again the model for the `insecticide` data.

```
m <- glm(cbind(deaths, total-deaths) ~ log(deposit) + insecticide,
  family = binomial, data = insecticide)
```

```r
d <- expand.grid(deposit = seq(2, 8, length = 100),
  insecticide = levels(insecticide$insecticide))
d$phat <- predict(m, newdata = d, type = "response")

p <- ggplot(insecticide, aes(x = deposit, y = deaths/total)) +
  geom_point() + facet_wrap(~ insecticide) +
  geom_line(aes(y = phat), data = d) + theme_minimal() +
  labs(x = "Deposit (mg per 10 square cm)", y = "Proportion of Deaths")
plot(p)
```



We can estimate the percent change in the probability of death from 4 to 6 mg per 10 square cm.

```r
margeff(m, type = "percent",
  a = list(deposit = 6, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 4, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

```
      estimate    se lower  upper tvalue  df   pvalue
g-BHC    53.82  6.57  41.0  66.69   8.20 Inf 2.49e-16
both      6.37  1.11   4.2   8.55   5.74 Inf 9.60e-09
DDT      85.62 11.03  64.0 107.24   7.76 Inf 8.39e-15
```

Note that here the percent change depends on where we make the increment.

```r
margeff(m, type = "percent",
  a = list(deposit = 8, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 6, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

```
      estimate    se lower upper tvalue  df   pvalue
g-BHC    17.15 1.997 13.24 21.07   8.59 Inf 8.80e-18
both      1.76 0.363  1.05  2.47   4.87 Inf 1.14e-06
DDT      30.36 3.294 23.91 36.82   9.22 Inf 3.02e-20
```

We can also estimate the percent change in the probability of death between two insecticides.

```r
margeff(m, type = "percent",
  a = list(deposit = c(2,4,6,8), insecticide = "g-BHC"),
  b = list(deposit = c(2,4,6,8), insecticide = "DDT"),
  cnames = c("2","4","6","8"))
```

```
  estimate    se lower upper tvalue  df   pvalue
2     91.3 34.75 23.17 159.4   2.63 Inf 0.00862
4     54.7 19.04 17.41  92.0   2.87 Inf 0.00405
6     28.2  9.13 10.31  46.1   3.09 Inf 0.00200
8     15.2  4.90  5.61  24.8   3.10 Inf 0.00191
```

Discrete Marginal Effects as Multiplicative Factors

Consider a regression model with (without loss of generality) two explanatory variables, X_1 and X_2. A multiplicative factor to describe the effect of changing X_1 from x_b to x_a when $X_2 = x_2$ is

$$f = \frac{E(Y|X_1 = x_a, X_2 = x_2)}{E(Y|X_1 = x_b, X_2 = x_2)},$$

meaning that

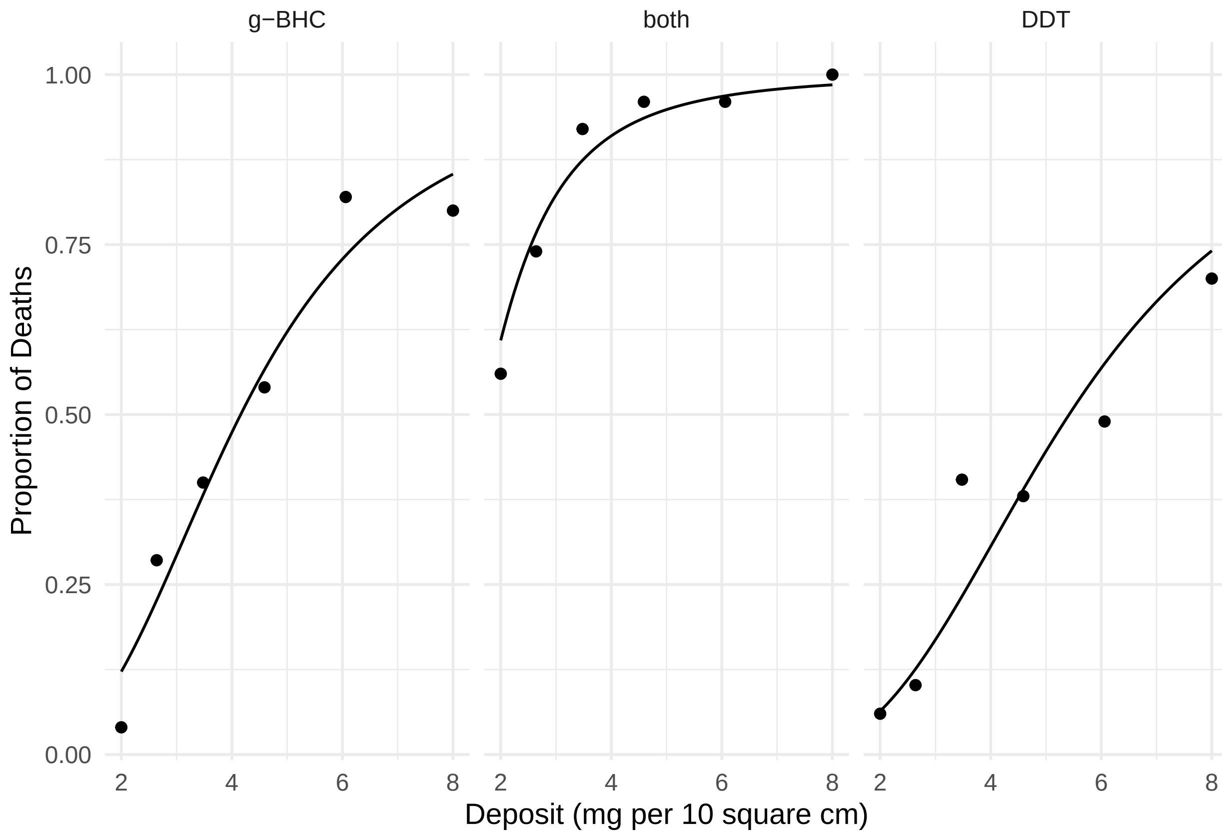
$$E(Y|X_1 = x_a, X_2 = x_2) = f \times E(Y|X_1 = x_b, X_2 = x_2).$$

Example: Consider again the model for the `insecticide` data.

```r
m <- glm(cbind(deaths, total-deaths) ~ log(deposit) + insecticide,
  family = binomial, data = insecticide)

d <- expand.grid(deposit = seq(2, 8, length = 100),
  insecticide = levels(insecticide$insecticide))
d$phat <- predict(m, newdata = d, type = "response")

p <- ggplot(insecticide, aes(x = deposit, y = deaths/total)) +
  geom_point() + facet_wrap(~ insecticide) +
  geom_line(aes(y = phat), data = d) + theme_minimal() +
  labs(x = "Deposit (mg per 10 square cm)", y = "Proportion of Deaths")
plot(p)
```



We can estimate the factor by which we increase probability by increasing deposit from 4 to 6 mg per 10 square cm.

```
margeff(m, type = "factor",
  a = list(deposit = 6, insecticide = c("g-BHC","both","DDT")),
  b = list(deposit = 4, insecticide = c("g-BHC","both","DDT")),
  cnames = c("g-BHC","both","DDT"))
```

```
      estimate     se lower upper tvalue  df    pvalue
g-BHC     1.54 0.0657  1.41  1.67   23.4 Inf 2.44e-121
both      1.06 0.0111  1.04  1.09   95.8 Inf  0.00e+00
DDT       1.86 0.1103  1.64  2.07   16.8 Inf  1.56e-63
```

We can also estimate the factor for comparing both insecticides with g-BHC only.

```
margeff(m, type = "factor",
  a = list(deposit = c(2,4,6,8), insecticide = "both"),
  b = list(deposit = c(2,4,6,8), insecticide = "g-BHC"),
  cnames = c("2","4","6","8"))
```

```
  estimate     se lower upper tvalue  df    pvalue
2     4.99 0.8719  3.29  6.70   5.73 Inf  1.02e-08
4     1.92 0.1420  1.64  2.20  13.52 Inf  1.14e-41
6     1.33 0.0546  1.22  1.44  24.31 Inf 1.47e-130
8     1.15 0.0310  1.09  1.21  37.24 Inf 1.44e-303
```

Using Different Kinds of Marginal Effects

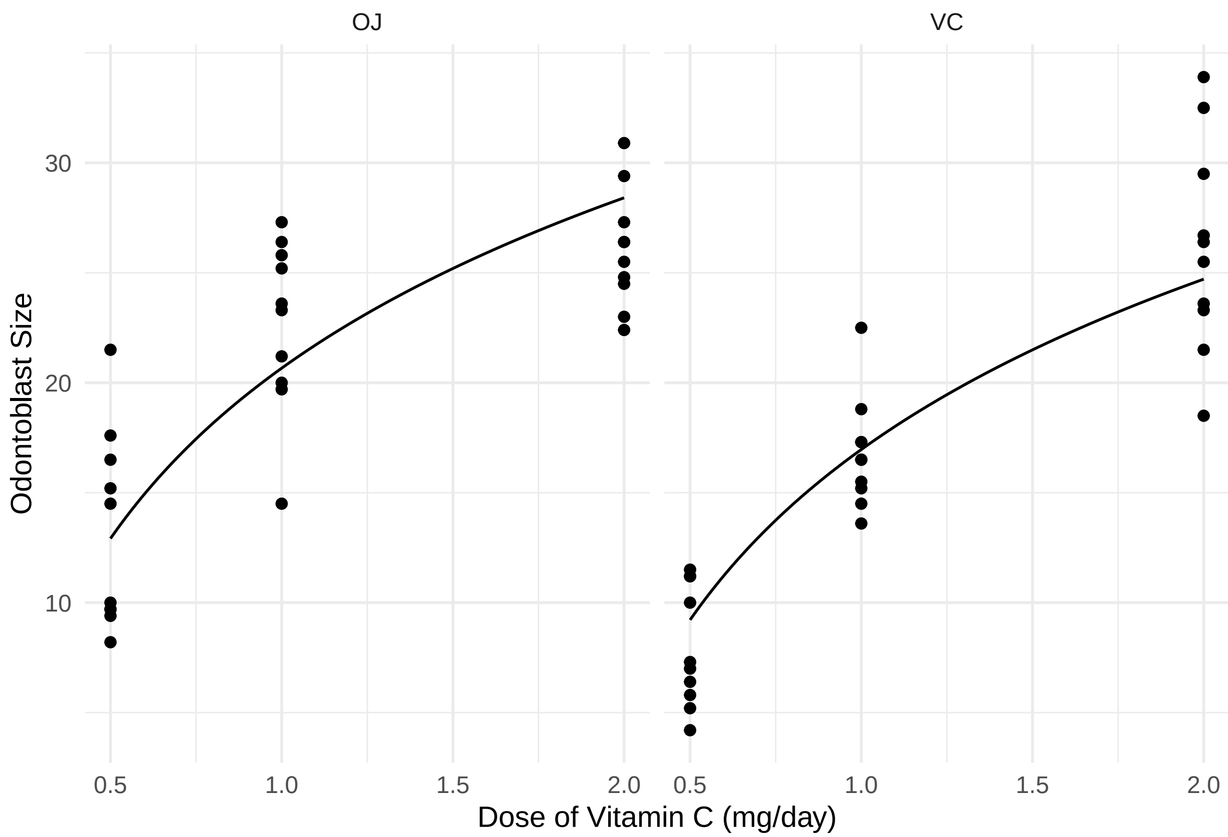
Marginal effects give us a variety of ways to summarize the statistical relationship between a response variable and an explanatory variable.

Example: Consider the following model for the `ToothGrowth` data.

```
m <- lm(len ~ log(dose) + supp, data = ToothGrowth)

d <- expand.grid(dose = seq(0.5, 2, length = 100), supp = c("OJ","VC"))
d$yhat <- predict(m, d)

p <- ggplot(ToothGrowth, aes(x = dose, y = len)) +
  geom_point() + facet_wrap(~ supp) +
  geom_line(aes(y = yhat), data = d) +
  labs(x = "Dose of Vitamin C (mg/day)", y = "Odontoblast Size") +
  theme_minimal()
plot(p)
```



We can use discrete marginal effects, such as when increasing dose from 0.5 to 1 mg/day.

```
margeff(m, cnames = c("OJ","VC"),
  a = list(dose = 1.0, supp = c("OJ","VC")),
  b = list(dose = 0.5, supp = c("OJ","VC")))
```

```
   estimate    se lower upper tvalue df   pvalue
OJ     7.75 0.609  6.53  8.97   12.7 57 2.74e-18
VC     7.75 0.609  6.53  8.97   12.7 57 2.74e-18
```

We can use instantaneous effects, such as the instantaneous effect at 1 mg/day.

```
margeff(m, cnames = c("OJ","VC"), delta = 0.001,
  a = list(dose = 1 + 0.001, supp = c("OJ","VC")),
  b = list(dose = 1, supp = c("OJ","VC")))
```

```
   estimate    se lower upper tvalue df   pvalue
OJ     11.2 0.878  9.41  12.9   12.7 57 2.74e-18
VC     11.2 0.878  9.41  12.9   12.7 57 2.74e-18
```

We can use the percent change, such as when increasing dose from 0.5 to 1 mg/day.

```
margeff(m, cnames = c("OJ","VC"), type = "percent",
  a = list(dose = 1.0, supp = c("OJ","VC")),
  b = list(dose = 0.5, supp = c("OJ","VC")))
```

```
   estimate    se lower upper tvalue df   pvalue
OJ     60.0  8.22  43.5  76.4   7.30 57 1.02e-09
VC     84.1 13.75  56.5 111.6   6.11 57 9.41e-08
```

We can use a multiplicative factor, such as when increasing dose form 0.5 to 1 mg/day.

```
margeff(m, cnames = c("OJ","VC"), type = "factor",
  a = list(dose = 1.0, supp = c("OJ","VC")),
  b = list(dose = 0.5, supp = c("OJ","VC")))
```

```
   estimate     se lower upper tvalue df   pvalue
OJ     1.60 0.0822  1.44  1.76   19.5 57 7.56e-27
VC     1.84 0.1375  1.57  2.12   13.4 57 3.08e-19
```

Note: There are functions in other packages for estimating some kinds of marginal effects (e.g., see the package **marginaleffects**).