Modèle linéaire généralisé mise en œuvre des exemples de Collett[1] avec SAS et Splus

Chabanet Claire
INRA - Laboratoire de Recherches sur les Arômes
email : chabanet@dijon.inra.fr

Dessaint Fabrice
INRA - Malherbologie
email : dessaint@dijon.inra.fr

BV1540, F-21034 DIJON cedex, FRANCE

10 décembre 1998

1. Modelling Binary Data, Collett, Chapman & Hall, 1991

L'ouvrage de Collett "Modelling Binary Data" Chapman & Hall (1991) a été étudié dans le cadre d'un groupe de travail INRA qui a fonctionné sur le centre de Dijon, sur le thème du modèle linéaire généralisé, en 1997. Dans le cadre de ce groupe de travail, les chapitres 1 à 5 ont été étudiés, et la mise en œuvre des analyses statistiques a été réalisée à l'aide des logiciels SAS (version 6.11) et Splus (version 3.4).

Les enchaînements de commandes SAS et Splus sont regroupés et commentés dans ce document, de même que les résultats numériques et graphiques obtenus, la plupart d'entre eux ayant été présentés par Collett. Ce document n'a pas d'autre vocation que de servir d'exemple de mise en œuvre. Il pourra être utilisé lors des formations au modèle linéaire généralisé dispensées par l'INRA. A ce titre, ce document est intégré à la valise du formateur (FPSTAT2, module 10, Modèle linéaire généralisé), et sera mis à la disposition des stagiaires qui le souhaiteront.

Pour plus de détail, le lecteur se réfèrera à l'ouvrage de référence :
Modelling Binary Data, Collett, Chapman & Hall, 1991.

Première partie

MISE EN ŒUVRE AVEC SAS

Chapitre 1

Introduction

1.1 Some examples

Exemple `Souris` *: (1).* Des souris sont placées dans des cages dans lesquelles on injecte (groupe `1-Cigarette`) ou non (groupe `0-Controle`) de la fumée de cigarette. Après un certain temps, on sacrifie les animaux et on observe ou non le développement d'une tumeur.

(1) ***PGM***

```
* Exemple 1.1 : Smoked mice ;

Data Souris;
 format Group $12.;
 input Tumeur$ Group y n ;
cards;
Oui     1-Cigarette     21 23
Oui     0-Controle      19 32
Non     1-Cigarette      2 23
Non     0-Controle      13 32
;
run;
```

PGM

Exemple `Prune` *: (2).* On cherche à comparer la proportion de plants qui "prennent" ou pas selon qu'ils sont plantés immédiatement (`1-pre`) ou au printemps (`0-tar`) et selon la taille des racines (courtes : `1-cou` ou longues : `0-lon`).

(2) ***PGM***

```
* Exemple 1.2 : plum root-stocks;

Data Prune;
 retain n 240;
 input longueur$ periode$ vivant;
 mort=n-vivant;
cards;
1-cou      1-pre       107
1-cou      0-tar       31
0-lon      1-pre       156
0-lon      0-tar       84
;
run;
```

PGM

Exemple (3)

(3) ***PGM***

```
* Exemple 1.3 : Germination of Orobranche;

Data Exe3;
```

```
 input sp$ racine$ y n;
cards;
o75    b       10      39
o75    b       23      62
o75    b       23      81
o75    b       26      51
o75    b       17      39
o75    c       5       6
o75    c       53      74
o75    c       55      72
o75    c       32      51
o75    c       46      79
o75    c       10      13
o73    b       8       16
o73    b       10      30
o73    b       8       28
o73    b       23      45
o73    b       0       4
o73    c       3       12
o73    c       22      41
o73    c       15      30
o73    c       32      51
o73    c       3       7
;
run;
```
PGM

Exemple **Rats** *: (4)*: On observe la proportion de rats survivants dans des portées, les rattes ayant été affectées à deux groupes dont les régimes alimentaires diffèrent.

(4) ***PGM***

```
* Exemple 1.4 : Rats;

Data Rats;
 do treat='traite', 'control';
    do i=1 to 16;
       input y n @@;
       output;
    end;
 end;
drop i;
cards;
13 13  12 12  9 9    9 9    8 8    8 8    12 13  11 12
 9 10   9 10  8 9    11 13  4 5    5 7    7 10   7 10
12 12  11 11  10 10  9 9    10 11  9 10   9 10   8 9
8 9    4 5    7 9    4 7    5 10   3 6    3 10   0 7
;
run;
```
PGM

Exemple **Aircraft** *: (5)*. Dans cet exemple, on teste la charge maximale que peuvent supporter des rivets en alliage utilisés dans l'aviation. On teste des lots de rivets avec des charges croissantes.

(5) ***PGM***

```
* Exemple 1.5 : Aircraft fasteners;

Data aircraft;
 do load=2500 to 4300 by 200;
    input k n @@;
    p=k/n;
    output;
```

```
 end;
cards;
10 50    17 70    30 100   21 60     18 40
43 85    54 90    33  50   60 80     51 65
;
run;
```
PGM

Exemple **`Serum`** *: (6).* Cinq doses d'un serum anti-pneumocoque sont expérimentées. Pour chaque dose, on utilise un lot de 40 souris. Les souris succombent ou pas.

(6) ***PGM***
```
* Exemple 1.6 : Anti-pneumococcus serum;

Data serum;
 input dose y n;
 logdose=log(dose);
 p=y/n;
cards;
.0025  35      40
.0056  21      40
.0112  9       40
.0225  6       40
.0450  1       40
;
run;
```
PGM

Exemple **`ESR`** *: (7).* Cet exemple concerne des données relatives au taux de sédimentation de globules rouges. Un taux de sédimentation inférieur à un certain seuil est un signe de bonne santé (Y=1), alors qu'un taux supérieur au même seuil est un indicateur d'un problème (Y=0).

(7) ***PGM***
```
* Exemple 1.7 : Determination of ESR;

Data ESR;
 do ind=1 to 32;
    input Fibri gammaG esr @@;
    output;
 end;
cards;
2.52 38 0     2.56 31 0      2.19 33 0      2.18 31 0      3.41 37 0
2.46 36 0     3.22 38 0      2.21 37 0      3.15 39 0      2.60 41 0
2.29 36 0     2.35 29 0      5.06 37 1      3.34 32 1      2.38 37 1
3.15 36 0     3.53 46 1      2.68 34 0      2.60 38 0      2.23 37 0
2.88 30 0     2.65 46 0      2.09 44 1      2.28 36 0      2.67 39 0
2.29 31 0     2.15 31 0      2.54 28 0      3.93 32 1      3.34 30 0
2.99 36 0     3.32 35 0
;
run;
```
PGM

Exemple **`Role`** *: (8).* L'exemple concerne le du rôle de la femme dans la société, ou plus précisément de l'attitude vis à vis de la femme au foyer. Ces données sont issues d'enquêtes d'opinion.

(8) ***PGM***
```
* Exemple 1.8 : Women's role in society;

Data Role;
 do educ=0 to 20 by 1;
    do sexe='hommes', 'femmes';
       input accord desacc @@;
       n=accord+desacc;
```

```
         output;
      end;
   end;
cards;
4       2       4       2
2       0       1       0
4       0       0       0
6       3       6       1
5       5       10      0
13      7       14      7
25      9       17      5
27      15      26      16
75      49      91      36
29      29      30      35
32      45      55      67
36      59      50      62
115     245     190     403
31      70      17      92
28      79      18      81
9       23      7       34
15      110     13      115
3       29      3       28
1       28      0       21
2       13      1       2
3       20      2       4
;
run;
```

PGM

Exemple (9)

(9) ***PGM***

```
* Exemple 1.9 : Prostatic cancer;

Data Prostate;
 input age acide x taille grade Y;
 n=1;
 lacide=log(acide);
cards;
   66  0.48 0      0     0 0
   68  0.56 0      0     0 0
   66  0.50 0      0     0 0
   56  0.52 0      0     0 0
   58  0.50 0      0     0 0
   60  0.49 0      0     0 0
   65  0.46 1      0     0 0
   60  0.62 1      0     0 0
   50  0.56 0      0     1 1
   49  0.55 1      0     0 0
   61  0.62 0      0     0 0
   58  0.71 0      0     0 0
   51  0.65 0      0     0 0
   67  0.67 1      0     1 1
   67  0.47 0      0     1 0
   51  0.49 0      0     0 0
   56  0.50 0      0     1 0
   60  0.78 0      0     0 0
   52  0.83 0      0     0 0
```

```
   56  0.98 0       0      0 0
   67  0.52 0       0      0 0
   63  0.75 0       0      0 0
   59  0.99 0       0      1 1
   64  1.87 0       0      0 0
   61  1.36 1       0      0 1
   56  0.82 0       0      0 1
   64  0.40 0       1      1 0
   61  0.50 0       1      0 0
   64  0.50 0       1      1 0
   63  0.40 0       1      0 0
   52  0.55 0       1      1 0
   66  0.59 0       1      1 0
   58  0.48 1       1      0 1
   57  0.51 1       1      1 1
   65  0.49 0       1      0 1
   65  0.48 0       1      1 0
   59  0.63 1       1      1 0
   61  1.02 0       1      0 0
   53  0.76 0       1      0 0
   67  0.95 0       1      0 0
   53  0.66 0       1      1 0
   65  0.84 1       1      1 1
   50  0.81 1       1      1 1
   60  0.76 1       1      1 1
   45  0.70 0       1      1 1
   56  0.78 1       1      1 1
   46  0.70 0       1      0 1
   67  0.67 0       1      0 1
   63  0.82 0       1      0 1
   57  0.67 0       1      1 1
   51  0.72 1       1      0 1
   64  0.89 1       1      0 1
   68  1.26 1       1      1 1
;
run;
```
PGM

1.2 Other examples

Exemple `Insecte` *: (10)*. On utilise un exemple sur la résistance d'un papillon de nuit au pyrethroid trans-cypermethrin.

(10) ***PGM***

```
* Exemple 3.7 : Insecte;

Data Insecte;
 retain n 20;
 do sexe='male    ','femelle';
    do dose=1,2,4,8,16,32;
       input k @@;
       output;
    end;
 end;
cards;
1 4 9 13 18 20
0 2 6 10 12 16
```

```
;
run;
```
PGM

Exemple `Beetle` *: (11).* Des lots d'environ 50 insectes sont exposés à trois types d'insecticides à différentes doses. Le premier insecticide est à base de DDT, le deuxième à base de BHC et le troisième insecticide est un mélange des 2 premiers.

(11) ***PGM***
```
* Exemple 4.3 : Beetle;

Data beetles;
 format insect $8.;
 Do insect='DDT', 'BHC', 'DDT+BHC';
    Do dose=2, 2.64, 3.48, 4.59, 6.06, 8;
       input y n @@;
       ldose=log(dose);
       output;
    end;
 end;
datalines;
 3  50   5  49  19  47  19  50  24  49  35  50
 2  50  14  49  20  50  27  50  41  50  40  50
28  50  37  50  46  50  48  50  48  50  50  50
;
run;
```
PGM

Exemple `CS2` *: (12).* Des lots d'environ 30 insectes sont exposés à différentes doses de CS^2. On observe le nombre d'insectes tués. L'expérience est répétée deux fois.

(12) ***PGM***
```
* Exemple 4.5 : CS2;

Data CS2;
 input conc @;
   do rep=1 to 2;
      input y n @;
      output;
   end;
cards;
49.06    2         29        4         30
52.99    7         30        6         30
56.91    9         28        9         34
60.84    14        27        14        29
64.76    23        30        29        33
68.69    29        31        24        28
72.61    29        30        32        32
76.54    29        29        31        31
;
run;
```
PGM

Exemple `Lupin` *: (13).* La probabilité de germination de graines de lupin est étudiée en fonction de la température, du niveau d'humidité et de la température de stockage.

(13) ***PGM***
```
* Exemple 5.3 : Lupin;

Data Lupin;
 do tp_germ=11, 21;
    do moisir=1,2,3;
       do tp_cons=21,42,62;
```

```
          input y @@;
          n=100;
          output;
       end;
    end;
 end;
cards;
98 96 62
94 79 3
92 41 1
94 93 65
94 71 2
91 30 1
;
run;
```

PGM

Chapitre 2

Statistical inference for binary data

2.1 The binomial distribution

Un marchand de graines annonce une probabilité de germination égale à 0.8. Sur un échantillon de 12 graines mises à germer, 7 graines ont germé. La probabilité estimée est de $7/12 = 0.583$, ce qui est peu encourageant. Peut-on accepter l'hypothèse d'une probabilité égale à 0.8 ?

H_0 : "$p = 0.8$"

H_1 : "$p < 0.8$" (test unilateral)

Test exact

PGM

```
%macro binom(n=, p=); * Debut de la macro;
Data Binom;
 do y=0 to &n;
   if y=0 then p=probbnml(&p,&n,y);
   else p=probbnml(&p,&n,y) - probbnml(&p,&n,y-1);
   pcum=probbnml(&p,&n,y);
   output;
 end;
run;
%mend binom;         * Fin de la macro;
%binom(n=12,p=0.8);
Proc print data=Binom round;
 format p pcum 6.3;
run;
Proc Gplot data=Binom;
 symbol1 v=none i=needle;
 axis1 offset=(0,0) order=0 to .3 by .05;
 axis2 offset=(1,1) order=0 to 12 by 2;
 plot p*y=1 /hm=0 vm=0 vaxis=axis1 haxis=axis2 frame;
run;
```

PGM

Si H_0 est vraie (si la probabilité est réellement égale à 0.8 et si j'ai réellement une loi binomiale, c'est à dire, si les graines sont indépendantes les unes des autres) et que je mets 12 graines à germer, la proba d'obtenir exactement 9 graines qui ont germé vaut 0.236; la proba d'obtenir 0, 1, 2, 3, 4, 5, 6, ou 7 graines qui germent vaut 0.073.

OUTPUT

```
OBS     Y         P      PCUM
```

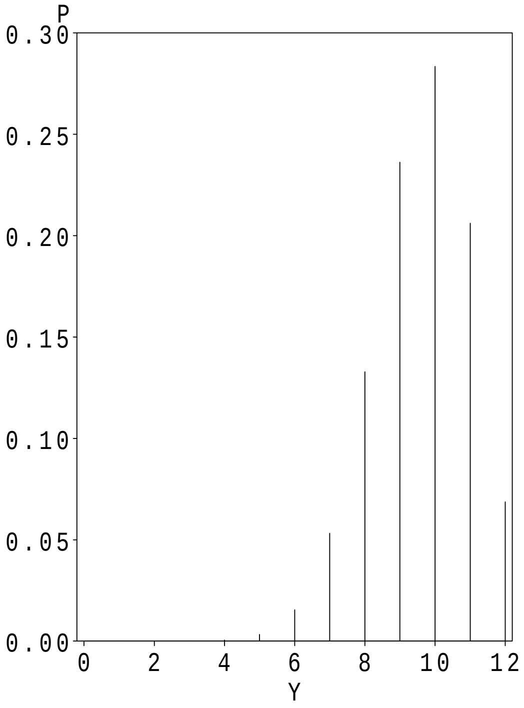


FIG. 2.1:

1	0	0.000	0.000
2	1	0.000	0.000
3	2	0.000	0.000
4	3	0.000	0.000
5	4	0.001	0.001
6	5	0.003	0.004
7	6	0.016	0.019
8	7	0.053	0.073
9	8	0.133	0.205
10	9	0.236	0.442
11	10	0.283	0.725
12	11	0.206	0.931
13	12	0.069	1.000

OUTPUT

Donc, sous l'hypothese H_0, la proba d'obtenir un nombre de germinations aussi petit que 7, vaut 0.073. La $p - value$ de mon test vaut 0.073. C'est une estimation du risque d'erreur si je rejette H_0. Si je me suis fixé un risque de première espèce (α) égal à 5%, ces données ne me permettent pas de rejetter H_0, elles ne me permettent pas de remettre en cause la parole du marchand. Par contre, si j'accepte un risque d'erreur égal à 0.10, je remets en cause l'hypothese nulle.
J'ai 7% de chances de me tromper si j'affirme que le marchand a surestimé sa proba de germination. On peut donc avoir des doutes, et soupçonner une surestimation de la probabilité de germination, sans qu'on puisse la mettre en évidence de facon certaine.

Test approché (basé sur une approximation normale)

PGM

```
Data _null_;
 z=(7/12 - 0.8) / sqrt(0.8*(1-0.8)/12);
 p=round(probnorm(z),.0001);
 put 'Probabilite p= ' p //;
run;
```

PGM

La $p - value$ vaut 0.03. Je peux donc rejetter H_0 avec un risque d'erreur faible (3%).

LOG

```
Probabilite p= 0.0303
```

LOG

2.2 Inference about the success probability

2.3 Comparison of two proportions

Utilisation de l'exemple `Souris` (exemple (1), page 5) de comparaison des effets du tabac sur le développement d'une tumeur. Dans cet exemple, des souris sont placées dans des cages dans lesquelles on injecte ou non de la fumée de cigarette. Après un certain temps, on sacrifie les animaux et on observe ou non le développement d'une tumeur.

PGM

```
Proc freq data=souris order=data;
 table Tumeur*Group /nocol norow;
 weight y;
run;
```

PGM

$H_0 : p_1 = p_2$
$H_1 : p_1 \neq p_2$
On met en oeuvre un test bilatéral comme le fait Collett, mais on aurait pu choisir, dans ce cas, de faire un test unilatéral.

OUTPUT

```
TABLE OF TUMEUR BY GROUP
TUMEUR     GROUP
Frequency|
Percent  |1-Cigare|0-Contro|  Total
         |tte     |le      |
---------+--------+--------+
Oui      |     21 |     19 |     40
         |  38.18 |  34.55 |  72.73
---------+--------+--------+
Non      |      2 |     13 |     15
         |   3.64 |  23.64 |  27.27
---------+--------+--------+
Total          23       32       55
            41.82    58.18   100.00
```

OUTPUT

Méthode approchée, test direct

PGM

```
Data _null_;
 n1=23; n2=32;
 p1=21/n1; p2=19/n2;
 diff=p1-p2;
 se_diff=sqrt( p1*(1-p1)/n1 + p2*(1-p2)/n2 );
```

```
 z=diff/se_diff;
 pvalue=round(2*(1-probnorm(z)),.00001);
 put 'Probabilite p= ' pvalue //;
 borneinf=diff - probit(.975)*se_diff;
 bornesup=diff + probit(.975)*se_diff;
 put 'IC = [' borneinf ',' bornesup ']' //;
run;
```

PGM

On calcule la probabilité (p) et l'intervalle de confiance (IC).

LOG

```
Probabilite p= 0.00232
IC = [0.113826031 ,0.5247609256 ]
```

LOG

Une autre façon de faire consiste à calculer l'écart-type de la difference des probabilités en remplaçant la probabilité de chacun des groupes par la probabilité commune.

PGM

```
Data _null_;
 n1=23; n2=32;
 p1=21/n1; p2=19/n2; p=(21+19)/(n1+n2);
 diff=p1-p2;
 se_diff=sqrt( p*(1-p)* (1/n1+1/n2));
 z=diff/se_diff;
 pvalue=round(2*(1-probnorm(z)),.00001);
 put 'Probabilite p= ' pvalue //;
run;
```

PGM

LOG

```
Probabilite p= 0.00873
```

LOG

Méthode directe

PGM

```
Proc freq data=souris;
 table tumeur*group /exact noprint;
 weight y;
run;
```

PGM

OUTPUT

```
STATISTICS FOR TABLE OF TUMEUR BY GROUP
Statistic                     DF     Value        Prob
------------------------------------------------------
Chi-Square                     1     6.878       0.009
Fisher's Exact Test (Left)                       0.999
                    (Right)                   8.34E-03
                    (2-Tail)                     0.013
Sample Size = 55
```

OUTPUT

Le test de Fisher (`Fisher's Exact Test (2-Tail)`) donne une $p - value = 0.013$ et le test χ^2 ((`Chi-Square`) une $p - value = 0.009$.

odds-ratio

PGM

```
Proc freq data=souris order=data;
 table tumeur*group /cmh;
 weight y;
run;
```

PGM

OUTPUT

```
      Estimates of the Common Relative Risk (Row1/Row2)
                                                   95%
Type of Study    Method             Value   Confidence Bounds
-------------------------------------------------------------
Case-Control     Mantel-Haenszel     7.184      1.624   31.789
  (Odds Ratio)   Logit               7.184      1.432   36.044
The confidence bounds for the M-H estimates are test-based.

Total Sample Size = 55
```

OUTPUT

La valeur de l'odds-ratio (`Odds Ratio`) est de 7.184 avec un intervalle de confiance [1.4 , 36]. Cet IC ne contient pas la valeur 1. La probabilité de tumeur est donc plus élevée pour les souris exposées que pour les autres.

2.4 Comparison of two or more proportions

Utilisation de l'exemple `Prune` (exemple (2), page 5). Dans cet exemple, on cherche à comparer la proportion de plants qui "prennent" ou pas selon qu'ils sont plantés immédiatement ou au printemps et selon la taille des racines (courte ou longue).

PGM

```
Data Table;
 set Prune;
 length etat $8.;
 array tmp* vivant mort;
     do i=1 to dim(tmp);
        poids=tmp(i);
        Group=substr(longueur,3,3)|| substr(periode,3,3);
        call vname(tmp(i),etat);
        output;
     end;
drop i vivant mort longueur periode;
run;
```

PGM

Calcul des statistiques.

PGM

```
Proc freq data=Table order=data;
 table etat*group/chisq nocol norow;
 weight poids;
run;
```

PGM

OUTPUT

```
TABLE OF ETAT BY GROUP
ETAT      GROUP
Frequency|
Percent  |coupre  |coutar  |lonpre  |lontar  |  Total
---------+--------+--------+--------+--------+
VIVANT   |    107 |     31 |    156 |     84 |    378
         |  11.15 |   3.23 |  16.25 |   8.75 |  39.38
---------+--------+--------+--------+--------+
MORT     |    133 |    209 |     84 |    156 |    582
         |  13.85 |  21.77 |   8.75 |  16.25 |  60.63
---------+--------+--------+--------+--------+
Total         240      240      240      240      960
            25.00    25.00    25.00    25.00   100.00


STATISTICS FOR TABLE OF ETAT BY GROUP
Statistic                     DF     Value        Prob
------------------------------------------------------
Chi-Square                     3   141.053       0.001
Sample Size = 960
```

OUTPUT

Collett présente un autre exemple qui montre qu'il est important de ne pas grouper les données, c'est à dire de négliger un facteur de variation. Le risque est d'aboutir à une conclusion contraire à celle obtenue en considérant tous les facteurs de variation.

PGM

```
Data enfants;
 input clinique$ periode$ etat$ poids;
 ge=substr(clinique,1,1)|| substr(periode,1,2);
cards;
A   -1mois vivant  176
A   -1mois mort      3
A   +1mois vivant  293
A   +1mois mort      4
B   -1mois vivant  197
B   -1mois mort     17
B   +1mois vivant   23
B   +1mois mort      2
;
run;
```

PGM

PGM

```
Proc freq data=enfants;
 table etat*ge/chisq noprint;
 weight poids;
run;
```

PGM

Statistique obtenue pour les 2 facteurs (Clinique et Période croisés).

OUTPUT

```
STATISTICS FOR TABLE OF ETAT BY GE
Statistic                     DF     Value        Prob
------------------------------------------------------
Chi-Square                     3    19.096       0.001
Sample Size = 715
```

OUTPUT

PGM

```
Proc freq data=enfants(where=(clinique ='A'));
 table etat*periode/chisq noprint;
 weight poids;
run;
```

PGM

Statistique obtenue pour la clinique A.

OUTPUT

```
STATISTICS FOR TABLE OF ETAT BY PERIODE
Statistic                     DF     Value        Prob
------------------------------------------------------
Chi-Square                     1     0.084       0.773
Sample Size = 476
WARNING:  50% of the cells have expected counts less
          than 5. Chi-Square may not be a valid test.
```

OUTPUT

Statistique obtenue pour la clinique B.

PGM

```
Proc freq data=enfants(where=(clinique ='B'));
 table etat*periode/chisq noprint;
 weight poids;
run;
```

PGM

OUTPUT

```
STATISTICS FOR TABLE OF ETAT BY PERIODE
Statistic                     DF     Value        Prob
------------------------------------------------------
Chi-Square                     1     0.000       0.992
Sample Size = 239
WARNING:  25% of the cells have expected counts less
          than 5. Chi-Square may not be a valid test.
```

OUTPUT

Statistique obtenue en groupant les données sur les 2 cliniques (A et B ignorées).

PGM

```
Proc freq data=enfants;
 table etat*periode/chisq noprint;
 weight poids;
run;
```

PGM

OUTPUT

```
STATISTICS FOR TABLE OF ETAT BY PERIODE
Statistic                     DF     Value        Prob
------------------------------------------------------
Chi-Square                     1     5.255       0.022
Sample Size = 715
```

OUTPUT

Pour chaque clinique, on ne trouve pas d'effet de la durée du soin prénatal (`periode`) sur la mortalité. Si on regroupe les données des 2 cliniques, donc sans tenir compte du facteur `clinique`, on trouve un effet de la durée du soin prénatal, ce qui est incohérent avec les résultats obtenus clinique par clinique.

Chapitre 3

Models for binary and binomial data

3.1 Statistical modelling

3.2 Linear Models

3.3 Methods of estimation

3.4 Fitting linear models to binomial data

Utilisation de l'exemple `Seum` (exemple (6), page 7) d'un serum anti-pneumocoque. Cinq doses de serum sont expérimentées. Pour chaque dose, on utilise un lot de 40 souris. Les souris succombent ou pas.
Le modèle linéaire est de la forme

$$E(y_i/n_i) = p_i = \beta_0 + \beta_1 d_i$$

avec y_i le nombre de souris mortes dans chaque lot et d_i la dose de serum.

PGM

```
Proc reg data=serum graphics;
 symbol1 v=dot;
 model p=dose /p;
 plot p*dose /haxis=0 to 0.05 by 0.02;
run;
```

PGM

... la sortie donne le modèle

$$\hat{p_i} = 0.64 - 16.08 d_i$$

.

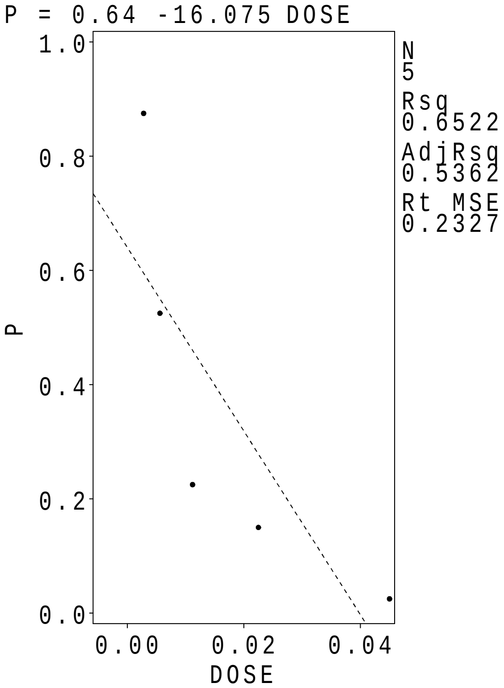
OUTPUT

```
Parameter Estimates

                  Parameter      Standard    T for H0:
Variable  DF       Estimate         Error   Parameter=0    Prob > |T|

INTERCEP   1       0.640025    0.15738886         4.067        0.0268
DOSE       1     -16.074895    6.77809894        -2.372        0.0984
```

```
         Dep Var   Predict
  Obs       P        Value  Residual

    1    0.8750    0.5950    0.2800
    2    0.5250    0.5500   -0.0250
    3    0.2250    0.4600   -0.2350
    4    0.1500    0.2783   -0.1283
    5    0.0250   -0.0833    0.1083
```

OUTPUT

On obtient une probabilité estimée négative, pour la dose la plus élevée ce qui est pour le moins génant.

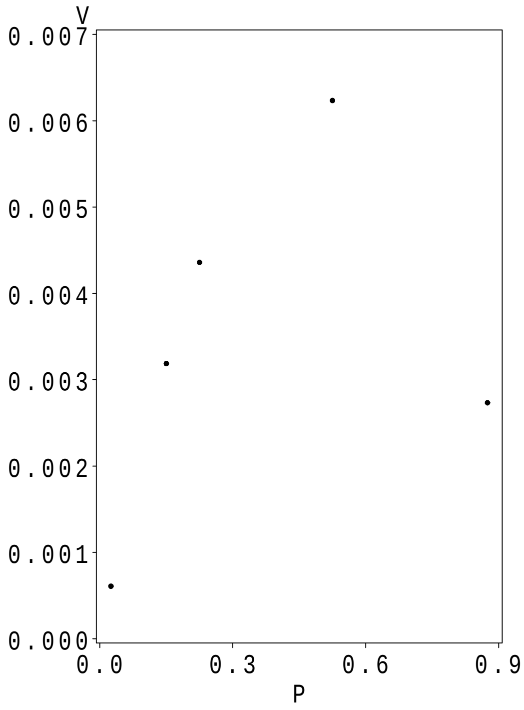


Dans le cas de données binomiales, la variance est égale à $p(1-p)/n$. Elle n'est donc pas constante selon la valeur de p.

PGM

```
Data serum;
 set serum;
 v=p*(1-p)/n;
run;
Proc gplot data=serum;
 symbol1 v=dot;
 plot v*p=1 / hm=0 vm=0 haxis=0 to 0.9 by 0.3;
run;
```

PGM



L'utilisation d'un modèle linéaire pose deux problèmes : le Modèle Linéaire ne tient pas compte (1) de l'hétérogénéité des variances; (2) de la contrainte d'appartenance des probabilités estimées à l'intervalle [0,1].

3.5 Models for binomial response data

Préalablement à l'utilisation d'un modèle linéaire on va transformer l'échelle des probabilités (0,1) en échelle $(-\infty, +\infty)$. Trois transformations sont possibles : la transformation logistique, la transformation probit et la transformation clog-log.

3.5.1 The logistic transformation

La transformation logistique d'une probabilité consiste à prendre $\log(p/(1-p))$.

3.5.2 The probit transformation

La transformation probit d'une probabilité p est définie comme étant la valeur ξ telle que :

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\xi} \exp(-\frac{1}{2}u^2)du = p$$

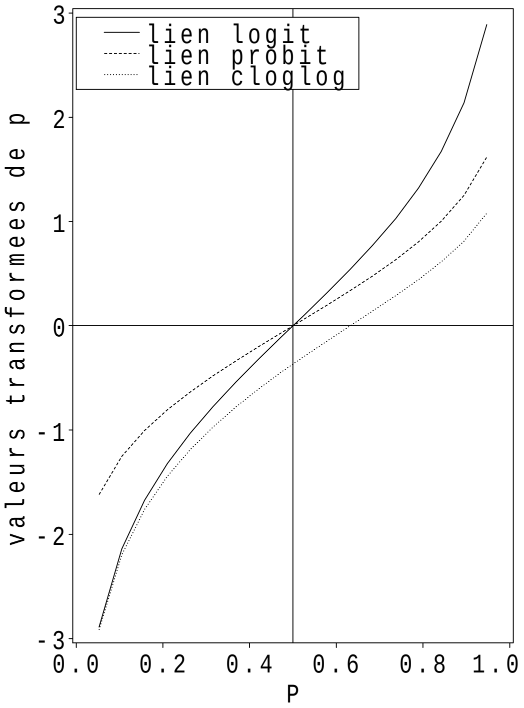
3.5.3 The complementary log-log transformation

La transformation cloglog d'une probabilité p est égale à $\log(-\log(1-p))$

PGM

```
Data Proba;
 do p=0 to 1 by 0.0001;
    plogit=log(p/(1-p));
    pprobit=probit(p);
    pclog=log(-log(1-p));
    output;
 end;
run;
```

```
axis1 label=(a=90 'valeurs transformees de p')
      order=-4 to 4 by 1 minor=none;
axis2 order=0 to 1 by .2 minor=none;

legend1 down=3 offset=(,-1)
        position=(top left inside) label=('') mode=protect
        value=('lien logit' 'lien probit' 'lien cloglog');

proc Gplot data=Proba;
 symbol1 v=none i=join l=1 c=black;
 symbol2 v=none i=join l=2 c=black;
 symbol3 v=none i=join l=33 c=black;
 plot (plogit pprobit pclog)*p /overlay frame legend=legend1
                                vref=0 vaxis=axis1 href=.5 haxis=axis2;
run;
```
PGM



3.6 The linear logistic model

3.7 Fitting the linear logistic model to binomial data

Utilisation de l'exemple `Aircraft` (exemple (5), page 6). Test de charge pour les rivets en alliage utilisés dans l'aviation. On teste des lots de rivets avec des charges croissantes.
Un modèle possible relie la probabilité p (après transformation logistique) à la charge de la façon suivante :

$$\log(p_i/(1-p_i)) = \beta_0 + \beta_1 \text{load}_i$$

PGM
```
Proc genmod data=aircraft;
 make 'obstats' out=glm1;
 model k/n=load /dist=bin corrb obstats residuals;
run;
```

PGM

L'équation du modèle ajusté est logit(p) = -5.3397 + 0.0015 load.

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

              Criterion              DF          Value      Value/DF

              Deviance                8         0.3719        0.0465
              Scaled Deviance         8         0.3719        0.0465
              Pearson Chi-Square      8         0.3707        0.0463
              Scaled Pearson X2       8         0.3707        0.0463
              Log Likelihood          .      -421.8560             .

                       Analysis Of Parameter Estimates

          Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

          INTERCEPT     1     -5.3397      0.5457     95.7499  0.0001
          LOAD          1      0.0015      0.0002     96.6084  0.0001
          SCALE         0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                         Estimated Correlation Matrix

                         Parameter
                         Number           PRM1     PRM2

                         PRM1           1.0000  -0.9884
                         PRM2          -0.9884   1.0000
```

OUTPUT

Un modèle équivalent, mais avec centrage de la variable explicative peut aussi être utilisé. Seule la paramétrisation change. Le terme constant correspond à l'ordonnée au centre du domaine de variation des abscisses (le terme constant correspondait à l'ordonnée à l'origine, dans le modèle précédent).

PGM

```
Proc standard data=aircraft out=air2 mean=0;
 var load;
run;
Proc genmod data=air2;
 model k/n=load /dist=bin corrb;
run;
```

PGM

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

              Criterion              DF          Value      Value/DF

              Deviance                8         0.3719        0.0465
              Scaled Deviance         8         0.3719        0.0465
              Pearson Chi-Square      8         0.3707        0.0463
              Scaled Pearson X2       8         0.3707        0.0463
```

```
                Log Likelihood          .      -421.8560            .

                          Analysis Of Parameter Estimates

            Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

            INTERCEPT     1     -0.0750      0.0830      0.8182  0.3657
            LOAD          1      0.0015      0.0002     96.6084  0.0001
            SCALE         0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                           Estimated Correlation Matrix

                          Parameter
                          Number          PRM1     PRM2

                          PRM1          1.0000  -0.0450
                          PRM2         -0.0450   1.0000
```
OUTPUT

La déviance est la même que pour le modèle précédent mais les valeurs des paramètres ont changé. La corrélation entre les 2 paramètres est presque nulle et les 2 paramètres estimés peuvent être interprétés indépendamment l'un de l'autre.
Les deux modèles sont équivalents : la déviance résiduelle est la même, les probabilités ajustées sont les mêmes ...

PGM

```
Data Result;
 merge aircraft(keep= load k n) glm1(keep=pred xbeta);
 p=k/n;
 drop n;
run;
Proc print data=Result;
 var load xbeta p pred k k_a;
```
PGM

OUTPUT

OBS	LOAD	XBETA	P	PRED	K	K_A
1	2500	-1.46863	0.20000	0.18715	10	9.3576
2	2700	-1.15894	0.24286	0.23886	17	16.7202
3	2900	-0.84925	0.30000	0.29959	30	29.9589
4	3100	-0.53957	0.35000	0.36829	21	22.0973
5	3300	-0.22988	0.45000	0.44278	18	17.7113
6	3500	0.07981	0.50588	0.51994	43	44.1950
7	3700	0.38949	0.60000	0.59616	54	53.6545
8	3900	0.69918	0.66000	0.66801	33	33.4003
9	4100	1.00887	0.75000	0.73280	60	58.6239
10	4300	1.31855	0.78462	0.78894	51	51.2812

OUTPUT

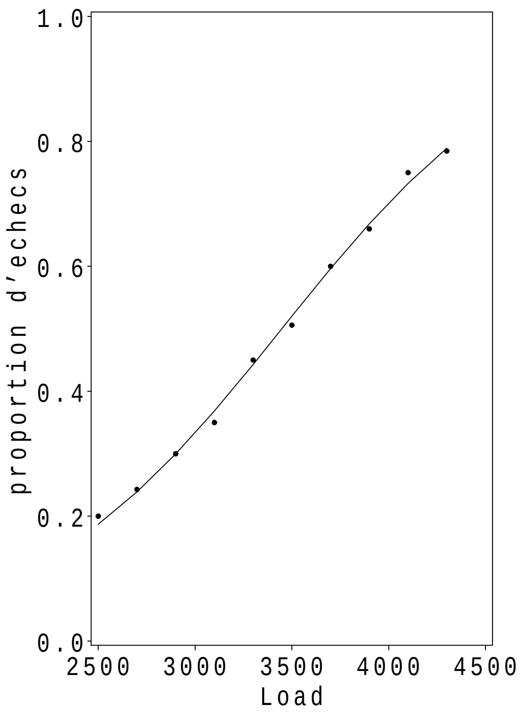
avec `XBETA` le prédicteur linéaire, `P` la probabilité observée, `PRED` la probabilité estimée, `K` le nombre observé et `K_A` le nombre estimé.
Le graphe de la courbe estimée et des observations est obtenu avec le programme suivant.

PGM

```
Proc Gplot data=Result;
```

```
 symbol1 v=star i=none c=black;
 symbol2 v=none i=join c=black;
 axis1 label=('Load') order=2500 to 4500 by 500 minor=none offset=(1,);
 axis2 label=(a=90 "proportion d'echecs") order=0 to 1 by .2 minor=none;
 plot p*load=1 pred*load=2 /overlay vaxis=axis2 haxis=axis1 frame;
run;
quit;
```

PGM



autre exemple : "les souris qui fument"
Utilisation de l'exemple `Souris` (exemple (1), page 5). Dans cet exemple, des souris sont placées dans des cages dans lesquelles on injecte (groupe 1) ou non (groupe 0) de la fumée de cigarette. Après un certain temps, on sacrifie les animaux et on regarde si l'on trouve une tumeur.
Deux modèles sont possibles :

Modèle 1 : $E(y_i) = n_i p_i$ avec $\log(p_i/(1-p_i)) = \beta_0,\ i = 1, 2$

Modèle 2 : $E(y_i) = n_i p_i$ avec $\log(p_i/(1-p_i)) = \beta_0 + \gamma_i,\ i = 1, 2$

avec γ_i représantant l'effet du traitement i.
Le modèle 1 suppose l'égalité des probabilités de tumeur entre les 2 groupes "cigarette" et "controle". La probabilité commune estimée par maximum de vraisemblance est équivalente à la proportion observée.

PGM

```
Proc genmod data=Souris;
 class group;
 model y/n= /dist=bin obstats;
run;
```

PGM

La probabilité estimée par maximum de vraisemblance est égale à la probabilité globale de développer une tumeur : $p = (19 + 21)/(23 + 32) = 0.727$.

OUTPUT

```
                  Criteria For Assessing Goodness Of Fit

             Criterion              DF          Value      Value/DF

             Deviance                1         7.6349        7.6349
             Scaled Deviance         1         7.6349        7.6349
             Pearson Chi-Square      1         6.8781        6.8781
             Scaled Pearson X2       1         6.8781        6.8781
             Log Likelihood          .       -32.2274             .

                     Analysis Of Parameter Estimates

         Parameter    DF     Estimate      Std Err    ChiSquare  Pr>Chi

         INTERCEPT     1       0.9808       0.3028      10.4948  0.0012
         SCALE         0       1.0000       0.0000            .       .

NOTE:  The scale parameter was held fixed.

                         Observation Statistics
         Y          N       Pred       Xbeta         Std     HessWgt       Lower

        21         23     0.7273      0.9808      0.3028      4.5620      0.5957
        19         32     0.7273      0.9808      0.3028      6.3471      0.5957
```
OUTPUT

Le modèle 2 fait l'hypothèse d'une inégalité des probabilités de tumeur entre les 2 groupes "cigarette" et "controle".

PGM

```
Proc genmod data=Souris;
 class group;
 model y/n=group /dist=bin obstats;
run;
```
PGM

Le modèle est saturé : on a autant de paramètres (2) que de traitements (2). Un traitement est une combinaison des niveaux des facteurs (ici, il n'y a qu'un facteur). C'est parce que le modèle est saturé que la déviance résiduelle est nulle.
Les probabilités estimées (par maximum de vraisemblance) sont égales aux proportions observées dans les deux groupes : $p_1 = 21/23 = 0.913$ et $p_2 = 19/32 = 0.594$.

OUTPUT

```
                  Criteria For Assessing Goodness Of Fit

             Criterion              DF          Value      Value/DF

             Deviance                0         0.0000             .
             Scaled Deviance         0         0.0000             .
             Pearson Chi-Square      0         0.0000             .
             Scaled Pearson X2       0         0.0000             .
             Log Likelihood          .       -28.4100             .

                     Analysis Of Parameter Estimates

    Parameter                  DF     Estimate      Std Err    ChiSquare  Pr>Chi
```

```
    INTERCEPT                  1       2.3514       0.7400      10.0964  0.0015
    GROUP      0-Controle      1      -1.9719       0.8229       5.7420  0.0166
    GROUP      1-Cigarette     0       0.0000       0.0000        .       .
    SCALE                      0       1.0000       0.0000        .       .

NOTE:  The scale parameter was held fixed.

                           Observation Statistics

         Y          N        Pred       Xbeta         Std     HessWgt       Lower

        21         23      0.9130      2.3514      0.7400      1.8261      0.7112
        19         32      0.5938      0.3795      0.3599      7.7188      0.4192
```
OUTPUT

Le même modèle, avec une autre paramétrisation : on enlève le terme constant. L'avantage de cette paramétrisation, est dans l'interprétation plus directe des paramètres estimés. Il n'y a plus de terme constant, mais il y a autant de paramètres que de niveaux du facteur. Les paramètres estimés sont alors les transformées logit des $\hat{p}$. Les $\hat{p}$ sont donc obtenus en appliquant la transformée inverse (réciproque) de la fonction de lien.

PGM

```
Proc genmod data=Souris;
 class group;
 model y/n=group /dist=bin obstats noint;
run;
```
PGM

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

              Criterion             DF          Value        Value/DF

              Deviance               0         0.0000               .
              Scaled Deviance        0         0.0000               .
              Pearson Chi-Square     0         0.0000               .
              Scaled Pearson X2      0         0.0000               .
              Log Likelihood         .       -28.4100               .

                       Analysis Of Parameter Estimates

    Parameter                 DF     Estimate      Std Err    ChiSquare  Pr>Chi

    INTERCEPT                  0       0.0000       0.0000        .       .
    GROUP      0-Controle      1       0.3795       0.3599       1.1116  0.2917
    GROUP      1-Cigarette     1       2.3514       0.7400      10.0964  0.0015
    SCALE                      0       1.0000       0.0000        .       .

NOTE:  The scale parameter was held fixed.

                           Observation Statistics

         Y          N        Pred       Xbeta         Std     HessWgt       Lower

        21         23      0.9130      2.3514      0.7400      1.8261      0.7112
        19         32      0.5938      0.3795      0.3599      7.7188      0.4192
```
OUTPUT

3.8 Goodness of fit of a linear logistic model

Utilisation de l'exemple `Aircraft` (exemple (5), page 6). Test de charge pour les rivets en alliage utilisés dans l'aviation. On teste des lots, de différentes tailles, de rivets avec des charges croissantes. Le modèle utilisé est le suivant :

$$\log(p_i/(1-p_i)) = \beta_0 + \beta_1 \mathrm{load}_i$$

On a 10 observations binomiales et un modèle à 2 paramètres donc 8 degrés de liberté. La déviance est égale à 0.37 pour 8 ddl et elle suit sous l'hypothèse H_0 d'un modèle adéquat une loi de χ^2 à 8 ddl. Que penser de cette valeur 0.37 ?
L'espérance d'un χ^2, c'est son ddl : en gros, le modèle est adéquat lorsque la déviance est de l'ordre de grandeur du ddl. Une déviance *trop grande* par rapport au ddl est un indicateur d'un manque d'adéquation du modèle.
Ici,la déviance est **SUSPECTE**. La probabilité d'obtenir 0.37, pour une V.A. qui suit une loi χ^2 à 8 ddd, vaut 10^{-5}. Le $(1-\alpha)$ quantile d'un χ^2 à 8 ddl vaut 15.5 (avec $\alpha = 0.05$), ce qui signifie qu'une déviance inférieure ou égale à 15.5 conduit à accepter l'hypothèse d'adequation du modèle. Deux explications sont possibles : (1) soit les données sont artificielles, (2) soit les observations binaires sont corrélées négativement (ce qui conduirait à une sous-dispersion).

PGM

```
Data _null_;
 p=probchi(0.37,8);
 put p;
 q=cinv(0.95,8);
 put q;
run;
```

PGM

LOG

```
0.0000421115
15.507313056
```

LOG

`deviance` et `scaled deviance`.
Dans les exemples précédents, `deviance` et `scaled deviance` étaient identiques, car le paramètre d'échelle (`scale parameter` = dispersion parameter) est fixé et égal à 1. Ce choix correspond au choix de la distribution binomiale, qui suppose l'indépendance entre les observations binaires.
Dans certain cas, on n'a pas indépendance entre les observations binaires. Les observations peuvent être corrélées négativement ou positivement (cas le plus courant). Une façon de résoudre ce problème est d'introduire un paramètre "dispersion" ϕ tel que :

$$\mathrm{Var}(\mu) = \phi\ \mathrm{Var}(Y)$$

c'est à dire :

$$\mathrm{Var}(p) = \phi p(1-p)$$

dans le cas d'une extension de la loi binomiale à des observations corrélées.
Si $\phi > 1$, on parle de sur-dispersion, si $\phi < 1$ on parle de sous-dispersion. La sur-dispersion résulte de corrélations positives, alors que la sous-dispersion résulte de corrélations négatives.
La statistique du X^2 de Pearson peut être obtenue en calculant la somme des carrés des résidus de Pearson. Dans cet exemple, elle est proche de la deviance. Des valeurs très différentes de ces deux statistiques (`Pearson Chi-Square` et déviance) sont des indicateurs de la mauvaise qualité de l'approximation du χ^2 ("on est loin de l'asymptotique").
On utilise la déviance pour apprécier la qualité de l'ajustement (lorsque les données sont groupées), plutôt que la statistique du χ^2, parce que cette statistique est cohérente avec la méthode d'estimation (maximum de vraisemblance).

3.9 Comparing linear logistic models

Utilisation de l'exemple `Prune` (exemple (2), page 5) des racines de pruniers. Deux périodes de plantation et deux tailles de racines sont utilisées. Pour chaque traitement, on utilise 240 morceaux de racines. Après un certain temps, on regarde si la racine est vivante ou morte.
Pour cet exemple on a 5 modèles possibles :

Modèle 1 : $\log(p_{jk}/(1-p_{jk})) = \beta_0$

Modèle 2 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \lambda_j$

Modèle 3 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \tau_k$

Modèle 4 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \lambda_j + \tau_k$

Modèle 5 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \lambda_j + \tau_k + (\lambda\tau)_{jk}$

avec p_{jk} la probabilité qu'une racine de taille j ($j = 1, 2$) plantée au temps k ($k = 1, 2$) survive.

PGM

```
* Modele 5;

proc genmod data=prune;
 class longueur periode;
 model vivant/n = periode | longueur /dist=bin type1;
run;
```

PGM

L'ajout de l'un ou l'autre des facteurs diminue significativement la déviance. L'ajout des deux facteurs améliore encore significativement le modèle. L'ajout du terme d'interaction n'apporte pas d'amélioration significative. Le modèle interactif complet est saturé (quatre paramètres sont ajustés pour quatre observations binomiales).

OUTPUT

LR Statistics For Type 1 Analysis

Source	Deviance	DF	ChiSquare	Pr>Chi
INTERCEPT	151.0193	0	.	.
PERIODE	53.4404	1	97.5789	0.0001
LONGUEUR	2.2938	1	51.1466	0.0001
LONGUEUR*PERIODE	0.0000	1	2.2938	0.1299

OUTPUT

On teste le modèle 4 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \lambda_j + \tau_k$.

PGM

```
* Modele 4;

proc genmod data=prune;
 class longueur periode;
 model vivant/n = periode longueur /dist=bin type1;
run;
```

PGM

OUTPUT

LR Statistics For Type 1 Analysis

Source	Deviance	DF	ChiSquare	Pr>Chi
INTERCEPT	151.0193	0	.	.
PERIODE	53.4404	1	97.5789	0.0001
LONGUEUR	2.2938	1	51.1466	0.0001

OUTPUT

On teste le modèle 3 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \tau_k$.

PGM

```
* Modele 3;
proc genmod data=prune;
 class longueur periode;
 model vivant/n = periode /dist=bin type1;
run;
```

PGM

OUTPUT

```
                    LR Statistics For Type 1 Analysis

              Source       Deviance    DF   ChiSquare  Pr>Chi

              INTERCEPT    151.0193     0           .       .
              PERIODE       53.4404     1     97.5789  0.0001
```

OUTPUT

On teste le modèle 2 : $\log(p_{jk}/(1-p_{jk})) = \beta_0 + \lambda_j$.

PGM

```
* Modele 2;
proc genmod data=prune;
 class longueur periode;
 model vivant/n = longueur /dist=bin type1;
run;
```

PGM

OUTPUT

```
                    LR Statistics For Type 1 Analysis

              Source       Deviance    DF   ChiSquare  Pr>Chi

              INTERCEPT    151.0193     0           .       .
              LONGUEUR     105.1824     1     45.8369  0.0001
```

OUTPUT

On teste le modèle 1 : $\log(p_{jk}/(1-p_{jk})) = \beta_0$.

PGM

```
* Modele 1;
proc genmod data=prune;
 class longueur periode;
 model vivant/n = longueur /dist=bin type1;
run;
```

PGM

OUTPUT

```
                    LR Statistics For Type 1 Analysis

              Source       Deviance    DF   ChiSquare  Pr>Chi

              INTERCEPT    151.0193     0           .       .
```

OUTPUT

Ce plan est équilibré : le nombre de répétitions par combinaison vaut 240. En modèle linéaire, un plan équilibré est un plan orthogonal : on a orthogonalité entre les sous-espaces vectoriels dans lesquels on projette le vecteur des observations. Les résultats (les sommes de carrés) ne dépendent pas de l'ordre d'introduction des termes dans le modèle, ils ne dépendent pas de l'ordre des projections successives.
En GLM, ça n'est pas le cas. Même si le plan est équilibré, on n'a pas orthogonalité. Les déviances dépendent de l'ordre d'introduction des termes dans le modèle.
Ainsi l'effet `longueur`, en ignorant l'effet `periode`, est associé à une déviance égale à 45.84 (Modèle 2). L'effet `longueur`, ajusté par rapport à l'effet `periode`, est associé à une déviance égale à 51.15 (Modèle 4).

Exemple ESR
Utilisation de l'exemple `ESR` (exemple (7), page 7). Cet exemple concerne des données relatives au taux de sédimentation de globules rouges. Un taux de sédimentation inférieur à un certain seuil est un signe de bonne santé (Y=1), alors qu'un taux supérieur au même seuil est un indicateur d'un problème (Y=0).
On cherche à savoir si la variable Y ainsi définie peut être expliquée par les variables fibrinogen et γ-globulin.

PGM

```
Data ESR;
 do ind=1 to 32;
    input Fibri gammaG reponse @@;
    n=1;
    output;
 end;
cards;
2.52 38 0     2.56 31 0      2.19 33 0      2.18 31 0      3.41 37 0
2.46 36 0     3.22 38 0      2.21 37 0      3.15 39 0      2.60 41 0
2.29 36 0     2.35 29 0      5.06 37 1      3.34 32 1      2.38 37 1
3.15 36 0     3.53 46 1      2.68 34 0      2.60 38 0      2.23 37 0
2.88 30 0     2.65 46 0      2.09 44 1      2.28 36 0      2.67 39 0
2.29 31 0     2.15 31 0      2.54 28 0      3.93 32 1      3.34 30 0
2.99 36 0     3.32 35 0
;
run;
```

PGM

Attention : ici, les données sont binaires. La déviance ne peut donc pas être utilisée pour apprécier l'adéquation du modèle.

PGM

```
proc Genmod data=ESR;
 model reponse/n=Fibri gammaG/dist=bin type1 type3;
run;
```

PGM

OUTPUT

```
                 LR Statistics For Type 1 Analysis

         Source       Deviance    DF   ChiSquare  Pr>Chi

         INTERCEPT     30.8850     0           .       .
         FIBRI         24.8404     1      6.0446  0.0139
         GAMMAG        22.9711     1      1.8692  0.1716
```

```
                LR Statistics For Type 3 Analysis

                Source       DF   ChiSquare  Pr>Chi

                FIBRI         1      5.9743  0.0145
                GAMMAG        1      1.8692  0.1716
```
OUTPUT

La probabilité que le taux de sédimentation dépasse le seuil fixé ne semble dépendre que de la variable `fibri`. Le modèle logistique ne faisant intervenir que la variable `fibri` est ajusté.

PGM

```
proc Genmod data=ESR;
 make 'obstats' out=Result;
 model reponse/n=Fibri/dist=bin corrb obstats;
run;
```
PGM

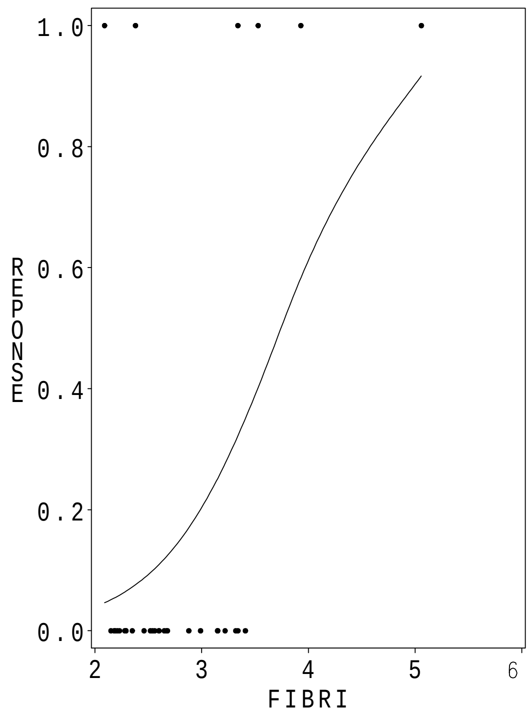
OUTPUT

```
               Criteria For Assessing Goodness Of Fit

        Criterion              DF          Value       Value/DF

        Deviance               30        24.8404         0.8280
        Scaled Deviance        30        24.8404         0.8280
        Pearson Chi-Square     30        41.8135         1.3938
        Scaled Pearson X2      30        41.8135         1.3938
        Log Likelihood          .       -12.4202              .


                  Analysis Of Parameter Estimates

     Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

     INTERCEPT     1     -6.8451      2.7703      6.1053  0.0135
     FIBRI         1      1.8271      0.9009      4.1134  0.0425
     SCALE         0      1.0000      0.0000           .       .
```
OUTPUT

Les représentations graphiques sont obtenues avec le programme suivant :

PGM

```
Data Result;
 merge ESR Result;
run;
Proc Gplot data=Result;
 symbol1 v=dot c=black;
 symbol2 v=none i=splines c=black;
 axis1 offset=(2,2) order=0 to 1 by .2;
 axis2 order=2 to 6 by 1;
 plot yvar1*fibri=1 pred*fibri=2 /overlay hm=0 vm=0 haxis=axis2 vaxis=axis1 frame;
run;
quit;
```
PGM



3.10 Linear trends in proportion

On utilise l'exemple `Insecte` (exemple (10), page 9), sur la résistance d'un papillon de nuit au pyrethroid trans-cypermethrin.
Dans le premier modèle, on considère la variable `dose` comme un facteur à 6 niveaux.

PGM
```
proc Genmod data=insecte;
 class sexe dose;
 model k/n=sexe dose /dist=bin type1 type3;
run;
```
PGM

OUTPUT
```
               LR Statistics For Type 1 Analysis

       Source       Deviance    DF   ChiSquare  Pr>Chi

       INTERCEPT    124.8756     0           .       .
       SEXE         118.7986     1      6.0770  0.0137
       DOSE           5.0128     5    113.7858  0.0001

               LR Statistics For Type 3 Analysis

             Source       DF   ChiSquare  Pr>Chi

             SEXE          1     10.1395  0.0015
             DOSE          5    113.7858  0.0001
```
OUTPUT

On a considéré la dose comme un facteur dont les niveaux sont non ordonnés. C'est dommage : on n'a pas tenu compte de toute l'information dont on dispose et en particulier du fait que les niveaux sont ordonnés. La dose est en fait une variable explicative quantitative.

PGM

```
proc Genmod data=insecte;
 class sexe;
 model k/n=sexe dose /dist=bin type1;
run;
```
PGM

OUTPUT

```
                  LR Statistics For Type 1 Analysis

             Source       Deviance    DF   ChiSquare  Pr>Chi

             INTERCEPT    124.8756     0           .       .
             SEXE         118.7986     1      6.0770  0.0137
             DOSE          27.9680     1     90.8306  0.0001
```
OUTPUT

On peut inclure dans le modèle des effets linéaires (dans l'échelle du lien), quadratiques, ou d'ordre plus élevé .

PGM

```
proc Genmod data=insecte;
 class sexe;
 model k/n=sexe dose dose*dose dose*dose*dose dose*dose*dose*dose
                dose*dose*dose*dose*dose /dist=bin type1;
run;
```
PGM

OUTPUT

```
                    LR Statistics For Type 1 Analysis

          Source                 Deviance    DF   ChiSquare  Pr>Chi

          INTERCEPT              124.8756     0           .       .
          SEXE                   118.7986     1      6.0770  0.0137
          DOSE                    27.9680     1     90.8306  0.0001
          DOSE*DOSE               14.8352     1     13.1328  0.0003
          DOSE*DOSE*DOSE           8.1801     1      6.6551  0.0099
          DOSE*DOSE*DOSE*DOSE      5.5152     1      2.6649  0.1026
          DOS*DOS*DOS*DOS*DOSE     5.0128     1      0.5024  0.4785
```
OUTPUT

Il faut aller jusqu'à un polynôme de degré 3.
Les doses sont des puissances de 2. Si on transforme la variable explicative `dose`, en considérant le log base 2 de la dose, un polynôme de degré 1 est suffisant. L'écart à la linéarité est non significatif.

PGM

```
Data Insecte;
 set Insecte;
 ldose=log2(dose);
run;
proc Genmod data=insecte;
 class sexe;
 model k/n=sexe ldose /dist=bin type1;
run;
```
PGM

OUTPUT

```
                LR Statistics For Type 1 Analysis

          Source       Deviance    DF   ChiSquare  Pr>Chi

          INTERCEPT    124.8756     0           .       .
          SEXE         118.7986     1      6.0770  0.0137
          LDOSE          6.7571     1    112.0415  0.0001

                 LR Statistics For Type 3 Analysis

                Source       DF   ChiSquare  Pr>Chi

                SEXE          1     10.2270  0.0014
                LDOSE         1    112.0415  0.0001
```

OUTPUT

3.11 Comparing stimulus-response relationships

Utilisation de l'exemple `Role` (exemple (8), page 7). L'exemple utilisé est celui du rôle de la femme dans la société, ou plus précisément de l'attitude vis à vis de la femme au foyer. Ces données sont issues d'enquêtes d'opinion.
Le premier modèle s'écrit :

$$\log(p/(1-p)) = \alpha_j + \beta_j \text{educ}$$

avec $j = 1$ pour les hommes et $j = 2$ pour les femmes.

PGM

```
Proc genmod data=role;
 class sexe;
 model accord/n= sexe educ(sexe) /dist=bin noint type1;
run;
```

PGM

OUTPUT

```
                  Criteria For Assessing Goodness Of Fit

            Criterion              DF         Value      Value/DF

            Deviance               37       57.1027        1.5433
            Scaled Deviance        37       57.1027        1.5433
            Pearson Chi-Square     37       72.7432        1.9660
            Scaled Pearson X2      37       72.7432        1.9660
            Log Likelihood          .    -1670.7128             .

                      Analysis Of Parameter Estimates

    Parameter             DF     Estimate     Std Err    ChiSquare  Pr>Chi

    INTERCEPT              0       0.0000      0.0000            .       .
    SEXE        femmes     1       3.0029      0.2724     121.5467  0.0001
    SEXE        hommes     1       2.0982      0.2355      79.3808  0.0001
    EDUC(SEXE)  femmes     1      -0.3154      0.0236     177.9016  0.0001
    EDUC(SEXE)  hommes     1      -0.2340      0.0202     134.3805  0.0001
```

```
   SCALE                 0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                     LR Statistics For Type 1 Analysis

              Source         Deviance    DF   ChiSquare  Pr>Chi

              SEXE           451.7105     0           .       .
              EDUC(SEXE)      57.1027     2    394.6078  0.0001
```
OUTPUT

Bien que la déviance soit grande par rapport au nombre de degrés de liberté, il est intéressant de la comparer à celle que l'on obtient en imposant l'hypothèse d'égalité des pentes entre hommes et femmes.
Le modèle 2 suppose l'égalité des pentes pour les 2 sexes : $\beta_1 = \beta_2 = \beta$.

$$\log(p/(1-p)) = \alpha_j + \beta \text{educ}$$

PGM

```
Proc genmod data=role;
 class sexe;
 model accord/n= sexe educ/dist=bin noint type1;
run;
```
PGM

OUTPUT

```
                   Criteria For Assessing Goodness Of Fit

            Criterion             DF          Value      Value/DF

            Deviance              38        64.0066        1.6844
            Scaled Deviance       38        64.0066        1.6844
            Pearson Chi-Square    38        72.6706        1.9124
            Scaled Pearson X2     38        72.6706        1.9124
            Log Likelihood         .     -1674.1648             .

                     Analysis Of Parameter Estimates

    Parameter           DF    Estimate     Std Err   ChiSquare  Pr>Chi

    INTERCEPT            0      0.0000      0.0000           .       .
    SEXE       femmes    1      2.4979      0.1828    186.7716  0.0001
    SEXE       hommes    1      2.5094      0.1839    186.2180  0.0001
    EDUC                 1     -0.2706      0.0154    308.3366  0.0001
    SCALE                0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                     LR Statistics For Type 1 Analysis

              Source        Deviance    DF   ChiSquare  Pr>Chi

              SEXE          451.7105     0           .       .
              EDUC           64.0066     1    387.7038  0.0001
```
OUTPUT

La différence entre les 2 modèles est hautement significative, il est nécessaire de distinguer les 2 pentes.

PGM

```
Proc genmod data=role;
 class sexe;
 model accord/n= sexe educ educ(sexe)/dist=bin noint type1;
run;
```

PGM

OUTPUT

```
                 LR Statistics For Type 1 Analysis

            Source        Deviance    DF   ChiSquare  Pr>Chi

            SEXE          451.7105     0           .       .
            EDUC           64.0066     1    387.7038  0.0001
            EDUC(SEXE)     57.1027     1      6.9039  0.0086
```

OUTPUT

La deviance du modèle avec des pentes différentes selon le sexe est élevée. On recherche donc un défaut d'ajustement. Le modèle est adéquat pour les hommes. Il est beaucoup moins bon pour les femmes.

PGM

```
Proc genmod data=role(where=(sexe='hommes'));
 make 'obstats' out=glm14h;
 model accord/n= educ /dist=bin corrb type1 obstats;
run;
```

PGM

OUTPUT

```
               Criteria For Assessing Goodness Of Fit

          Criterion             DF          Value      Value/DF

          Deviance              19        18.9450        0.9971
          Scaled Deviance       19        18.9450        0.9971
          Pearson Chi-Square    19        19.4072        1.0214
          Scaled Pearson X2     19        19.4072        1.0214
          Log Likelihood         .      -767.9222             .

                  Analysis Of Parameter Estimates

      Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

      INTERCEPT     1      2.0982      0.2355     79.3807  0.0001
      EDUC          1     -0.2340      0.0202    134.3804  0.0001
      SCALE         0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                    Estimated Correlation Matrix

                   Parameter
                   Number           PRM1     PRM2
```

```
                    PRM1          1.0000  -0.9651
                    PRM2         -0.9651   1.0000

                  LR Statistics For Type 1 Analysis

           Source       Deviance    DF   ChiSquare  Pr>Chi

           INTERCEPT    182.9219    0           .       .
           EDUC          18.9450    1    163.9768  0.0001
```
OUTPUT

PGM

```
Proc genmod data=role(where=(sexe='femmes'));
 make 'obstats' out=glm14f;
 model accord/n= educ /dist=bin corrb type1 obstats;
run;
```
PGM

OUTPUT

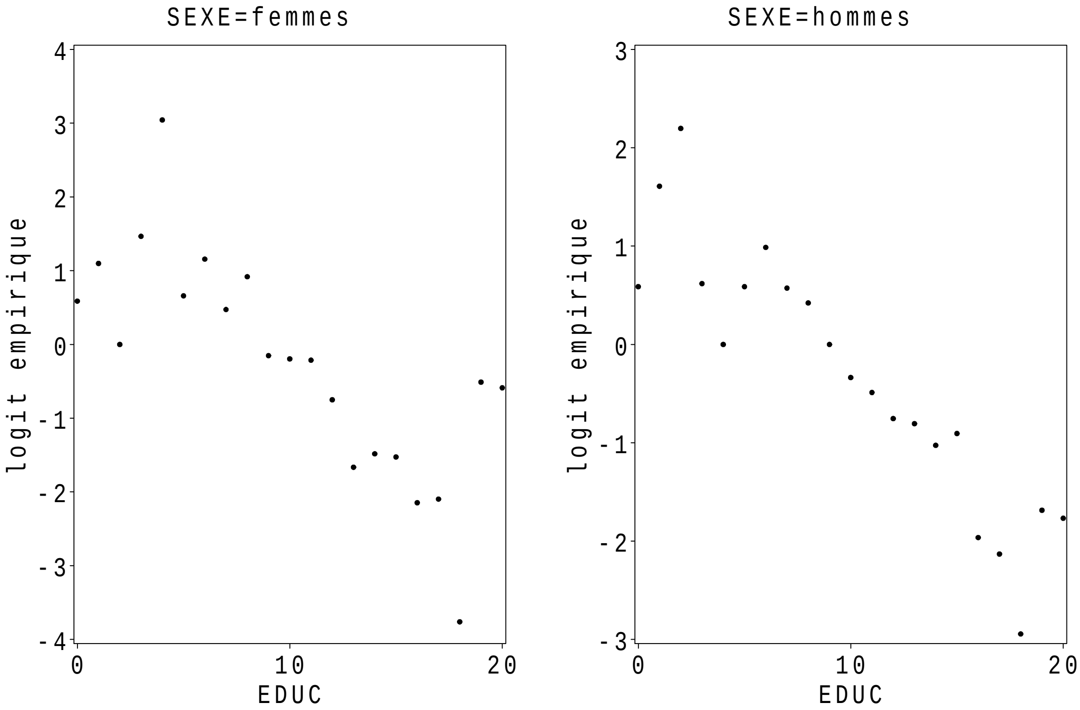
```
                 Criteria For Assessing Goodness Of Fit

            Criterion             DF          Value      Value/DF

            Deviance              18        38.1577        2.1199
            Scaled Deviance       18        38.1577        2.1199
            Pearson Chi-Square    18        53.3360        2.9631
            Scaled Pearson X2     18        53.3360        2.9631
            Log Likelihood         .      -902.7906             .

                   Analysis Of Parameter Estimates

        Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

        INTERCEPT     1      3.0029      0.2724    121.5467  0.0001
        EDUC          1     -0.3154      0.0236    177.9016  0.0001
        SCALE         0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                    Estimated Correlation Matrix

                    Parameter
                    Number           PRM1     PRM2

                    PRM1          1.0000  -0.9779
                    PRM2         -0.9779   1.0000

                  LR Statistics For Type 1 Analysis

           Source       Deviance    DF   ChiSquare  Pr>Chi

           INTERCEPT    268.7886    0           .       .
           EDUC          38.1577    1    230.6310  0.0001
```
OUTPUT

Pour mieux comprendre et tenter d'expliquer le fait que le modèle soit adéquat pour les hommes et qu'il ne le soit beaucoup moins pour les femmes, on représente graphiquement les données dans l'échelle logit, en appliquant la transformation logit empirique aux réponses. En toute logique, ce graphique aurait dû être réalisé avant le premier ajustement.

PGM

```
Data Rolem;
 set role;
 p=log((accord+0.5)/(desacc+0.5));
run;
Proc sort data=Rolem;
 by sexe;
run;
Proc Gplot data=rolem;
 label p='logit empirique';
 plot p*educ=1 /frame hm=0 vm=0;
 by sexe;
run;
quit;
```

PGM



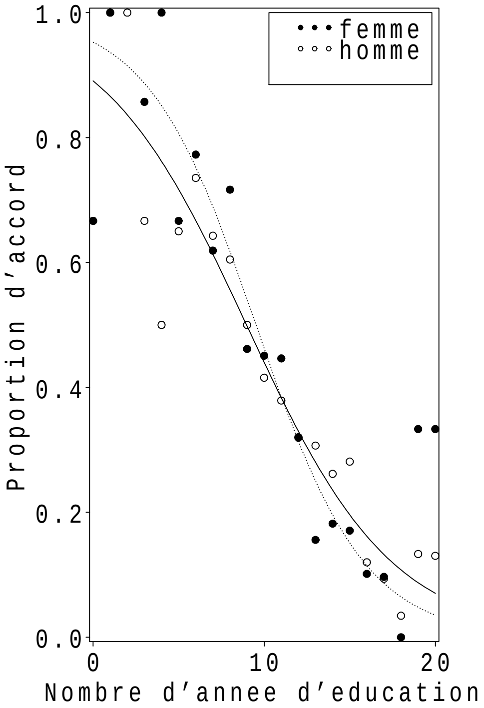
L'hypothèse de linéarité de l'effet "nombre d'années d'éducation" semble plausible pour les hommes et les femmes. Les graphiques ne suggèrent aucune modification de la variable explicative. La valeur élevée de la déviance, pour les femmes, est due aux valeurs extrêmes prises par la variable nombre d'années d'éducation.
Ces observations correspondent aux effectifs faibles.
On conserve le modèle incluant une pente différente pour les sexes.

PGM

```
Proc genmod data=role;
 make 'obstats' out=glm;
 class sexe;
 model accord/n= sexe educ(sexe) /dist=bin noint obstats;
run;
Data Glm;
 merge Role glm;
 p=accord/n;
```

```
run;
Proc sort data=Glm; by sexe; run;
legend1 down=3 frame offset=(-1,)
        position=(top right inside) label=('')
        value=('femme' 'homme');
Proc Gplot data=GLM;
 symbol1 v=dot c=black h=2pct; symbol3 v=none i=splines c=black l=33;
 symbol2 v=circle c=black h=2pct; symbol4 v=none i=splines c=black;
 axis1 label=(a=90 "Proportion d'accord") order=0 to 1 by .2;
 axis2 label=("Nombre d'annee d'education");
 plot p*educ=sexe /frame hm=0 vm=0 vaxis=axis1 haxis=axis2 legend=legend1;
 plot2 pred*educ=sexe /noaxis nolegend;
run; quit;
```

PGM



3.12 Non convergence and overfitting

On rencontre ce problème lorsque l'une au moins des proportions observées est exactement égale à 0 ou à 1. Le problème arrive donc lorsque les données sont "creuses" ("sparse"), lorsque le modèle est très complexe par rapport à la quantité de données, lorsque le modèle est surparamétré.
Les symptômes sont :

- un message de non convergence;
- des écarts-type pour certains paramètres très grands.

Cela suggère un modèle trop complexe.

PGM

```
Data bidon;
 input fact$ k n;
cards;
2 0 4
1 5 6
```

```
;
run;
proc genmod data=bid;
 class fact;
 model k/n=fact /dist=bin type1 maxit=10;
run;
```
PGM

OUTPUT

```
                 Criteria For Assessing Goodness Of Fit

           Criterion              DF          Value       Value/DF

           Deviance                0         0.0000              .
           Scaled Deviance         0         0.0000              .
           Pearson Chi-Square      0         0.0000              .
           Scaled Pearson X2       0         0.0000              .
           Log Likelihood          .        -2.7034              .

                    Analysis Of Parameter Estimates

      Parameter       DF    Estimate     Std Err   ChiSquare  Pr>Chi

      INTERCEPT        1    -13.7037    472.8123      0.0008  0.9769
      FACT       1     1     15.3131    472.8136      0.0010  0.9742
      FACT       2     0      0.0000      0.0000           .       .
      SCALE            0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                   LR Statistics For Type 1 Analysis

             Source       Deviance    DF   ChiSquare  Pr>Chi

             INTERCEPT      8.4562     0           .       .
             FACT           0.0000     1      8.4562  0.0036
```
OUTPUT

3.13 A further example on model selection

On travaille avec l'exemple `Prostate` (exemple (9), page 8).

PGM

```
Proc univariate data=Prostate plot;
 var acide lacide;
run;
```
PGM

La distribution dissymétrique de la variable acide suggère l'utilisation de log(acide) à la place de la variable acide, comme régresseur.

OUTPUT

Variable=ACIDE

```
                Moments

 N                53  Sum Wgts          53
 Mean       0.694151  Sum            36.79
```

```
Std Dev     0.262015  Variance   0.068652
Skewness    2.251881  Kurtosis    7.29481
USS          29.1077  CSS        3.569887
CV          37.74606  Std Mean    0.03599
T:Mean=0    19.28707  Pr>|T|       0.0001
Num ^= 0          53  Num > 0          53
M(Sign)         26.5  Pr>=|M|      0.0001
Sgn Rank       715.5  Pr>=|S|      0.0001

  Stem Leaf                     #  Boxplot
    18 7                        1     *
    17
    16
    15
    14
    13 6                        1     0
    12 6                        1     0
    11
    10 2                        1     |
     9 589                      3     |
     8 122349                   6     |
     7 001256688                9  +-----+
     6 22356777                 8  *--+--*
     5 0000012255669           13  +-----+
     4 0067888999              10     |
       ----+----+----+----+
   Multiply Stem.Leaf by 10**-1
```

Variable=LACIDE

```
                Moments

N                 53  Sum Wgts         53
Mean         -0.4189  Sum        -22.2017
Std Dev     0.315093  Variance   0.099283
Skewness    0.959626  Kurtosis   1.258185
USS         14.46301  CSS         5.16274
CV          -75.2192  Std Mean   0.043281
T:Mean=0    -9.67853  Pr>|T|       0.0001
Num ^= 0          53  Num > 0           4
M(Sign)        -22.5  Pr>=|M|      0.0001
Sgn Rank      -650.5  Pr>=|S|      0.0001

  Stem Leaf                     #  Boxplot
     6 3                        1     0
     5
     4
     3 1                        1     |
     2 3                        1     |
     1                                |
     0 2                        1     |
    -0 521                      3     |
    -1 972                      3     |
    -2 97755100                 8  +-----+
    -3 6643                     4  |     |
    -4 88632000                 8  *--+--*
```

```
 -5 883                     3  |     |
 -6 9999975500             10  +-----+
 -7 86333111                8     |
 -8                               |
 -9 22                      2     |
    ----+----+----+----+
Multiply Stem.Leaf by 10**-1
```
OUTPUT

Cela est confirmé par la comparaison des déviances des modèles associés.

PGM

```
Proc genmod data=prostate;
 model Y/n=acide /dist=bin corrb;
run;
```
PGM

OUTPUT

```
                 Criteria For Assessing Goodness Of Fit

           Criterion             DF          Value      Value/DF

           Deviance              51        67.1159        1.3160
           Scaled Deviance       51        67.1159        1.3160
           Pearson Chi-Square    51        54.5068        1.0688
           Scaled Pearson X2     51        54.5068        1.0688
           Log Likelihood         .       -33.5580         .
```
OUTPUT

PGM

```
Proc genmod data=prostate;
 model Y/n=lacide /dist=bin corrb;
run;
```
PGM

OUTPUT

```
                  Criteria For Assessing Goodness Of Fit

           Criterion             DF          Value      Value/DF

           Deviance              51        64.8127        1.2708
           Scaled Deviance       51        64.8127        1.2708
           Pearson Chi-Square    51        53.6795        1.0525
           Scaled Pearson X2     51        53.6795        1.0525
           Log Likelihood         .       -32.4064         .
```
OUTPUT

Néanmoins, si on supprime la 24^{e} observation (l'individu pour lequel la valeur observée de acide vaut 1.87), l'intérêt de la transformation log pour la variable acide n'est plus évident.

PGM

```
Proc genmod data=prostate(where=(acide^=1.87));
 model Y/n=acide /dist=bin corrb;
run;
```
PGM

OUTPUT

```
                Criteria For Assessing Goodness Of Fit

           Criterion              DF          Value      Value/DF

           Deviance               50        60.4524        1.2090
           Scaled Deviance        50        60.4524        1.2090
           Pearson Chi-Square     50        50.4534        1.0091
           Scaled Pearson X2      50        50.4534        1.0091
           Log Likelihood          .       -30.2262         .
```

OUTPUT

PGM

```
Proc genmod data=prostate(where=(acide^=1.87));
 model Y/n=lacide /dist=bin corrb;
run;
```

PGM

OUTPUT

```
               Criteria For Assessing Goodness Of Fit

           Criterion              DF          Value      Value/DF

           Deviance               50        59.7608        1.1952
           Scaled Deviance        50        59.7608        1.1952
           Pearson Chi-Square     50        50.7400        1.0148
           Scaled Pearson X2      50        50.7400        1.0148
           Log Likelihood          .       -29.8804         .
```

OUTPUT

Comme on n'a aucune raison médicale de supprimer le patient 24, on le conserve. D'autant plus que la distribution de la variable acide, pour la population qui a un nodule, semble dissymétrique, même sans l'individu 24 !
Le patient 24 est donc conservé, et on considère le log(acide) comme régresseur.
Il existe 32 modèles possibles pour ces données. On introduit les effets principaux, pas à pas (méthode ascendante).

PGM

```
Proc logistic data=prostate des;
 model Y/n=age lacide X taille grade/link=logit
        selection=score;
run;
```

PGM

OUTPUT

```
                   Regression Models Selected by Score Criterion

           Number of        Score
           Variables        Value   Variables Included in Model

                   1      11.2831   X
                   1       7.4383   TAILLE
                   1       5.3243   LACIDE
                   1       4.0746   GRADE
                   1       1.0936   AGE
        ------------------------------
                   2      15.7144   X TAILLE
                   2      14.2718   LACIDE X
```

```
                    2      13.1538   X GRADE
                    2      12.6224   LACIDE TAILLE
                    2      12.3452   AGE X
                    2      10.0377   LACIDE GRADE
                    2       8.5943   TAILLE GRADE
                    2       8.4229   AGE TAILLE
                    2       6.3275   AGE LACIDE
                    2       4.9768   AGE GRADE
          ------------------------------------
                    3      18.8925   LACIDE X TAILLE
                    3      16.6959   LACIDE X GRADE
                    3      16.6953   AGE X TAILLE
                    3      16.1718   X TAILLE GRADE
                    3      15.2686   AGE LACIDE X
                    3      14.1760   LACIDE TAILLE GRADE
                    3      14.0866   AGE X GRADE
                    3      13.5233   AGE LACIDE TAILLE
                    3      10.8395   AGE LACIDE GRADE
                    3       9.4890   AGE TAILLE GRADE
          -------------------------------------------
                    4      19.8071   AGE LACIDE X TAILLE
                    4      19.6371   LACIDE X TAILLE GRADE
                    4      17.5444   AGE LACIDE X GRADE
                    4      17.0961   AGE X TAILLE GRADE
                    4      14.9742   AGE LACIDE TAILLE GRADE
          --------------------------------------------------
                    5      20.4784   AGE LACIDE X TAILLE GRADE
          ----------------------------------------------------
```
OUTPUT

Il faudrait la table d'analyse de déviance, pour savoir jusqu'où on introduit les effets principaux. C'est ce qui est fait avec l'option `selection=f`.

PGM

```
Proc logistic data=prostate des;
 model Y/n=age lacide X taille grade/link=logit
        selection=f details;
run;
```
PGM

OUTPUT

```
                        Forward Selection Procedure

Step  0. Intercept entered:

                  Analysis of Maximum Likelihood Estimates

             Parameter Standard    Wald       Pr >    Standardized     Odds
 Variable DF  Estimate   Error  Chi-Square Chi-Square   Estimate      Ratio

 INTERCPT 1    -0.5008   0.2834     3.1229     0.0772              .      .

             Residual Chi-Square = 20.4784 with 5 DF (p=0.0010)

                           The LOGISTIC Procedure

                    Analysis of Variables Not in the Model
```

```
                          Score        Pr >
         Variable      Chi-Square    Chi-Square

         AGE              1.0936        0.2957
         LACIDE           5.3243        0.0210
         X               11.2831        0.0008
         TAILLE           7.4383        0.0064
         GRADE            4.0746        0.0435

Step  1. Variable X entered:

    Model Fitting Information and Testing Global Null Hypothesis BETA=0

                              Intercept
               Intercept         and
 Criterion       Only        Covariates    Chi-Square for Covariates

 AIC            72.252        63.001          .
 SC             74.222        66.941          .
 -2 LOG L       70.252        59.001       11.251 with 1 DF (p=0.0008)
 Score            .             .          11.283 with 1 DF (p=0.0008)

                 Analysis of Maximum Likelihood Estimates

             Parameter Standard    Wald       Pr >    Standardized     Odds
 Variable DF  Estimate   Error  Chi-Square Chi-Square   Estimate      Ratio

 INTERCPT 1    -1.1701  0.3816     9.4033     0.0022           .     .
 X        1     2.1817  0.6975     9.7835     0.0018     0.547014    8.861

       Association of Predicted Probabilities and Observed Responses

                Concordant = 48.3%          Somers' D = 0.429
                Discordant =  5.5%          Gamma     = 0.797
                Tied       = 46.2%          Tau-a     = 0.205
                (660 pairs)                 c         = 0.714

            Residual Chi-Square = 11.6479 with 4 DF (p=0.0202)

                 Analysis of Variables Not in the Model

                          Score        Pr >
         Variable      Chi-Square    Chi-Square

         AGE              1.3523        0.2449
         LACIDE           3.7971        0.0513
         TAILLE           5.6394        0.0176
         GRADE            2.3710        0.1236

Step  2. Variable TAILLE entered:

    Model Fitting Information and Testing Global Null Hypothesis BETA=0

                              Intercept
               Intercept         and
```

```
Criterion       Only       Covariates    Chi-Square for Covariates

AIC              72.252        59.353         .
SC               74.222        65.264         .
-2 LOG L         70.252        53.353       16.899 with 2 DF (p=0.0002)
Score              .             .          15.714 with 2 DF (p=0.0004)

                 Analysis of Maximum Likelihood Estimates

            Parameter Standard    Wald       Pr >    Standardized     Odds
Variable DF  Estimate   Error  Chi-Square Chi-Square   Estimate      Ratio

INTERCPT 1    -2.0446   0.6100    11.2360     0.0008           .        .
X        1     2.1194   0.7468     8.0537     0.0045     0.531411    8.326
TAILLE   1     1.5883   0.7000     5.1479     0.0233     0.441942    4.895

      Association of Predicted Probabilities and Observed Responses

                Concordant = 68.8%          Somers' D = 0.580
                Discordant = 10.8%          Gamma     = 0.730
                Tied       = 20.5%          Tau-a     = 0.278
                (660 pairs)                 c         = 0.790

            Residual Chi-Square = 6.6755 with 3 DF (p=0.0830)

                  Analysis of Variables Not in the Model

                                 Score        Pr >
                  Variable      Chi-Square    Chi-Square

                  AGE               1.2678       0.2602
                  LACIDE            4.5808       0.0323
                  GRADE             0.5839       0.4448

Step  3. Variable LACIDE entered:

    Model Fitting Information and Testing Global Null Hypothesis BETA=0

                              Intercept
              Intercept         and
Criterion       Only       Covariates    Chi-Square for Covariates

AIC              72.252        56.986         .
SC               74.222        64.867         .
-2 LOG L         70.252        48.986       21.266 with 3 DF (p=0.0001)
Score              .             .          18.893 with 3 DF (p=0.0003)

                 Analysis of Maximum Likelihood Estimates

            Parameter Standard    Wald       Pr >    Standardized     Odds
Variable DF  Estimate   Error  Chi-Square Chi-Square   Estimate      Ratio

INTERCPT 1    -1.1994   0.7162     2.8046     0.0940           .        .
LACIDE   1     2.2922   1.1387     4.0520     0.0441     0.398208    9.897
X        1     2.0550   0.7976     6.6380     0.0100     0.515257    7.807
TAILLE   1     1.7638   0.7483     5.5562     0.0184     0.490771    5.834
```

```
         Association of Predicted Probabilities and Observed Responses

                   Concordant = 84.2%          Somers' D = 0.688
                   Discordant = 15.5%          Gamma     = 0.690
                   Tied       =  0.3%          Tau-a     = 0.329
                   (660 pairs)                 c         = 0.844

             Residual Chi-Square = 2.4067 with 2 DF (p=0.3002)

                       Analysis of Variables Not in the Model

                                        Score         Pr >
                       Variable      Chi-Square    Chi-Square

                       AGE               1.3006        0.2541
                       GRADE             1.2319        0.2670

NOTE: No (additional) variables met the 0.05 significance level for entry
      into the model.

                       Summary of Forward Selection Procedure

                       Variable    Number        Score         Pr >
               Step    Entered         In    Chi-Square    Chi-Square

                  1    X                1       11.2831        0.0008
                  2    TAILLE           2        5.6394        0.0176
                  3    LACIDE           3        4.5808        0.0323
```
OUTPUT

L'ajout de termes de degré 2, pour les variables age et acide, est-il nécessaire? non.

PGM

```
Proc genmod data=prostate;
 class X taille grade;
 model y/n=age lacide X taille grade lacide*lacide age*age/dist=bin type1;
run;
```
PGM

OUTPUT

```
                     Criteria For Assessing Goodness Of Fit

              Criterion               DF          Value      Value/DF

              Deviance                45        43.6212        0.9694
              Scaled Deviance         45        43.6212        0.9694
              Pearson Chi-Square      45        43.8590        0.9746
              Scaled Pearson X2       45        43.8590        0.9746
              Log Likelihood           .       -21.8106             .

                         Analysis Of Parameter Estimates

      Parameter            DF     Estimate      Std Err    ChiSquare  Pr>Chi

      INTERCEPT             1      42.6784      35.3872       1.4545  0.2278
      AGE                   1      -1.3373       1.2254       1.1910  0.2751
```

```
    LACIDE                1     1.0566      2.0587      0.2634  0.6078
    X               0     1    -2.1738      0.8686      6.2638  0.0123
    X               1     0     0.0000      0.0000           .       .
    TAILLE          0     1    -1.2847      0.8204      2.4525  0.1173
    TAILLE          1     0     0.0000      0.0000           .       .
    GRADE           0     1    -0.8793      0.8418      1.0911  0.2962
    GRADE           1     0     0.0000      0.0000           .       .
    LACIDE*LACIDE         1    -3.5314      3.0956      1.3014  0.2540
    AGE*AGE               1     0.0113      0.0106      1.1316  0.2874
    SCALE                 0     1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                    LR Statistics For Type 1 Analysis

           Source           Deviance    DF   ChiSquare  Pr>Chi

           INTERCEPT         70.2522     0           .       .
           AGE               69.1622     1      1.0899  0.2965
           LACIDE            63.6519     1      5.5104  0.0189
           X                 53.7806     1      9.8713  0.0017
           TAILLE            47.6807     1      6.0999  0.0135
           GRADE             46.5599     1      1.1208  0.2897
           LACIDE*LACIDE     44.8635     1      1.6964  0.1928
           AGE*AGE           43.6212     1      1.2423  0.2650
```
OUTPUT

Est-il nécessaire d'introduire des termes d'interaction ?
Deux termes d'interaction sont introduits : log(acide)*grade, taille*grade.
Chacun de ces termes apporte une explication supplémentaire significative. Remarque : pour bien faire, il faudrait tester l'ajout de tous les termes d'interaction possibles.

PGM

```
Proc genmod data=prostate;
 class X taille grade;
 model y/n=age lacide X taille grade taille*grade lacide*grade/dist=bin type3;
run;
```
PGM

OUTPUT

```
                 Criteria For Assessing Goodness Of Fit

           Criterion             DF          Value       Value/DF

           Deviance              45        35.3376         0.7853
           Scaled Deviance       45        35.3376         0.7853
           Pearson Chi-Square    45        39.0340         0.8674
           Scaled Pearson X2     45        39.0340         0.8674
           Log Likelihood         .       -17.6688              .

                     Analysis Of Parameter Estimates

    Parameter            DF     Estimate      Std Err   ChiSquare  Pr>Chi

    INTERCEPT             1      10.3901       4.7916      4.7020  0.0301
    AGE                   1      -0.0703       0.0739      0.9050  0.3414
    LACIDE                1      10.5615       6.1749      2.9254  0.0872
```

```
    X              0        1     -2.3345       1.0645       4.8093  0.0283
    X              1        0      0.0000       0.0000            .       .
    TAILLE         0        1      2.2557       1.9588       1.3262  0.2495
    TAILLE         1        0      0.0000       0.0000            .       .
    GRADE          0        1     -3.1737       3.2486       0.9544  0.3286
    GRADE          1        0      0.0000       0.0000            .       .
    TAILLE*GRADE   0  0     1     -5.5352       2.3694       5.4574  0.0195
    TAILLE*GRADE   0  1     0      0.0000       0.0000            .       .
    TAILLE*GRADE   1  0     0      0.0000       0.0000            .       .
    TAILLE*GRADE   1  1     0      0.0000       0.0000            .       .
    LACIDE*GRADE   0        1     -8.7156       6.3718       1.8710  0.1714
    LACIDE*GRADE   1        0      0.0000       0.0000            .       .
    SCALE                   0      1.0000       0.0000            .       .

NOTE:  The scale parameter was held fixed.

                     LR Statistics For Type 3 Analysis

                   Source         DF   ChiSquare  Pr>Chi

                   AGE             1      0.9496  0.3298
                   LACIDE          1      8.4945  0.0036
                   X               1      5.7665  0.0163
                   TAILLE          1      0.2015  0.6535
                   GRADE           1      5.7822  0.0162
                   TAILLE*GRADE    1      7.1805  0.0074
                   LACIDE*GRADE    1      3.0934  0.0786
```
OUTPUT

Faut-il ajouter d'autres interactions d'ordre 2? d'ordre 3? non.

PGM

```
Proc genmod data=prostate;
 class X taille grade;
 model y/n=age|lacide|X|taille|grade@2/dist=bin type3;
run;
```
PGM

OUTPUT

```
                      Analysis Of Parameter Estimates

   Parameter             DF    Estimate     Std Err    ChiSquare  Pr>Chi

   INTERCEPT              1     24.1082     30.8117       0.6122  0.4340
   AGE                    1     -0.3069      0.4882       0.3953  0.5295
   LACIDE                 1     41.2343     33.8025       1.4881  0.2225
   AGE*LACIDE             1     -0.5387      0.5459       0.9737  0.3237
   X              0       1      2.7271     22.7999       0.0143  0.9048
   X              1       0      0.0000      0.0000            .       .
   AGE*X          0       1     -0.1035      0.3726       0.0772  0.7811
   AGE*X          1       0      0.0000      0.0000            .       .
   LACIDE*X       0       1     -0.2144      5.0867       0.0018  0.9664
   LACIDE*X       1       0      0.0000      0.0000            .       .
   TAILLE         0       1     15.1812     18.7204       0.6576  0.4174
   TAILLE         1       0      0.0000      0.0000            .       .
   AGE*TAILLE     0       1     -0.1876      0.2822       0.4419  0.5062
   AGE*TAILLE     1       0      0.0000      0.0000            .       .
```

```
LACIDE*TAILLE  0       1      4.8986      4.3296       1.2801  0.2579
LACIDE*TAILLE  1       0      0.0000      0.0000            .       .
X*TAILLE       0  0    1      1.0190      3.5485       0.0825  0.7740
X*TAILLE       0  1    0      0.0000      0.0000            .       .
X*TAILLE       1  0    0      0.0000      0.0000            .       .
X*TAILLE       1  1    0      0.0000      0.0000            .       .
GRADE          0       1    -13.4603     13.9841       0.9265  0.3358
GRADE          1       0      0.0000      0.0000            .       .
AGE*GRADE      0       1      0.2012      0.2195       0.8395  0.3595
AGE*GRADE      1       0      0.0000      0.0000            .       .
LACIDE*GRADE   0       1     -7.7960      6.8962       1.2780  0.2583
LACIDE*GRADE   1       0      0.0000      0.0000            .       .
X*GRADE        0  0    1     -0.7376      3.5484       0.0432  0.8353
X*GRADE        0  1    0      0.0000      0.0000            .       .
X*GRADE        1  0    0      0.0000      0.0000            .       .
X*GRADE        1  1    0      0.0000      0.0000            .       .
TAILLE*GRADE   0  0    1     -7.3703      3.5286       4.3628  0.0367
TAILLE*GRADE   0  1    0      0.0000      0.0000            .       .
TAILLE*GRADE   1  0    0      0.0000      0.0000            .       .
TAILLE*GRADE   1  1    0      0.0000      0.0000            .       .
SCALE                  0      1.0000      0.0000            .       .

NOTE:  The scale parameter was held fixed.

                    LR Statistics For Type 3 Analysis

                  Source           DF   ChiSquare  Pr>Chi

                  AGE               1      1.4205  0.2333
                  LACIDE            1      1.4645  0.2262
                  AGE*LACIDE        1      0.9937  0.3188
                  X                 1      0.0163  0.8984
                  AGE*X             1      0.0767  0.7818

                    LR Statistics For Type 3 Analysis

                  Source           DF   ChiSquare  Pr>Chi

                  LACIDE*X          1      0.0018  0.9664
                  TAILLE            1      0.6220  0.4303
                  AGE*TAILLE        1      0.5533  0.4570
                  LACIDE*TAILLE     1      1.5095  0.2192
                  X*TAILLE          1      0.0859  0.7695
                  GRADE             1      2.4137  0.1203
                  AGE*GRADE         1      1.0182  0.3129
                  LACIDE*GRADE      1      1.4657  0.2260
                  X*GRADE           1      0.0459  0.8304
                  TAILLE*GRADE      1      7.9150  0.0049
```

OUTPUT

Est-il possible de supprimer des effets principaux, qui n'interviennent dans aucun terme d'interaction ? oui, on peut enlever la variable age.
Remarques : l'effet principal taille est non significatif. Mais il intervient dans un terme d'interaction. Donc, ne pas le supprimer du modèle.
Avec l'analyse de type 1, c'est l'effet grade qui est non significatif, alors que l'effet taille est significatif. Certaines des corrélations entre les paramètres sont donc non négligeables.
Ceci est lié au problème général de non orthogonalité en GLM.

Conclusion pratique : pour interpréter, utiliser le type 3, ou bien faire des tests de sous-modeles.

OUTPUT
```
Proc genmod data=prostate;
 class X taille grade;
 model y/n=lacide X taille grade taille*grade lacide*grade/dist=bin type3;
run;
```
OUTPUT

OUTPUT
```
                    Criteria For Assessing Goodness Of Fit

               Criterion             DF          Value      Value/DF

               Deviance              46        36.2871        0.7889
               Scaled Deviance       46        36.2871        0.7889
               Pearson Chi-Square    46        42.7826        0.9301
               Log Likelihood         .       -18.1436             .
               Scaled Pearson X2     46        42.7826        0.9301

                       Analysis Of Parameter Estimates

        Parameter           DF    Estimate     Std Err   ChiSquare  Pr>Chi

        INTERCEPT            1      7.2391      3.4133      4.4980  0.0339
        LACIDE               1     12.1345      6.5154      3.4686  0.0625
        X             0      1     -2.3404      1.0845      4.6571  0.0309
        X             1      0      0.0000      0.0000           .       .
        TAILLE        0      1      2.5098      2.0218      1.5410  0.2145
        TAILLE        1      0      0.0000      0.0000           .       .
        GRADE         0      1     -4.3134      3.2696      1.7404  0.1871
        GRADE         1      0      0.0000      0.0000           .       .
        TAILLE*GRADE  0  0   1     -5.6477      2.4346      5.3814  0.0204
        TAILLE*GRADE  0  1   0      0.0000      0.0000           .       .
        TAILLE*GRADE  1  0   0      0.0000      0.0000           .       .
        TAILLE*GRADE  1  1   0      0.0000      0.0000           .       .
        LACIDE*GRADE  0      1    -10.4260      6.6403      2.4652  0.1164
        LACIDE*GRADE  1      0      0.0000      0.0000           .       .
        SCALE                0      1.0000      0.0000           .       .

NOTE:  The scale parameter was held fixed.

                      LR Statistics For Type 3 Analysis

                    Source          DF   ChiSquare  Pr>Chi

                    LACIDE           1      9.4004  0.0022
                    X                1      5.6945  0.0170
                    TAILLE           1      0.0772  0.7812
                    GRADE            1      7.3800  0.0066
                    TAILLE*GRADE     1      6.8699  0.0088
                    LACIDE*GRADE     1      4.1672  0.0412
```
OUTPUT

3.14 Predicting a binary response probability

On veut prévoir la probabilité d'un nouvel individu : par exemple dans le cas précédent, on veut obtenir la probabilité pour un individu qui a les caractéristiques suivantes :

X=0, taille=1, grade=1, acide=0.7

PGM

```
Data new;
 input Y X taille grade acide @@;
 n=1;
 lacide=log(acide);
cards;
. 0 1 1 .7
. 1 1 1 .7
;
run;
Proc format;
 value invbin 0=1
              1=0;
 value spe 0=1
           1=2;
run;
Data Test;
 set prostate new;
run;
```

PGM

PGM

```
Proc genmod data=Test order=formatted;
 class X taille grade;
 format X taille grade invbin.;
 model y/n=lacide X taille grade taille*grade lacide*grade/
          dist=bin covb obstats;
run;
```

PGM

OUTPUT

OUTPUT

On voudrait déterminer une règle d'affectation d'un individu à l'un des deux groupes (avec ou sans nodule). La règle la plus simple consiste à affecter un individu au groupe "sans nodule" si la proba estimée est inférieure à 0.5, et au groupe "avec nodule" dans le cas contraire.
La régle d'affectation consiste donc à choisir une valeur seuil (0.5 ici).
Un choix plus astucieux de la valeur seuil consiste à étudier les proportions de mauvais classement, de faux positifs et de faux négatifs en fonction du seuil, et à choisir le seuil en tenant compte des probabilités de conclure à un faux positif ou à un faux negatif et des conséquences d'une conclusion erronée.
Ici, il est important de minimiser la proba associée à un faux négatif.
En principe, la PROC LOGISTIC fait ces calculs, mais les résultats sont faux. Peut-être un bug de PROC LOGISTIC? l'inconvénient de la PROC LOGISTIC, c'est qu'elle ne sait pas gérer les interactions. Il faut donc créer un nouveau terme par interaction.

PGM

```
Data test;
 set prostate new;
 tg=taille*grade;
 lg=lacide*grade;
run;
```

```
Proc logistic data=test(where=(y is not missing)) order=formatted;
 format X taille grade invbin.;
 model y/n= lacide X taille grade tg lg/covb
       link=logit ctable pprob=(0.25, .35, .45, .5, .55, .65);
run;
```

PGM

OUTPUT

```
   Model Fitting Information and Testing Global Null Hypothesis BETA=0

                              Intercept
               Intercept        and
 Criterion       Only       Covariates    Chi-Square for Covariates

 AIC            72.252        50.287          .
 SC             74.222        64.079          .
 -2 LOG L       70.252        36.287        33.965 with 6 DF (p=0.0001)
 Score            .             .           26.314 with 6 DF (p=0.0002)

                 Analysis of Maximum Likelihood Estimates

             Parameter Standard    Wald       Pr >    Standardized     Odds
 Variable DF  Estimate   Error  Chi-Square Chi-Square   Estimate      Ratio

 INTERCPT 1    -2.5527  1.0397     6.0282     0.0141            .        .
 LACIDE   1     1.7085  1.4204     1.4469     0.2290     0.296805    5.521
 X        1     2.3404  1.0845     4.6571     0.0309     0.586824   10.386
 TAILLE   1     3.1380  1.1707     7.1851     0.0074     0.873144   23.057
 GRADE    1     9.9612  4.6667     4.5562     0.0328     2.687528  999.000
 TG       1    -5.6477  2.4346     5.3814     0.0204    -1.416065    0.004
 LG       1    10.4260  6.6403     2.4652     0.1164     1.582686  999.000

      Association of Predicted Probabilities and Observed Responses

                 Concordant = 92.1%         Somers' D = 0.844
                 Discordant =  7.7%         Gamma     = 0.845
                 Tied       =  0.2%         Tau-a     = 0.404
                 (660 pairs)                c         = 0.922

                       Estimated Covariance Matrix

Variable         INTERCPT          LACIDE                X          TAILLE

INTERCPT     1.0809831493    0.4024219044     -0.569979768    -0.856552619
LACIDE       0.4024219044    2.0175039515     0.1721046484    0.3799217816
X            -0.569979768    0.1721046484     1.1762074337    0.5052864434
TAILLE       -0.856552619    0.3799217816     0.5052864434    1.3704476926
GRADE        -1.479530593    -0.282081023     1.3924203042     1.209864494
TG           1.1211675397    -0.459821912     -1.051344486    -1.605028531
LG           -1.022852202    -1.830165831     1.1082122499    0.1700890017

                          Classification Table

            Correct      Incorrect                Percentages
          ------------  ------------  ------------------------------------
   Prob          Non-          Non-          Sensi-  Speci-  False  False
```

```
Level  Event  Event  Event  Event  Correct  tivity  ficity   POS    NEG
-----------------------------------------------------------------------
0.250     16     22     11      4     71.7    80.0    66.7  40.7   15.4
0.350     15     26      7      5     77.4    75.0    78.8  31.8   16.1
0.450     12     28      5      8     75.5    60.0    84.8  29.4   22.2
0.500     10     28      5     10     71.7    50.0    84.8  33.3   26.3
0.550     10     28      5     10     71.7    50.0    84.8  33.3   26.3
0.650     10     30      3     10     75.5    50.0    90.9  23.1   25.0
```

OUTPUT

Chapitre 4

Bioassay and some other applications

4.1 The tolerance distribution

On utilise l'exemple `serum` (exemple (6), page 7). Cinq doses de serum anti-pneumocoque sont expérimentées. Pour chaque dose, on utilise un lot de 40 souris. Les souris succombent ou pas.

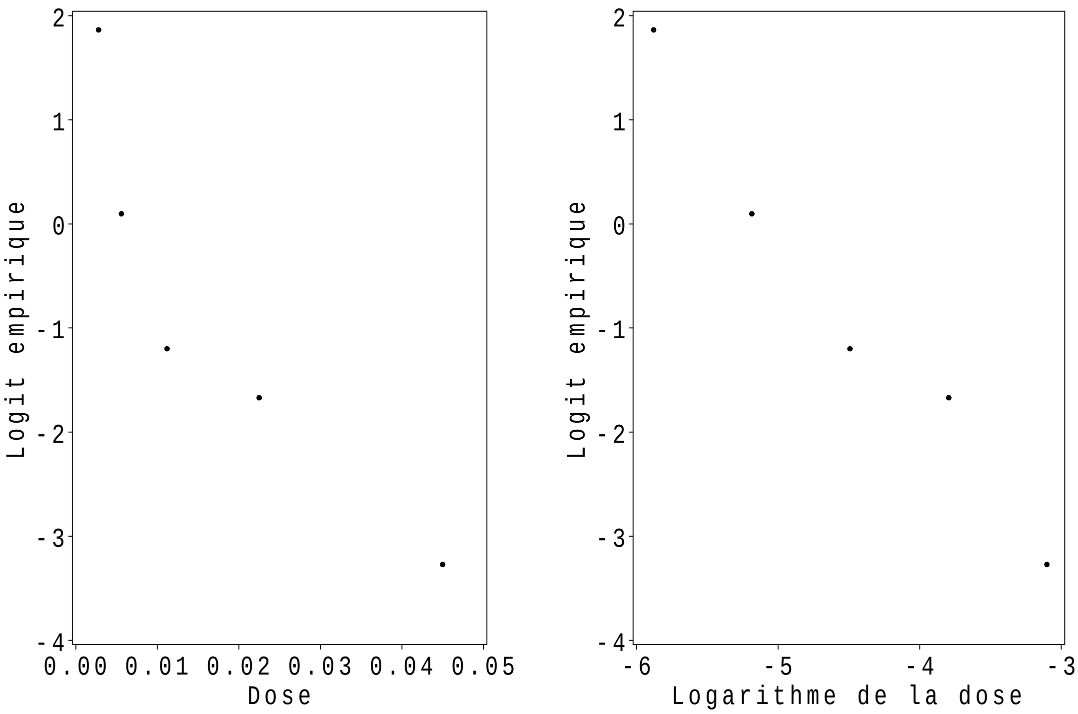
Les représentations graphiques des logit empiriques des proportions en fonction de la dose, ou de son logarithme montrent que l'hypothèse de linéarité est plus vraisemblable lorsqu'on choisit le log(dose) comme variable explicative.

PGM

```
Data serum;
 set serum;
 logit=log((y+.5)/(n-y+.5));
run;
Proc Gplot data=Serum;
 symbol1 v=dot i=none;
 axis1 label=(a=90 "Logit empirique") order=-4 to 2 by 1;
 axis2 label=("Dose") order=0 to 0.05 by 0.01;
 axis3 label=("Logarithme de la dose") order=-6 to -3 by 1;
 plot logit*dose=1  / frame vaxis=axis1 haxis=axis2 vm=0 hm=0;
 plot logit*ldose=1 / frame vaxis=axis1 haxis=axis3 vm=0 hm=0;
run;
quit;
```

PGM

On le confirme en ajustant chacun des deux modèles avec `dose` ou log(`dose`) comme régresseur. Avec log(`dose`), la déviance est de l'ordre de grandeur du degré de liberté résiduel, tandis qu'avec `dose`, la déviance est bien supérieure au degré de liberté. C'est donc le logarithme de la dose qu'il faut choisir.



PGM

```
Proc genmod data=serum;
 model y/n=dose/dist=bin;
run;
Proc genmod data=serum;
 model y/n=ldose/dist=bin;
run;
```

PGM

OUTPUT

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	3	15.9000	5.3000
Scaled Deviance	3	15.9000	5.3000
Pearson Chi-Square	3	16.3416	5.4472
Scaled Pearson X2	3	16.3416	5.4472
Log Likelihood	.	-93.6079	.

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT	1	1.2179	0.2930	17.2813	0.0001
DOSE	1	-146.6927	26.3630	30.9619	0.0001
SCALE	0	1.0000	0.0000	.	.

OUTPUT

OUTPUT

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF

```
Deviance                 3         2.8089         0.9363
Scaled Deviance          3         2.8089         0.9363
Pearson Chi-Square       3         2.9174         0.9725
Scaled Pearson X2        3         2.9174         0.9725
Log Likelihood           .       -87.0623              .

                 Analysis Of Parameter Estimates

Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

INTERCEPT     1     -9.1894      1.2551     53.6056  0.0001
LDOSE         1     -1.8296      0.2545     51.6638  0.0001
SCALE         0      1.0000      0.0000           .       .
```

OUTPUT

On réalise de nouveau les deux ajustements, mais avec un lien probit. Les déviances obtenues sont légèrement inférieures aux déviances des modèles logit correspondants mais la différence est très petite. On choisit un modèle logit.

PGM

```
Proc genmod data=serum;
 model y/n=dose/dist=bin link=probit;
run;
Proc genmod data=serum;
 model y/n=ldose/dist=bin link=probit;
run;
```

PGM

OUTPUT

```
              Criteria For Assessing Goodness Of Fit

Criterion               DF          Value       Value/DF

Deviance                 3        19.0910         6.3637
Scaled Deviance          3        19.0910         6.3637
Pearson Chi-Square       3        20.9316         6.9772
Scaled Pearson X2        3        20.9316         6.9772
Log Likelihood           .       -95.2034              .

                 Analysis Of Parameter Estimates

Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

INTERCEPT     1      0.6397      0.1651     15.0167  0.0001
DOSE          1    -75.0138     12.2349     37.5908  0.0001
SCALE         0      1.0000      0.0000           .       .
```

OUTPUT

OUTPUT

```
              Criteria For Assessing Goodness Of Fit

Criterion               DF          Value       Value/DF

Deviance                 3         3.1931         1.0644
Scaled Deviance          3         3.1931         1.0644
Pearson Chi-Square       3         3.1970         1.0657
```

```
        Scaled Pearson X2      3        3.1970        1.0657
        Log Likelihood         .      -87.2544             .

                     Analysis Of Parameter Estimates

Parameter    DF    Estimate     Std Err   ChiSquare  Pr>Chi

INTERCEPT     1     -5.2767      0.6441     67.1095  0.0001
LDOSE         1     -1.0545      0.1329     62.9669  0.0001
SCALE         0      1.0000      0.0000           .       .
```

OUTPUT

4.2 Estimating an effective dose

On souhaite estimer la DL_{50} (dose léthale) ou DE_{50} (dose efficace). Il s'agit de la dose qui provoque une réponse chez 50% des individus.
On travaille sur l'exemple `serum` (exemple (6), page 7).
On calcule la DE_{50} et l'intervalle de confiance pour la DE_{50}. Le premier intervalle de confiance est le résultat d'une approximation normale.

PGM

```
proc logistic data=serum outest=estimate
           (drop= _link_ _lnlike_) covout;
   model y/n = ldose /covb;
run;
 %let p=0.50;
options nosource;
Data _null_;
 set estimate end=fin;
 retain Ed v00 v01 v11 beta0 beta1 0;
 if _type_='PARMS' then do;
      logit=log(&p/(1-&p));
      beta0=intercep; beta1=ldose;
      Ed=(logit -beta0)/beta1;
    end;
 if _type_='COV' then do;
     if _name_='INTERCPT' then do; v00=intercep; v01=ldose;
    end;
     else v11=ldose;
 end;
if fin then do;
     var=(v00 + 2*Ed*v01 + Ed*Ed*v11)/(beta1*beta1);
     se=sqrt(var);
     Binf=Ed- sqrt(var)*1.96; Bsup=Ed + sqrt(var)*1.96;
     put "P= &p:     Ed = " Ed  "   SE = " se;
     put "Intervalle de confiance " Binf '< ' Ed '<' Bsup;
end;
run;
```

PGM

LOG

```
P= 0.50:     Ed = -5.022564949    SE = 0.1052285488
Intervalle de confiance -5.228812905 < -5.022564949 <-4.816316993
```

LOG

L'estimation de l'IC obtenu ci-dessus concerne la variable `ldose`. Pour obtenir l'intervalle de confiance de la DE_{50}, on peut utiliser l'intervalle $[e^{-5.228}, e^{-4.816}] = [0.0054, 0.0081]$.
Le deuxième IC est calculé à partir du théorème de Fieller.

PGM

```
%let k=-1,0; %let h=0,1;
proc iml;
  use estimate;
  start fieller;
  title 'Confidence Intervals';
  use estimate;
  read all into beta where (_type_='PARMS');
  beta=beta`;
  read all into cov where (_type_='COV');
  ratio=(&k`*beta) / (&h`*beta);
  a=(&h`*beta)**2-(3.84)*(&h`*cov*&h);
  b=2*(3.84*(&k`*cov*&h)-(&k`*beta)*(&h`*beta));
  c=(&k`*beta)**2 -(3.84)*(&k`*cov*&k);
   disc=((b**2)-4*a*c);
  if (disc<=0 | a<=0) then do;
  print "confidence interval can't be computed", ratio;
  stop; end;
  sroot=sqrt(disc);
   l_b=((-b)-sroot)/(2*a);
   u_b=((-b)+sroot)/(2*a);
  interval=l_b||u_b;
  lname="l_bound", "u_bound";
  print "Intervalle de confiance (base sur Fieller)", interval[colname=lname] ratio;
  finish fieller;
  run fieller;
quit iml;
```

PGM

OUTPUT

```
Intervalle de confiance (base sur Fieller)

INTERVAL   l_bound   u_bound     RATIO
         -5.249095  -4.81979 -5.022565
```

OUTPUT

Pour obtenir l'intervalle de confiance de la DE_{50}, on peut utiliser l'intervalle $[e^{-5.249}, e^{-4.820}] = [0.0053, 0.0081]$. Les deux intervalles sont quasiment identiques.
On peut faire la même chose pour la DE_{90}.

4.3 Relative potency

On travaille cette fois sur l'exemple `Beetle` (exemple (11), page 10). Des lots d'environ 50 insectes sont exposés à trois types d'insecticides à différentes doses. Le premier insecticide est à base de DDT, le deuxième à base de BHC et le troisième insecticide est un mélange des 2 premiers.
Trois modèles sont possibles. Le premier modèle fait l'hypothèse de droites différentes.

PGM

```
Proc Genmod data=Beetle;
 class insect;
 model y/n=insect ldose*insect /dist=bin noint;
run;
```

PGM

OUTPUT

```
                     Criteria For Assessing Goodness Of Fit

              Criterion              DF          Value      Value/DF

              Deviance               12        17.8897        1.4908
              Scaled Deviance        12        17.8897        1.4908
              Pearson Chi-Square     12        17.6109        1.4676
              Scaled Pearson X2      12        17.6109        1.4676
              Log Likelihood          .      -413.9844             .

                        Analysis Of Parameter Estimates

    Parameter               DF    Estimate     Std Err   ChiSquare  Pr>Chi

    INTERCEPT                0      0.0000      0.0000           .       .
    INSECT        BHC        1     -4.0428      0.4972     66.1070  0.0001
    INSECT        DDT        1     -3.9150      0.5094     59.0753  0.0001
    INSECT        DDT+BHC    1     -2.1207      0.5908     12.8848  0.0003
    LDOSE*INSECT  BHC        1      2.8381      0.3392     69.9995  0.0001
    LDOSE*INSECT  DDT        1      2.2779      0.3224     49.9231  0.0001
    LDOSE*INSECT  DDT+BHC    1      3.3884      0.5733     34.9263  0.0001
    SCALE                    0      1.0000      0.0000           .       .
```

OUTPUT

Le second modèle suppose l'égalité des pentes.

PGM

```
Proc Genmod data=Beetle;
 class insect;
 model y/n=insect ldose /dist=bin noint;
run;
```

PGM

OUTPUT

```
                     Criteria For Assessing Goodness Of Fit

              Criterion              DF          Value      Value/DF

              Deviance               14        21.2819        1.5201
              Scaled Deviance        14        21.2819        1.5201
              Pearson Chi-Square     14        20.3235        1.4517
              Scaled Pearson X2      14        20.3235        1.4517
              Log Likelihood          .      -415.6805             .

                        Analysis Of Parameter Estimates

      Parameter             DF    Estimate     Std Err   ChiSquare  Pr>Chi

      INTERCEPT              0      0.0000      0.0000           .       .
      INSECT     BHC         1     -3.8425      0.3327    133.4122  0.0001
      INSECT     DDT         1     -4.5553      0.3611    159.0966  0.0001
      INSECT     DDT+BHC     1     -1.4248      0.2851     24.9813  0.0001
      LDOSE                  1      2.6958      0.2157    156.2122  0.0001
      SCALE                  0      1.0000      0.0000           .       .
```

OUTPUT

Enfin, le dernier modèle suppose l'existence d'une droite unique.

PGM

```
Proc Genmod data=Beetle;
 class insect;
 model y/n=ldose/dist=bin;
run;
```

PGM

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

             Criterion              DF          Value       Value/DF

             Deviance               16       246.8321        15.4270
             Scaled Deviance        16       246.8321        15.4270
             Pearson Chi-Square     16       219.8400        13.7400
             Scaled Pearson X2      16       219.8400        13.7400
             Log Likelihood          .      -528.4556              .

                       Analysis Of Parameter Estimates

         Parameter    DF     Estimate     Std Err    ChiSquare  Pr>Chi

         INTERCEPT     1      -2.4653      0.2386     106.7232  0.0001
         LDOSE         1       2.0106      0.1703     139.4419  0.0001
         SCALE         0       1.0000      0.0000            .       .
```

OUTPUT

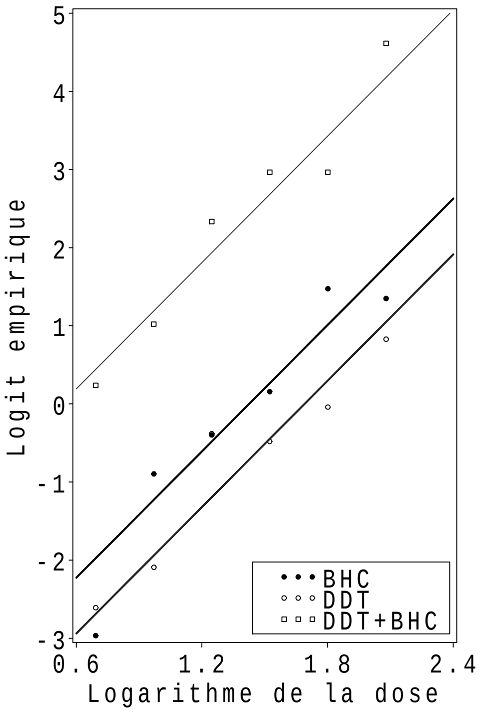
L'augmentation de déviance lorsqu'on impose l'égalité des pentes n'est pas significative. Par contre, l'augmentation de la déviance est significative, lorsqu'on impose l'égalité des ordonnées à l'origine.

Les droites (dans l'échelle du lien) sont donc parallèles, mais avec des ordonnées à l'origine différentes. On utilise le modèle avec des droites parallèles pour comparer l'efficacité des trois insecticides.

PGM

```
Proc Genmod data=Beetle;
 make 'obstats' out=result;
 class insect;
 model y/n=insect ldose /dist=bin noint obstats;
run;
Data Result;
 merge Beetle(keep=y n ldose Insect) Result(keep=xbeta);
 logit=log((y+.5)/(n-y+.5));
run;
```

PGM



PGM
```
legend1 down=3 frame offset=(-1,+1)
        position=(bottom right inside) label=('');
Proc Gplot data=result;
 symbol1 v=dot c=black;
 symbol2 v=circle c=black;
 symbol3 v=square c=black;
 symbol4 v=none i=rl r=2;
 axis1 label=(a=90 "Logit empirique") order=-3 to 5 by 1;
 axis2 label=("Logarithme de la dose") order=0.6 to 2.4 by .6;
 plot  logit*ldose=insect /frame haxis=axis2 vaxis=axis1 hm=0 vm=0 legend=legend1;
 plot2 xbeta*ldose=insect /noaxis nolegend;
run;
```
PGM

On calcule la DE_{50} pour chacun des trois types d'insecticide.
L'efficacité relative du mélange DDT+BHC, par rapport au DDT seul peut être estimée en calculant le rapport des DE_{50}. Le mélange est trois fois plus efficace que le DDT seul et 2.5 fois plus efficace que le BHC seul.

PGM
```
A FAIRE !!!!!
```
PGM

4.4 Natural response

On travaille sur le même exemple (`Beetle`). Dans ce jeux de données, il existe un groupe de contrôle qui n'a pas été exposé aux insecticides. Il permet d'estimer le taux de mortalité naturelle à 10%[1]. La fonction de lien utilisée est une extension du lien logit, prenant en compte la mortalité naturelle supposée connue et égale à 10%.

PGM

1. On ne retrouve pas tout à fait les mêmes résultats que Collett. Mais il y avait une erreur dans le tableau des données.

```
Proc Probit data=Beetle c=.10;
 class insect;
 model y/n=insect ldose /d=logistic noint;
run;
```
PGM

OUTPUT

```
Log Likelihood for LOGISTIC -418.2341575
Probit Procedure

Variable  DF   Estimate  Std Err ChiSquare  Pr>Chi Label/Value

INTERCPT   0          0        0         .    .     Intercept

INSECT     3                     155.0194  0.0001
           1 -4.7244101 0.447093  111.6605  0.0001 BHC
           1 -5.6343001 0.504852  124.5521  0.0001 DDT
           1 -1.9967951 0.339462  34.60065  0.0001 DDT+BHC

LDOSE      1 3.11348644 0.276373  126.9122  0.0001

C=0.1000
```
OUTPUT

Les DE_{50} obtenues en tenant compte de la mortalité naturelle sont supérieures à celles que l'on obtient lorsqu'on n'en tient pas compte.
On peut ajuster le même modèle indiquant qu'il existe une mortalité naturelle mais sans spécifier de valeur.

PGM

```
Proc Probit data=Beetle optc;
 class insect;
 model y/n=insect ldose /d=logistic;
run;
```
PGM

OUTPUT

```
Log Likelihood for LOGISTIC -416.5348325

Probit Procedure

Variable  DF   Estimate  Std Err ChiSquare  Pr>Chi Label/Value

INTERCPT   1 -1.2486908 0.359072  12.09334  0.0005 Intercept

INSECT     2                      104.313  0.0001
           1 -2.3341867  0.25729   82.3051  0.0001 BHC
           1 -2.9892155 0.303253  97.16378  0.0001 DDT
           0          0        0         .    .     DDT+BHC

LDOSE      1  2.5741029 0.257953   99.5793  0.0001
_C_        1          0 0.052369                   Lower threshold
```
OUTPUT

La valeur obtenue pour le taux de mortalité naturelle est égale à 0. Cela signifie que le jeu de données ne contient pas suffisamment d'information pour justifier la prise en compte d'une mortalité naturelle non nulle.

4.5 Non-linear logistic regression models

On travaille sur l'exemple CS2 (exemple (12), page 10). Des lots d'environ 30 insectes sont exposés à différentes doses de CS^2. On observe le nombre d'insectes tués. L'expérience est répétée deux fois. L'ajout d'un terme de degré 2 améliore significativement le modèle. L'ajout d'un terme de degré 3 au modèle précédent n'apporte pas d'amélioration significative.

PGM

```
Proc genmod data=CS2;
 model y/n=conc conc*conc conc*conc*conc/dist=bin type1;
run;
```

PGM

OUTPUT

```
                     Criteria For Assessing Goodness Of Fit

              Criterion              DF          Value        Value/DF

              Deviance               12         7.8816          0.6568
              Scaled Deviance        12         7.8816          0.6568
              Pearson Chi-Square     12         7.3950          0.6162
              Scaled Pearson X2      12         7.3950          0.6162
              Log Likelihood          .      -182.0906               .

                        Analysis Of Parameter Estimates

        Parameter         DF     Estimate     Std Err   ChiSquare  Pr>Chi

        INTERCEPT          1     -11.3278     91.8055      0.0152  0.9018
        CONC               1       0.4604      4.6328      0.0099  0.9208
        CONC*CONC          1      -0.0100      0.0774      0.0166  0.8974
        CONC*CONC*CONC     1       0.0001      0.0004      0.0446  0.8327
        SCALE              0       1.0000      0.0000           .       .

                       LR Statistics For Type 1 Analysis

              Source            Deviance    DF   ChiSquare  Pr>Chi

              INTERCEPT         289.1413     0           .       .
              CONC               12.5053     1    276.6360  0.0001
              CONC*CONC           7.9273     1      4.5779  0.0324
              CONC*CONC*CONC      7.8816     1      0.0458  0.8306
```

OUTPUT

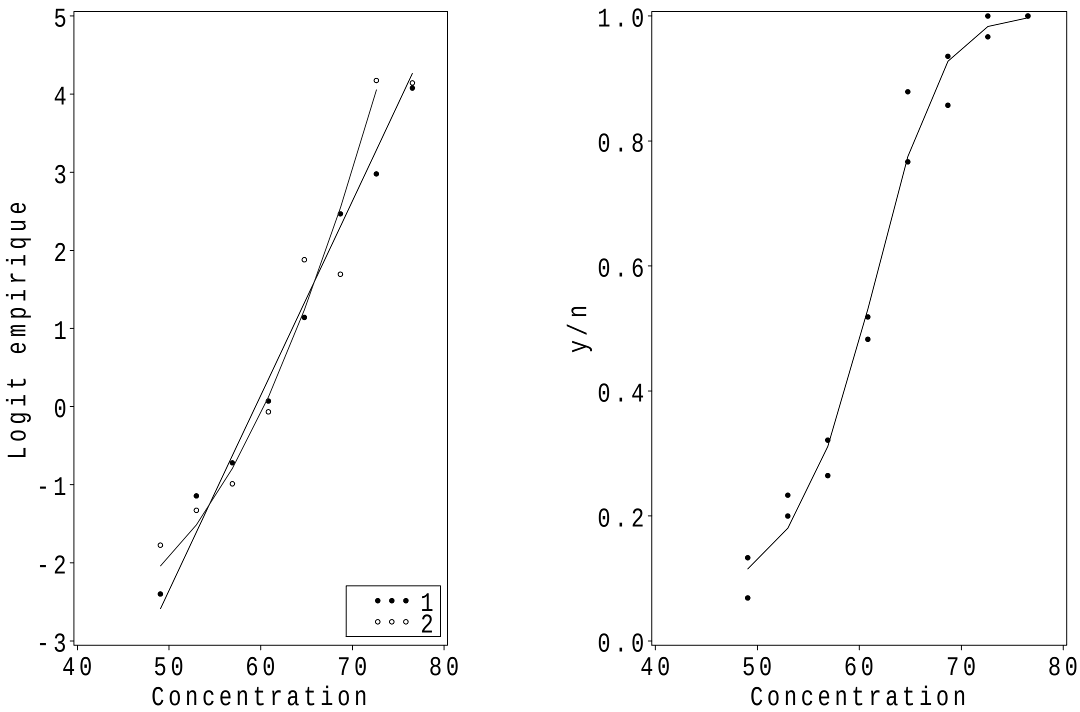
La représentation graphique et le test de sous-modèle sont en faveur de l'introduction d'un terme de degré 2, alors que la déviance du modèle de degré 1 conduit à accepter l'hypothèse d'adéquation du modèle de degré 1.

PGM

```
Proc genmod data=CS2;
 make 'obstats' out=result1(keep=xbeta) noprint;
 model y/n=conc/dist=bin obstat;
run;
Proc genmod data=CS2;
 make 'obstats' out=result2(keep=xbeta pred) noprint;
 model y/n=conc conc*conc/dist=bin obstat;
run;
```

```
Data Result;
 merge CS2 Result1(rename=(xbeta=Predl)) Result2(rename=(xbeta=Predq));
 logit=log((y+.5)/(n-y+.5));
 p=y/n;
run;
```

PGM



PGM

```
legend1 down=2 frame offset=(-1,+1)
        position=(bottom right inside) label=(”);
Proc Gplot data=result;
 symbol1 v=dot c=black;
 symbol2 v=circle c=black;
 symbol3 v=none i=join r=1;
 axis1 label=(a=90 "Logit empirique") order=-3 to 5 by 1;
 axis2 label=("Concentration") order=40 to 80 by 10;
 axis3 label=(a=90 "y/n") order=0 to 1 by .2;
 plot  logit*conc=rep/frame haxis=axis2 vaxis=axis1 hm=0 vm=0 legend=legend1;
 plot2 (predl predq)*conc /overlay haxis=axis2 vaxis=axis1 noaxis nolegend;
 plot p*conc=1 pred*conc=3 /overlay frame haxis=axis2 vaxis=axis3 hm=0 vm=0;
run;
```

PGM

4.6 Applications of the complementary log-log model

L'exemple utilisé est l'exemple `Malaria`. On dispose de la proportion d'individus atteints par classe d'âge.

PGM

```
Data Malaria;
 input age$ agem y n;
 lagem=log(agem);
datalines;
```

```
0-11mois    0.5   3    10
1-2ans      1.5   1    10
2-4ans      3.0   5    29
5-9ans      7.0   39   69
10-14ans   12.0   31   51
15-19ans   17.0   8    15
+20ans     30.0   91   108
;
run;
```

PGM

PGM

```
Proc genmod data=malaria;
 make 'parmest' out=param;
 model y/n= /dist=bin link=cloglog offset=lagem;
run;
```

PGM

On estime le taux d'infection annuel h par individu. On trouve 71 pour mille individus. On calcule l'écart-type associé et on obtient un intervalle de confiance pour h.

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

              Criterion             DF          Value       Value/DF

              Deviance               6        20.2533         3.3755
              Scaled Deviance        6        20.2533         3.3755
              Pearson Chi-Square     6        32.6445         5.4407
              Scaled Pearson X2      6        32.6445         5.4407
              Log Likelihood         .      -171.5918              .

                       Analysis Of Parameter Estimates

          Parameter    DF     Estimate     Std Err    ChiSquare  Pr>Chi

          INTERCEPT     1      -2.6068      0.0831     982.9760  0.0001
          SCALE         0       1.0000      0.0000            .       .
```

OUTPUT

PGM

```
Data _null_;
 set param(obs=1);
 h=1-exp(-exp(estimate));
 se=exp(estimate) * exp(-exp(estimate)) * stderr;
 binf=h-probit(.975)*se; bsup=h+probit(.975)*se;
 put "Intervalle de confiance  " binf h bsup;
run;
```

PGM

LOG

```
Intervalle de confiance  0.05994 0.07111 0.08228
```

LOG

Le modèle suppose un taux d'infection constant. Un écart à cette hypothèse conduit à un défaut d'adéquation du modèle. Le test d'adéquation, basé sur l'examen de la déviance revient donc à examiner dans quelle mesure les données sont compatibles avec l'hypothèse d'un taux d'infection constant. La déviance est significativement élevée. On rejette l'hypothèse.
On envisage donc un modèle avec un taux d'infection h qui dépend de la classe d'âge. Ce modèle est saturé.

PGM

```
Proc genmod data=malaria order=data;
 make 'parmest' out=param;
 class age;
 model y/n= age /dist=bin link=cloglog offset=lagem noint;
run;
```

PGM

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

              Criterion              DF          Value       Value/DF

              Deviance                0         0.0000              .
              Scaled Deviance         0         0.0000              .
              Pearson Chi-Square      0         0.0000              .
              Scaled Pearson X2       0         0.0000              .
              Log Likelihood          .      -161.4651              .

                       Analysis Of Parameter Estimates

     Parameter             DF    Estimate     Std Err    ChiSquare  Pr>Chi

     INTERCEPT              0      0.0000      0.0000            .       .
     AGE        0-11mois    1     -0.3378      0.5804       0.3387  0.5606
     AGE        1-2ans      1     -2.6558      1.0005       7.0469  0.0079
     AGE        2-4ans      1     -2.7633      0.4479      38.0665  0.0001
     AGE        5-9ans      1     -2.1287      0.1648     166.8578  0.0001
     AGE        10-14ans    1     -2.5509      0.1862     187.6202  0.0001
     AGE        15-19ans    1     -3.1048      0.3622      73.4931  0.0001
     AGE        +20ans      1     -2.7866      0.1204     535.5682  0.0001
     SCALE                  0      1.0000      0.0000            .       .
```

OUTPUT

PGM

```
Data Tx;
 merge param(firstobs=2 obs=8) malaria(keep=agem);
 h=1-exp(-exp(estimate));
 se=exp(estimate) * exp(-exp(estimate)) * stderr;
 binf=h-probit(.975)*se; bsup=h+probit(.975)*se;
 h=int(h*1000); binf=int(binf*1000); bsup=int(bsup*1000);
run;
Proc print data=Tx;
 var level1 h binf bsup;
```

PGM

OUTPUT

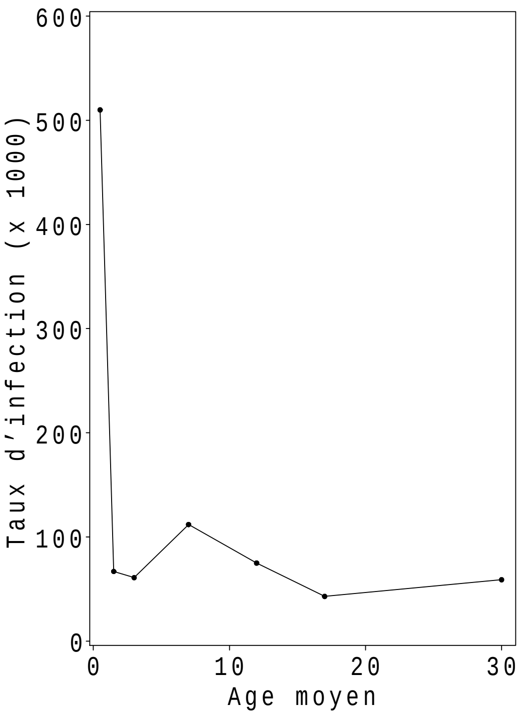
```
 LEVEL1      H     BINF    BSUP

0-11mois    510     112     907
```

```
1-2ans        67      -60      196
2-4ans        61        9      113
5-9ans       112       78      146
10-14ans      75       48      101
15-19ans      43       13       74
+20ans        59       46       73
```
OUTPUT

On représente les taux d'infection par classe d'âge.

PGM

```
Proc gplot data=Tx;
 symbol1 v=dot i=join;
 axis1 label=(a=90 "Taux d'infection (x 1000)") order=0 to 600 by 100;
 axis2 label=("Age moyen") order=0 to 30 by 10 offset=(,2);
 plot h*agem=1 / frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
run;
quit;
```
PGM



Chapitre 5

Model checking

5.1 Definition of residuals

On utilise l'exemple `serum` (exemple (6), page 7). Cinq doses de serum anti-pneumocoque sont expérimentées. Pour chaque dose, on utilise un lot de 40 souris. Les souris succombent ou pas.

PGM

```
Proc genmod data=serum;
 make 'obstats' out=result noprint;
 model y/n=ldose/dist=bin obstats residuals;
run;
Data result;
 merge serum(keep=dose y) result(drop=yvar1--upper);
run;
Proc Print data=result noobs;
run;
```

PGM

La PROC GENMOD fournit les résidus suivants :

resraw : résidu brut $y_i - n_i\hat{p}_i$

reschi : résidu de Pearson $X_i = \frac{y_i - n_i\hat{p}_i}{\sqrt{n_i\hat{p}_i(1-\hat{p}_i)}}$ avec $X^2 = \sum X_i$ la statistique de Pearson.

resdev : résidu de déviance

stresdev : résidu de déviance standardisé

streschi : résidu de Pearson standardisé $r_{Pi} = \frac{y_i - n_i\hat{p}_i}{\sqrt{\hat{v}_i(1-h_i)}}$

reslik : résidu de vraisemblance

OUTPUT

DOSE	Y	RESRAW	RESCHI	RESDEV	STRESDEV	STRESCHI	RESLIK
0.0028	35	1.91510	0.80077	0.83443	1.28313	1.23136	1.25351
0.0056	21	-1.95006	-0.62348	-0.62096	-0.80827	-0.81156	-0.80962
0.0112	9	-1.98707	-0.70389	-0.71856	-0.89804	-0.87971	-0.89148
0.0225	6	2.17693	1.17072	1.09010	1.38979	1.49258	1.43022
0.0450	1	-0.15490	-0.14627	-0.14963	-0.17496	-0.17103	-0.17391

OUTPUT

On travaille avec un autre exemple CS2 (exemple (12), page 10). Des lots d'environ 30 insectes sont exposés à différentes doses de CS^2. On observe le nombre d'insectes tués. L'expérience est répétée deux fois.

PGM

```
Proc genmod data=CS2;
 make 'obstats' out=result noprint;
 model y/n=conc/dist=bin obstats residuals;
run;
Data result;
 merge Cs2(keep=Conc rep y) result(drop=yvar1--upper);
run;
Proc Print data=result noobs;
run;
```

PGM

OUTPUT

CONC	REP	Y	RESRAW	RESCHI	RESDEV	STRESDEV	STRESCHI	RESLIK
49.06	1	2	-0.03481	-0.02531	-0.02537	-0.02730	-0.02723	-0.02729
49.06	2	4	1.89502	1.35453	1.21558	1.31139	1.46129	1.33351
52.99	1	7	1.98108	0.96906	0.92575	1.01357	1.06100	1.02159
52.99	2	6	0.98108	0.47990	0.46842	0.51286	0.52543	0.51497
56.91	1	9	-0.74201	-0.29440	-0.29625	-0.31890	-0.31691	-0.31863
56.91	2	9	-2.82958	-1.01880	-1.04165	-1.14086	-1.11584	-1.13674
60.84	1	14	-1.84578	-0.72141	-0.71675	-0.76179	-0.76674	-0.76236
60.84	2	14	-3.01954	-1.13875	-1.12901	-1.20575	-1.21615	-1.20704
64.76	1	23	-0.71446	-0.32052	-0.31629	-0.34047	-0.34503	-0.34110
64.76	2	29	2.91410	1.24649	1.32819	1.44121	1.35256	1.42821
68.69	1	29	0.80671	0.50492	0.52992	0.56882	0.54199	0.56535
68.69	2	24	-1.46491	-0.96477	-0.89702	-0.95585	-1.02804	-0.96475
72.61	1	29	0.08391	0.08210	0.08315	0.08735	0.08625	0.08725
72.61	2	32	1.15617	1.09522	1.53465	1.61787	1.15461	1.57758
76.54	1	29	0.40263	0.63898	0.90050	0.92707	0.65784	0.91397
76.54	2	31	0.43039	0.66065	0.93103	0.96049	0.68155	0.94598

OUTPUT

5.2 Checking the form of the linear predictor

On travaille avec un autre exemple lupin (exemple (13), page 10). La probabilité de germination de graines de lupin est étudiée en fonction de la température, du niveau d'humidité et de la température de stockage.

PGM

```
proc Genmod data=Lupin;
 make 'obstats' out=result noprint;
 class tp_germ moisir tp_cons;
 model y/n=tp_germ moisir tp_cons tp_cons*moisir / dist=bin obstats residuals;
run;
```

PGM

OUTPUT

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	8	4.4298	0.5537

```
Scaled Deviance         8        4.4298          0.5537
Pearson Chi-Square      8        4.3639          0.5455
Scaled Pearson X2       8        4.3639          0.5455
Log Likelihood          .      -586.4107              .

                    Analysis Of Parameter Estimates

Parameter                 DF    Estimate     Std Err   ChiSquare  Pr>Chi

INTERCEPT                  1     -4.7427      0.7158     43.9054  0.0001
TP_GERM         11         1      0.2765      0.1492      3.4345  0.0638
TP_GERM         21         0      0.0000      0.0000           .       .
MOISIR          1          1      5.1608      0.7259     50.5401  0.0001
MOISIR          2          1      0.9318      0.8428      1.2224  0.2689
MOISIR          3          0      0.0000      0.0000           .       .
TP_CONS         21         1      6.9886      0.7549     85.7054  0.0001
TP_CONS         42         1      4.0045      0.7260     30.4223  0.0001
TP_CONS         62         0      0.0000      0.0000           .       .
MOISIR*TP_CONS  1   21     1     -4.3581      0.8495     26.3197  0.0001
MOISIR*TP_CONS  1   42     1     -1.7085      0.8031      4.5254  0.0334
MOISIR*TP_CONS  1   62     0      0.0000      0.0000           .       .
MOISIR*TP_CONS  2   21     1     -0.5561      0.9292      0.3581  0.5495
MOISIR*TP_CONS  2   42     1      0.7715      0.8713      0.7840  0.3759
MOISIR*TP_CONS  2   62     0      0.0000      0.0000           .       .
MOISIR*TP_CONS  3   21     0      0.0000      0.0000           .       .
MOISIR*TP_CONS  3   42     0      0.0000      0.0000           .       .
MOISIR*TP_CONS  3   62     0      0.0000      0.0000           .       .
```
OUTPUT

5.2.1 Plots of residuals

Bien que les tableaux soient utiles, il est plus facile de lire un graphique. On peut représenter sous différentes formes graphiques les informations contenues dans les tableaux.
On utilise l'exemple `lupin` (exemple (13), page 10). La représentation graphique est donnée par la Figure 5.1.

PGM
```
legend1 frame offset=(,-2) shape=symbol(0.01,3)
        position=(top center inside) label=('');
Proc Gplot data=Result;
 symbol1 v=circle c=black h=3;
 symbol2 v=dot c=black h=3;
 symbol3 v=square c=black h=3;
 axis1 label=(a=90 "Residu standardise de deviance") order=-2 to 2 by 1;
 axis2 label=("Temperature de conservation") order=20 to 70 by 10;
 plot stresdev*tp_cons=tp_germ /frame vref=0 vaxis=axis1 haxis=axis2
                                vm=0 hm=0 legend=legend1;
 plot stresdev*tp_cons=moisir /frame vref=0 vaxis=axis1 haxis=axis2
                                vm=0 hm=0 legend=legend1;
run; quit;
```
PGM

PGM
```
Proc tabulate data=Result noseps;
 class tp_germ moisir tp_cons;
 var stresdev;
```

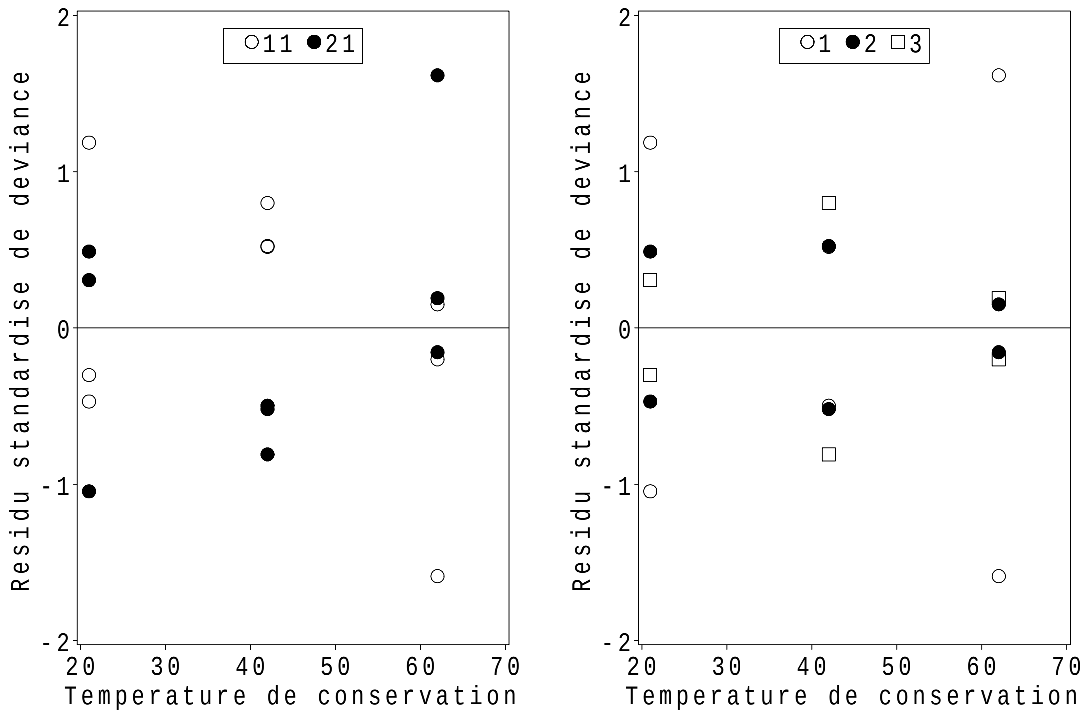


FIG. 5.1:

```
 table tp_germ*moisir,tp_cons*stresdev*sum*f=9.1;
 keylabel sum=' ';
run;
```
PGM

Les valeurs les moins bien ajustées sont celles correspondant à la température de conservation la plus elevée, 62 ° C et le taux d'humidité le plus faible. Les résidus correspondant à une température de conservation de 42 ° C sont tous positifs lorsque la température de germination est de 11 ° C.

OUTPUT

```
---------------------------------------------------
|                  |           TP_CONS            |
|                  |------------------------------|
|                  |   21    |   42    |   62    |
|                  |---------+---------+---------|
|                  |StResdev |StResdev |StResdev |
|------------------+---------+---------+---------|
|TP_GERM  MOISIR   |         |         |         |
|11       1        |      1.2|      0.5|     -1.6|
|         2        |     -0.5|      0.5|      0.2|
|         3        |     -0.3|      0.8|     -0.2|
|21       1        |     -1.0|     -0.5|      1.6|
|         2        |      0.5|     -0.5|     -0.2|
|         3        |      0.3|     -0.8|      0.2|
---------------------------------------------------
```
OUTPUT

Cela suggère une interaction entre la température de germination et celle de conservation.

PGM

```
proc Genmod data=Lupin;
```

```
 class tp_germ moisir tp_cons;
 model y/n=tp_germ moisir tp_cons tp_cons*moisir tp_germ*tp_cons
         / dist=bin type1;
run;
```
PGM

Mais le modèle obtenu n'est pas significativement meilleur.

OUTPUT

```
                    LR Statistics For Type 1 Analysis

             Source             Deviance    DF   ChiSquare  Pr>Chi

             INTERCEPT         1230.8604     0           .       .
             TP_GERM           1229.3937     1      1.4667  0.2259
             MOISIR             982.2939     2    247.0997  0.0001
             TP_CONS             63.1903     2    919.1036  0.0001
             MOISIR*TP_CONS       4.4298     4     58.7606  0.0001
             TP_GERM*TP_CONS      1.8924     2      2.5373  0.2812
```
OUTPUT

On représente les résidus en fonction des valeurs estimées (Figure 5.2).

5.2.2 Half-normal plots of residuals

On reprend l'exemple CS2 (exemple (12), page 10) et on représente les résidus de déviance standardisés en fonction du prédicteur linéaire (Figure 5.3).

PGM

```
Proc Gplot data=Result;
 symbol1 v=dot c=black;
 symbol2 v=circle c=black;
 axis1 label=(a=90 "Residu standardise de deviance") order=-2 to 2 by 1;
 axis2 label=("Predicteur lineaire") order=-3 to 5 by 1;
 axis3 label=(a=90 "Residu de deviance") order=-2 to 2 by 1;
 plot resdev*xbeta /frame haxis=axis2 vaxis=axis3 hm=0 vm=0;
 plot stresdev*xbeta /frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
run;
quit;
```
PGM

Cette représentation est très proche de celle des résidus de déviance non standardisés.
On peut aussi utiliser le graphique semi-normal des résidus (Figure 5.4) avec enveloppe simulée (half normal plot).

PGM

```
Data Result;
 merge CS2 Result;
 val=abs(stresdev);
run;
Proc sort data=Result; by val; run;

Data Result;
 set Result; ;
 i=_n_;
 phi=probit((i + 16 - 1/8)/(2 * 16 + 1/2));
run;
Proc Gplot data=Result;
```

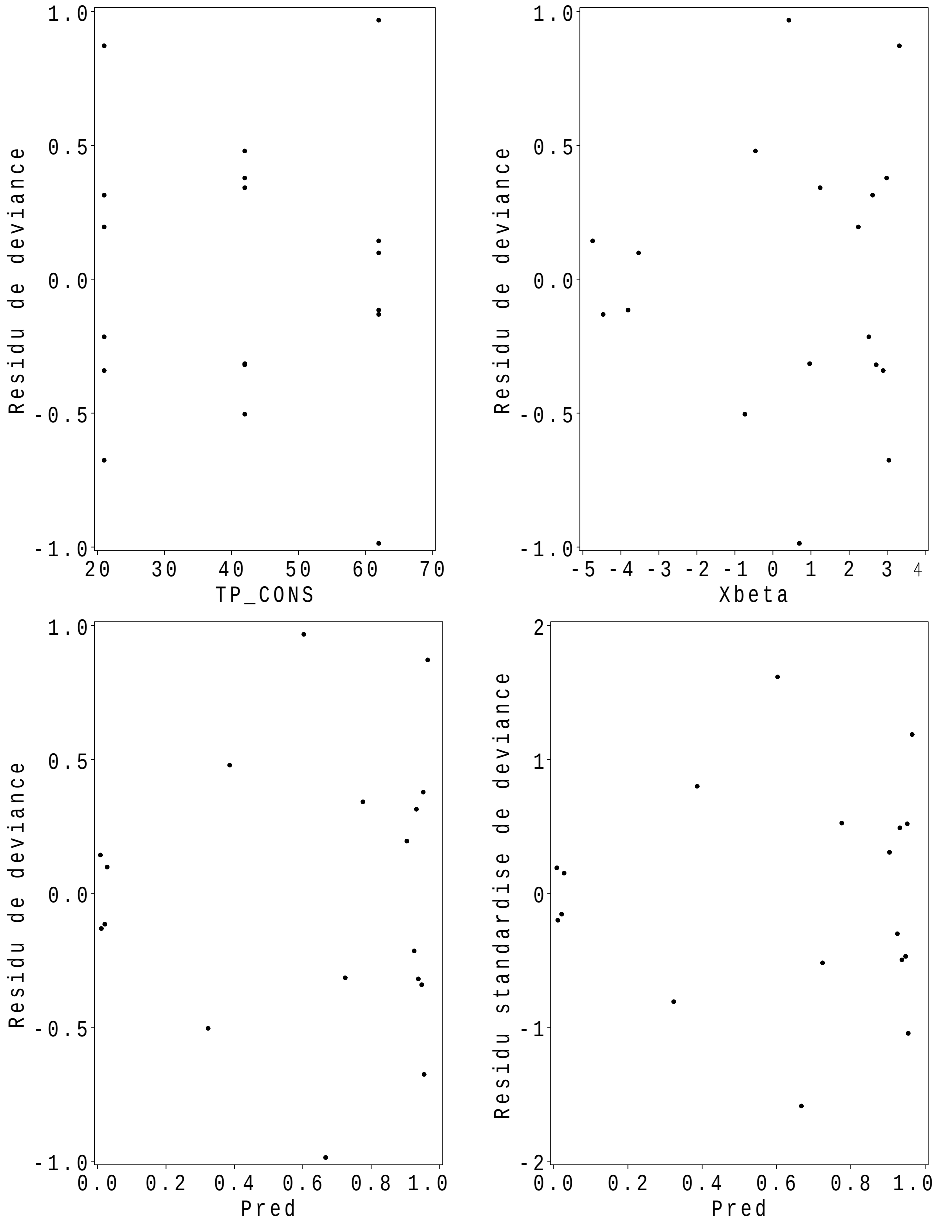

Residu de deviance
TP_CONS
Residu de deviance
Xbeta
Residu de deviance
Pred
Residu standardise de deviance
Pred

FIG. 5.2:

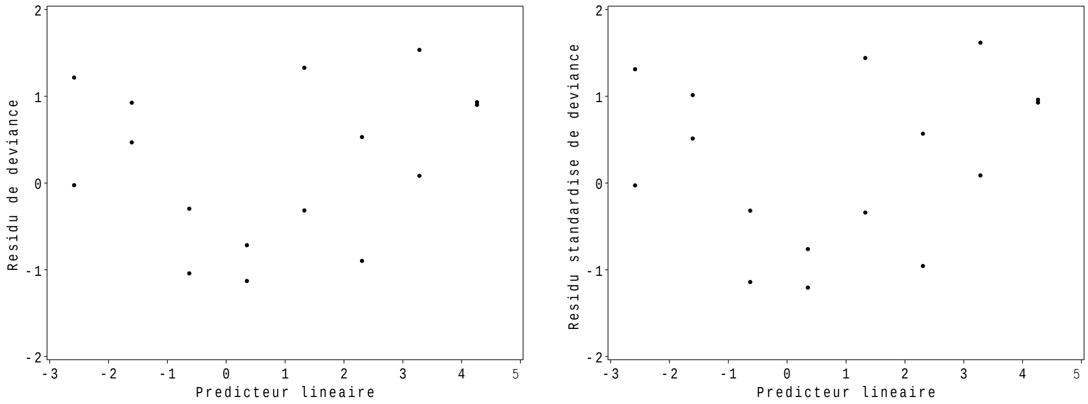


FIG. 5.3:

```
 symbol v=dot c=black;
 axis1 label=(a=90 h=2 "Val. abs. residus standardise de deviance") order=0 to 3 by .5;
 axis2 label=("Quantile de la loi normale") order=0 to 2.5 by .5;
 plot val*phi=1 /frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
run;
quit;
```
PGM

La représentation graphique des résidus en fonction du prédicteur linéaire suggère une structure quadratique des résidus (Figure 5.3) et donc l'ajout du régresseur conc^2.
On ajuste un modèle avec un effet quadratique. La structure des résidus est alors supprimée mais l'enveloppe simulée suggère un modèle surparamétré car tous les résidus, sauf un, sont inférieurs à la moyenne des résidus simulés (Figure 5.5)

5.2.3 The added variable plot

La question posée est *Faut-il ajouter une nouvelle variable explicative ?* L'exemple utilisé est l'exemple CS2 (exemple (12), page 10). La question de l'ajout d'un terme quadratique se pose. Le graphique de la variable ajoutée (ou "added variable plot") est un outil d'aide à la décision.
On essaie de voir si la conc^2 doit être introduite.

PGM

```
Proc genmod data=CS2;
 make 'obstats' out=Result noprint;
  model y/n=conc/dist=bin obstats residuals;
run;
Data _tmp_;
 merge CS2(keep=n conc) Result(keep=Reschi Hesswgt);
 ui=conc*conc;
run;
proc reg data=_tmp_;
 weight hesswgt;
 model ui= conc;
 output out=sortie r=ecart;
run;
Data _tmp_;
```

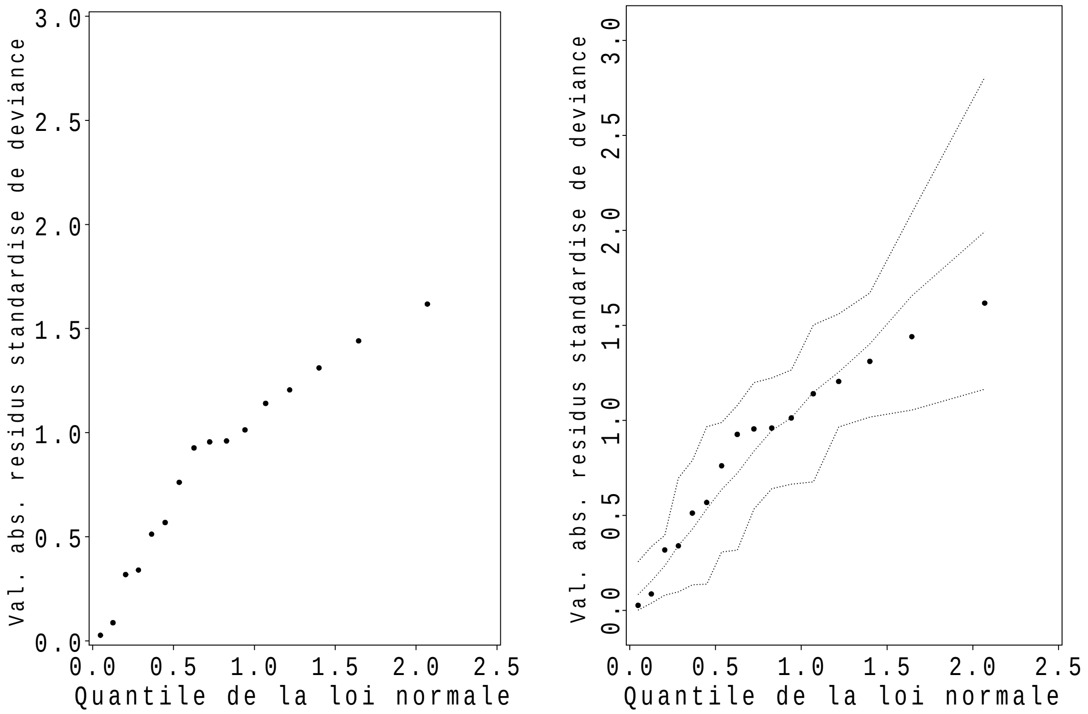

Val. abs. residus standardise de deviance
Quantile de la loi normale
0.0 0.5 1.0 1.5 2.0 2.5
0.0 0.5 1.0 1.5 2.0 2.5 3.0

FIG. 5.4:

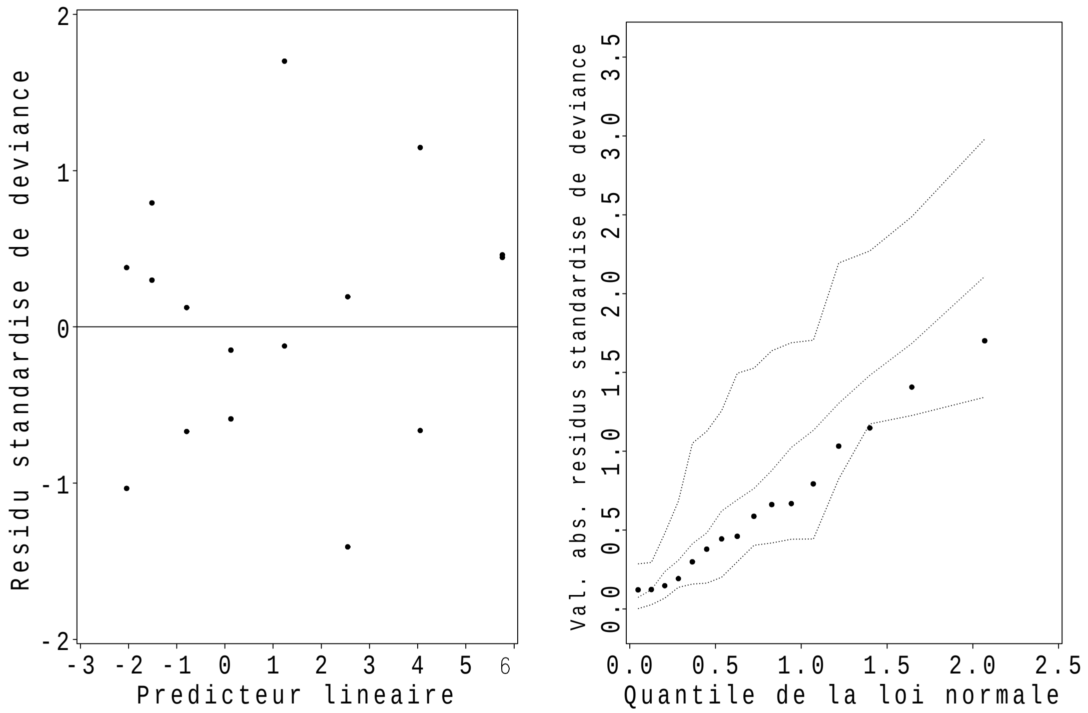

Residu standardise de deviance
Predicteur lineaire
-3 -2 -1 0 1 2 3 4 5 6
-2 -1 0 1 2
Val. abs. residus standardise de deviance
Quantile de la loi normale
0.0 0.5 1.0 1.5 2.0 2.5
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5

FIG. 5.5:

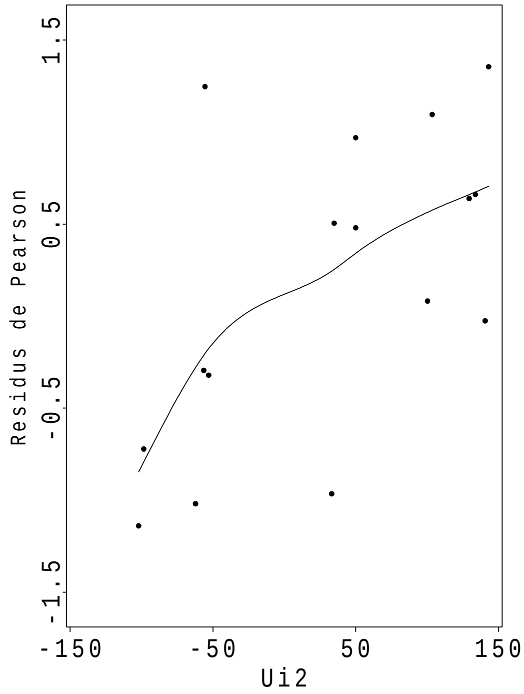


FIG. 5.6:

```
 merge _tmp_ sortie;
 ui2=ecart*sqrt(hesswgt);
run;
```
PGM

On obtient la Figure 5.6. Ce graphique suggère l'existence d'une structure des résidus de Pearson, explicable par la variable ajoutée conc^2.
On ajuste le modèle.

PGM
```
Proc genmod data=CS2;
  model y/n=conc/dist=bin type1;
run;
```
PGM

OUTPUT
```
                     Criteria For Assessing Goodness Of Fit

              Criterion             DF          Value       Value/DF

              Deviance              13         7.9273         0.6098
              Scaled Deviance       13         7.9273         0.6098
              Pearson Chi-Square    13         7.3590         0.5661
              Scaled Pearson X2     13         7.3590         0.5661
              Log Likelihood         .      -182.1135              .

                       LR Statistics For Type 1 Analysis

                Source       Deviance    DF   ChiSquare  Pr>Chi

                INTERCEPT    289.1413     0           .       .
                CONC          12.5053     1    276.6360  0.0001
                CONC*CONC      7.9273     1      4.5779  0.0324
```
OUTPUT

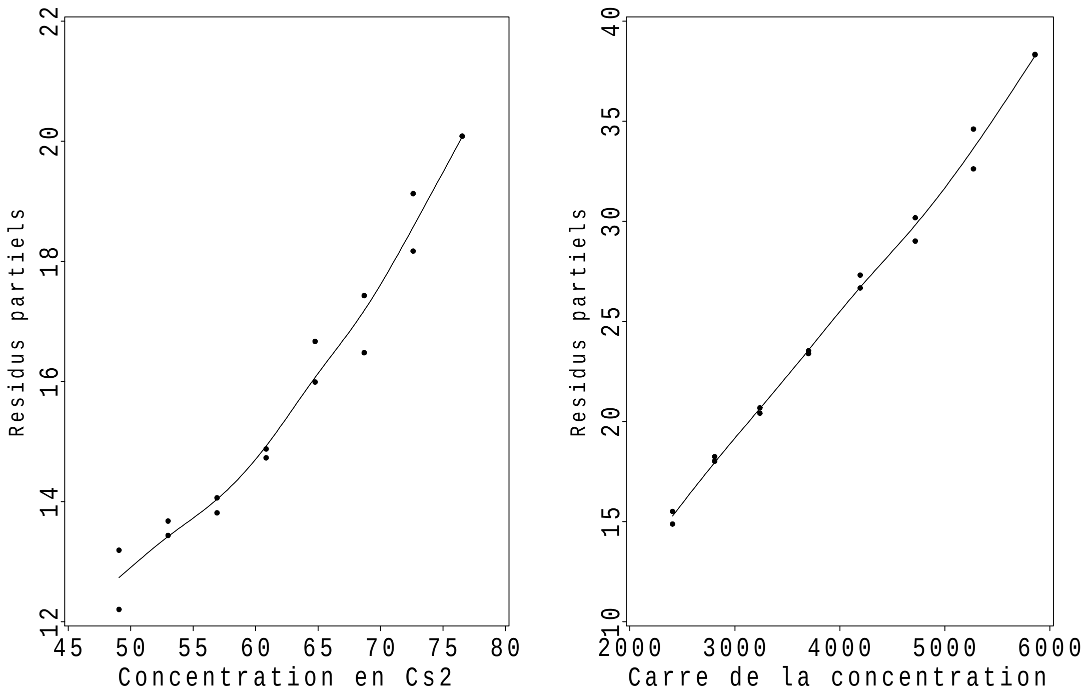


FIG. 5.7:

5.2.4 The partial residual plot

Cette fois la question que l'on se pose est *Faut-il transformer la variable explicative ?* Pour cela on utilise le graphique des résidus partiels. On reprend l'exemple CS2 (exemple (12), page 10) et on cherche à savoir si on doit modifier la variable conc.

PGM

```
Proc genmod data=CS2;
 make 'Parmest' out=Param(keep=parm estimate);
 make 'Obstats' out=Result noprint;
 model y/n=conc/dist=bin obstats residuals;
run;
Data _tmp_;
 merge CS2(keep=CONC) Result(keep=Resraw Hesswgt);
 if _n_=1 then merge param(where=(parm='CONC'));
 retain estimate 0;
 rp=(resraw/hesswgt) + estimate*conc;
run;
Proc Gplot data=_tmp_;
 symbol1 v=dot i=sm60s;
 axis1 label=(a=90 h=2 "Residus partiels")
       value=(a=90 h=3pct) order=12 to 22 by 2;
 axis2 label=("Concentration en Cs2") order=45 to 80 by 5;
 plot rp*conc=1 / frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
run;
quit;
```

PGM

On obtient la Figure 5.7a. Si on fait la même opération avec la conc^2, on obtient la Figure 5.7b. L'absence de courbure dans le graphique obtenu avec le modèle quadratique suggère une prise en compte correcte de l'effet concentration.

5.2.5 The constructed variable plot

La question de la nécessité de transformer la variable explicative et du choix de la transformation peut se poser. La méthode graphique des "constructed variable plot" peut permettre de répondre à la question. On reprend l'exemple `CS2` (exemple (12), page 10) et on cherche la meilleure transformation de la variable conc.

PGM
```
Proc genmod data=CS2;
 make 'Parmest' out=Param(keep=parm estimate);
 make 'Obstats' out=Result noprint;
 model y/n=conc/dist=bin obstats residuals;
run;
Data _tmp_;
 merge CS2(keep=n Conc) Result(keep=Reschi Hesswgt);
 if _n_=1 then merge param(where=(parm='CONC'));
 retain estimate 0;
 zi=estimate*conc*log(conc);
run;
Proc reg data=_tmp_;
 weight hesswgt;
 model zi= conc;
 output out=_out_ r=ecart;
run;
Data _tmp_;
 merge _tmp_ _out_;
 ui2=ecart*sqrt(hesswgt);
run;
```
PGM

Le graphique est obtenu avec les instructions suivantes. On obtient la Figure 5.8. Le graphe suggère l'introduction d'un terme quadratique de la concentration.

PGM
```
Proc Gplot data=_tmp_;
 symbol1 v=dot i=sm60s;
 axis1 label=(a=90 h=2pct "Residus de Pearson")
       value=(a=90 h=2pct) order=-1.5 to 1.5 by .5;
 axis2 label=(h=2pct "Residus de la variable construite")
       value=(h=2pct) order=-0.3 to 0.4 by 0.1;
 plot reschi*ui2=1 / frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
run;
quit;
```
PGM

5.3 Checking the adequacy of the link function

5.4 Identification of outlying observations

On travaille avec l'exemple `Beetle` (exemple (11), page 10). On ajuste un modèle

PGM
```
Proc Genmod data=beetles;
 make 'obstats' out=Result noprint;
 class insect;
 model y/n=insect ldose/noint dist=bin obstats residuals;
```

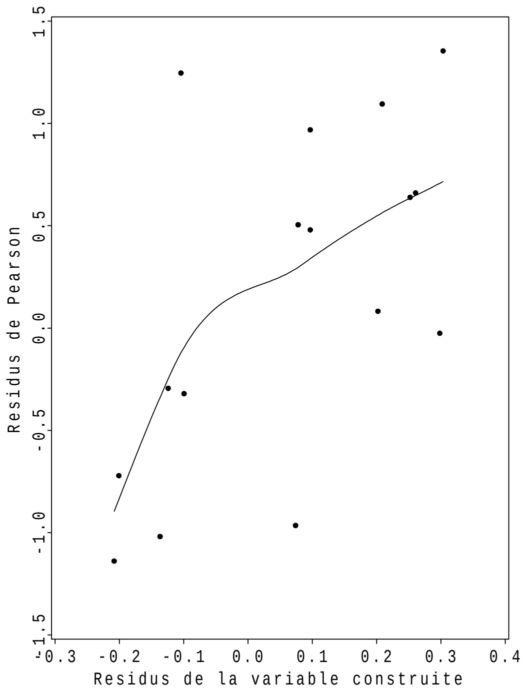


FIG. 5.8:

```
run;
Proc Print data=Result noobs;
 format Reschi--Reslik 6.3;
 var Reschi Resdev streschi stresdev Reslik;
run;
```
PGM

... et on regarde les résidus.

OUTPUT

RESCHI	RESDEV	STRESCHI	STRESDEV	RESLIK
-0.109	-0.110	-0.119	-0.120	-0.120
-0.502	-0.517	-0.558	-0.574	-0.571
2.785	2.612	3.115	2.922	2.962
-0.145	-0.145	-0.166	-0.167	-0.166
-1.205	-1.197	-1.416	-1.407	-1.410
-0.659	-0.649	-0.800	-0.787	-0.791
-1.771	-2.027	-2.013	-2.304	-2.242
0.982	0.955	1.123	1.092	1.099
0.261	0.260	0.298	0.297	0.297
-0.371	-0.370	-0.424	-0.423	-0.423
1.378	1.435	1.592	1.659	1.642
-1.072	-1.026	-1.236	-1.182	-1.195
-0.713	-0.708	-0.953	-0.946	-0.949
-0.454	-0.448	-0.537	-0.530	-0.532
0.980	1.041	1.086	1.154	1.142
0.693	0.741	0.742	0.794	0.787
-0.353	-0.339	-0.370	-0.355	-0.357
0.874	1.232	0.902	1.271	1.252

OUTPUT

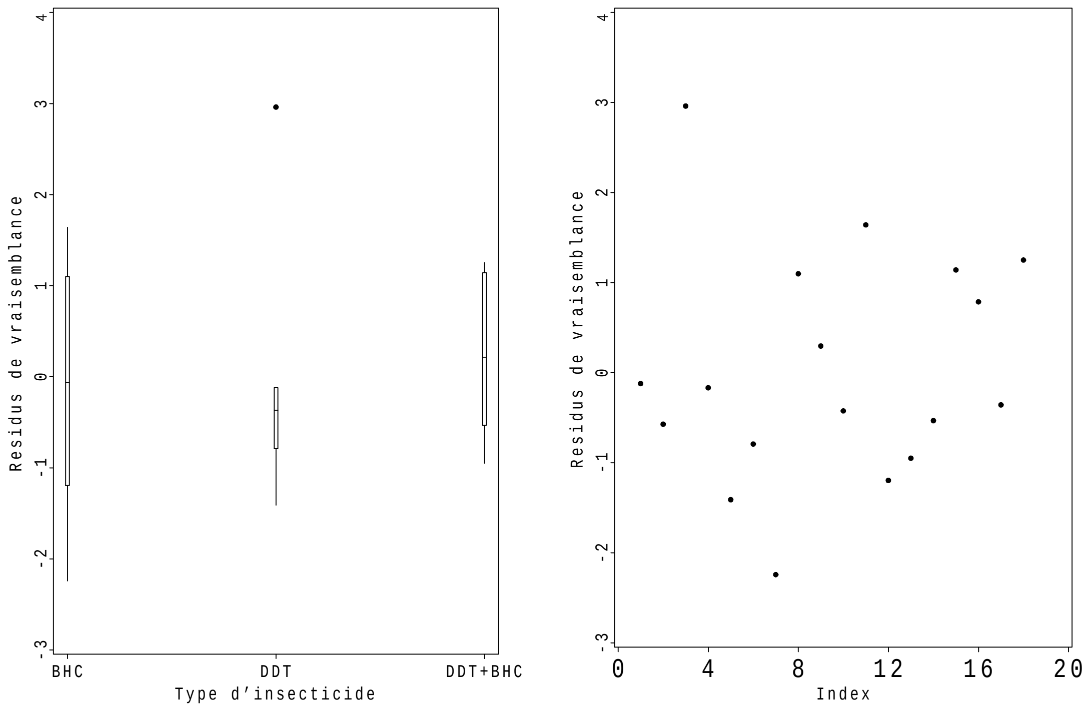


FIG. 5.9:

et les représentations graphiques (Figure 5.9).

PGM

```
Data _tmp_;
 merge Beetles Result(keep=reslik);
 index=_n_;
run;
Proc Gplot data=_tmp_ ;
 symbol1 v=dot i=box w=100 c=black;
 symbol2 v=dot i=none c=black;
 axis1 label=(a=90 h=2pct "Residus de vraisemblance")
       value=(a=90 h=2pct) order=-3 to 4 by 1;
 axis2 label=(h=2pct "Type d'insecticide") value=(h=2pct) offset=(2,2);
 axis3 label=(h=2pct "Index") order=0 to 20 by 4;
 plot reslik*insect=1 / frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
 plot reslik*index=2 / frame haxis=axis3 vaxis=axis1 hm=0 vm=0;
run;
quit;
```

PGM

Pour l'insecticide DDT, tous les résidus de vraisemblance sont négatifs sauf 1 (Figure 5.9a). Ce résidu correspond à la troisième observation (Figure 5.9b).
Les enveloppes confirment que ce residu est particulierement élevé (Figure 5.10). On peut faire un test : le résidu le plus grand (3e observation) correspond-il à une observation discordante ? test du rapport de vraisemblance.
Il faut ajuster le même modèle, en supprimant la 3ieme observation et comparer la différence des déviances à un chi2 à 1 degré de liberté.
Une observation est discordante. Que faire ? Soit le modèle est inadéquat, soit l'observation a une particularité.
Si c'est le modèle que l'on remet en cause, on peut l'améliorer en ajoutant d'autres variables explicatives. On peut aussi modifier certains régresseurs (dose ou log(dose) ...), autoriser des non-linéarités dans la liaison entre le logit et le régresseur (en remplaçant des effets linéaires par des

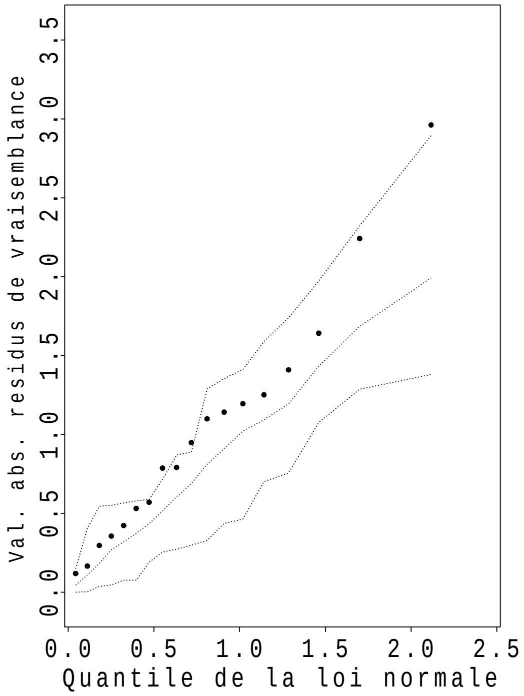


FIG. 5.10:

effets quadratiques ..., remplacer un effet linéaire par un effet différent pour chaque niveau du facteur). On peut également remettre en cause le lien choisi.
Si la validité du modèle n'est pas remise en cause, s'il est considéré comme adéquat, c'est l'observation qui est suspecte. Dans cet exemple, l'outlier détecté reflète le fait que le lot de 47 souris exposées à 3.48 mg/10cm^2 de DDT est plus sensible au poison que les autres lots.
Bien sur, on peut rejeter l'outlier (la valeur extérieure) et réajuster le modèle sur le nouveau jeu de données. Bien que cette démarche puisse être appropriée quand le modèle est considéré comme inviolable, cette procédure ne doit pas être adoptée comme une procédure de routine.
Si un ou des outliers sont identifiés, l'analyse peut-être refaite en excluant les outliers. Si les conclusions sont identiques, pas de problème. Si les conclusions sont différentes, la façon de considérer ces outliers est cruciale.
Ce ne sont pas des considerations statistiques qui détermineront l'attitude à adopter en présence d'outliers, ce sont des discussions avec les personnes qui ont recueilli les données qui permettront de décider du statut de ces données.

5.5 Identification of influential observations

Une observation influente n'est pas nécessairement détectée par l'examen des résidus. D'autres méthodes de détection doivent donc être mises en oeuvre. Lorsqu'une observation est distante des autres en terme de valeurs prises par les prédicteurs, elle peut être influente.
Cette distance peut être mesurée par le `leverage : h`. h dépend essentiellement des valeurs prises par les variables explicatives. Lorsque h est élevé, l'observation peut être influente, mais ne le sera pas forcément.
On crée un jeu de données artificielles.

PGM

```
Data artifice;
 input x y @@;
 n=10;
datalines;
```

```
.1  2   .2  2   .3  4   .4  7   .5  5   .6  6   .7  5   1.5  5
;
run;
```
PGM

sur lequel on ajuste le modèle $\text{logit}(p) = x$,

PGM
```
Proc genmod data=artifice;
 make 'obstats' out=stat noprint;
 model y/n=x /dist=bin obstats residuals;
run;
```
PGM

et on regarde les résidus et le leverage.

PGM
```
Data Stat;
 merge Artifice Stat;
 h=1 - ((ResChi**2)/(StResChi**2)) ;
 ychap=10*pred;
 logit=log( (y/n) / (1- (y/n)) );
 areslik=abs(reslik);
run;
proc print data=stat;
 var ychap h StResChi StResDev;
run;
```
PGM

Sur ce tableau on voit que l'observation 8 a un fort leverage.

OUTPUT
```
OBS      YCHAP         H        STRESCHI    STRESDEV

 1      3.74016    0.26235    -1.32415    -1.38778
 2      3.90908    0.20850    -1.39065    -1.45761
 3      4.08066    0.16764    -0.05688    -0.05694
 4      4.25450    0.14068     1.89433     1.89082
 5      4.43021    0.12837     0.38852     0.38725
 6      4.60736    0.13122     0.94789     0.94712
 7      4.78550    0.14953     0.14724     0.14717
 8      6.19253    0.81172    -1.78984    -1.76370
```
OUTPUT

de même que sur le graphique (Figure 5.11a).
On peut aussi faire un "index-plot" des valeurs de h. Les valeurs de h supérieures à $2p/n$ (avec p/n la moyenne des éléments diagonaux de la matrice H) peuvent être considérées comme particulièrement grandes.
Une représentation plus intéressante consiste à faire le graphe des résidus (de vraisemblance par exemple) en fonction de h.
Dans l'exemple artificiel, cette représentation montre qu'une observation a une valeur de h particulèrement élevée (Figure 5.11b) et que le résidu correspondant est aussi assez grand.
Si on refait tourner le modèle en éliminant l'observation 8,

PGM
```
Proc genmod data=artifice(where=(x ne 1.5));
 make 'obstats' out=stat noprint;
 model y/n=x /dist=bin obstats residuals;
run;
```

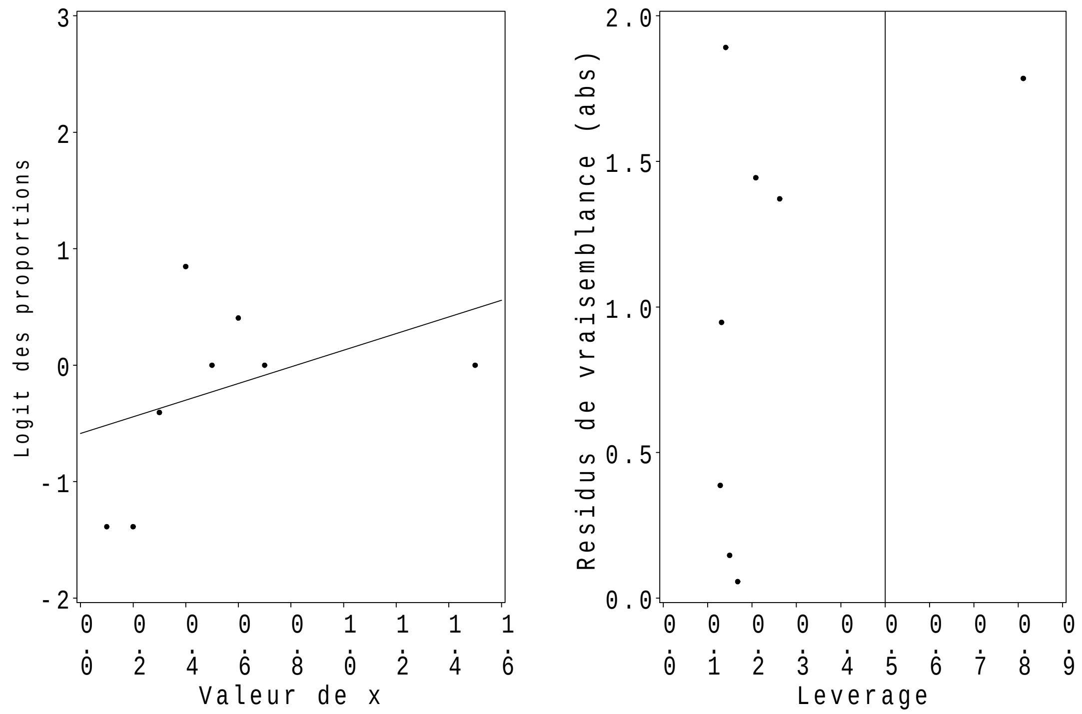


FIG. 5.11:

PGM

OUTPUT

OBS	YCHAP	H	STRESCHI	STRESDEV
1	2.57091	0.42384	-0.54423	-0.55918
2	3.12275	0.29183	-0.91041	-0.94864
3	3.73351	0.19345	0.19400	0.19312
4	4.38752	0.15342	1.80939	1.81270
5	5.06352	0.18681	-0.04455	-0.04455
6	5.73720	0.29346	0.19992	0.20050
7	6.38460	0.45719	-1.23696	-1.21237
8	.	.	.	.

OUTPUT

on obtient la Figure 5.12b.

5.5.1 Influence of the ith observation on the overall fit of the model

L'influence de la i^e observation sur l'ajustement global peut être mesurée par la réduction de déviance obtenue lorsque l'on omet cette observation. Cette différence de déviance peut être approchée par le carré du résidu de vraisemblance. Les résidus de vraisemblance sont plus faciles à calculer que les différences de déviances qui nécessitent l'estimation d'autant de modèles qu'il y a d'observations.
Un index plot des carrés des résidus de vraisemblance est un bon outil pour étudier la contribution de chaque observation à l'ajustement global.
La PROC LOGISTIC possède une option qui permet de sortir tout un tas de diagnostiques.

PGM

```
Proc logistic data=artifice;
 model y/n=x /link=logit influence scale=none aggregate;
run;
```

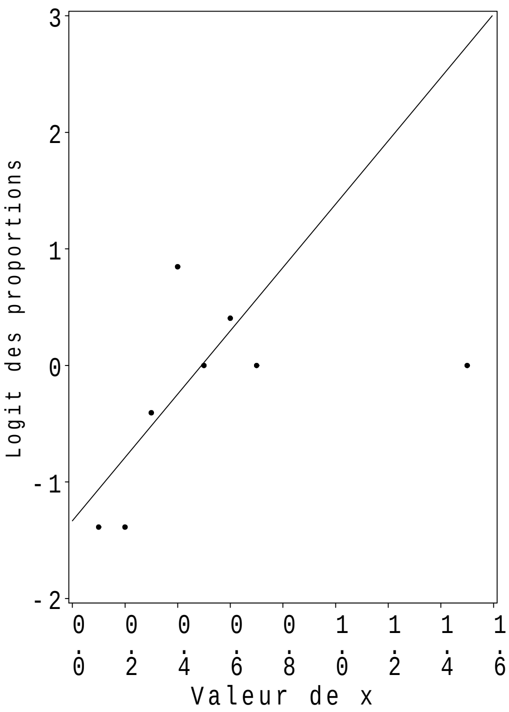


FIG. 5.12:

____ *PGM*

et en particulier `DIFDEV` et `DIFCHISQ`.

OUTPUT ____

```
                         DIFDEV                           DIFCHISQ

 Case                 (1 unit = 0.22)                   (1 unit = 0.22)
Number      Value     0 2 4 6 8  12  16      Value     0 2 4 6 8  12  16

     1     1.8807    |        *        |    1.7534    |        *        |
     2     2.0849    |         *       |    1.9339    |         *       |
     3    0.00324    |*                |   0.00324    |*                |
     4     3.5771    |                *|    3.5885    |                *|
     5     0.1501    | *               |    0.1510    | *               |
     6     0.8972    |    *            |    0.8985    |    *            |
     7     0.0217    |*                |    0.0217    |*                |
     8     3.1860    |              * |    3.2035    |              * |
```

____ *OUTPUT*

5.5.2 Influence of the ith observation on the set of parameter estimates

L'influence de la i^{e} observation sur l'ensemble des paramètres peut être évaluée avec la statistique `C` de SAS. Cette statistique est équivalente à la distance de Cook utilisée dans le cadre du modèle linéaire.

OUTPUT ____

```
                         Regression Diagnostics


                          C                                CBAR
```

```
 Case                   (1 unit = 0.86)                (1 unit = 0.16)
Number       Value    0 2 4 6 8  12  16        Value    0 2 4 6 8  12  16

     1      0.6236   | *               |      0.4600   |   *              |
     2      0.5094   | *               |      0.4032   |  *               |
     3    0.000652   |*                |    0.000542   |*                 |
     4      0.5875   | *               |      0.5048   |   *              |
     5      0.0222   |*                |      0.0194   |*                 |
     6      0.1357   |*                |      0.1179   | *                |
     7     0.00381   |*                |     0.00324   |*                 |
     8     13.8109   |               *|      2.6004   |                 *|
```
OUTPUT

5.5.3 Influence of the ith observation on the jth parameter estimate

L'influence de la i^e observation sur l'un des paramètres du modèle peut être évaluée avec la statistique `Dfbeta` de SAS.

OUTPUT

```
                                Regression Diagnostics

                    INTERCPT Dfbeta                    X        Dfbeta

 Case                   (1 unit = 0.23)                (1 unit = 0.43)
Number       Value    -8  -4  0 2 4 6 8        Value    -8  -4  0 2 4 6 8

     1     -0.7856   |     *  |         |      0.5801   |        |*        |
     2     -0.6949   |     *  |         |      0.4576   |        |*        |
     3     -0.0236   |        *         |      0.0129   |        *         |
     4      0.6334   |        |  *      |     -0.2469   |       *|         |
     5      0.1002   |        *         |     -0.0138   |        *         |
     6      0.1737   |        |*        |      0.0563   |        *         |
     7      0.0160   |        *         |      0.0230   |        *         |
     8      1.8747   |        |       *|     -3.4266   |*       |         |
```
OUTPUT

5.6 Checking the assumption of a binomial distribution

En cas de surdispersion, la déviance obtenue après ajustement d'un modèle adéquat est plus grande que celle espérée sous l'hypothèse d'une distribution binomiale.
On regarde ce problème avec l'exemple `rats` (exemple (4), page 6). On observe la proportion de rats survivants dans des portées, les rattes ayant été affectées à deux groupes dont les régimes alimentaires diffèrent.
On ajuste un modèle ne comprenant qu'un effet (`treat`).

PGM

```
Proc genmod data=rats;
 make 'obstats' out=stat(keep=pred resdev stresdev reschi) noprint;
 class treat;
 model y/n=treat /dist=bin obstats residuals;
run;
```
PGM

OUTPUT

```
                    Criteria For Assessing Goodness Of Fit

             Criterion              DF          Value       Value/DF
```

```
Deviance              30       86.1871        2.8729
Scaled Deviance       30       86.1871        2.8729
Pearson Chi-Square    30       80.6353        2.6878
Scaled Pearson X2     30       80.6353        2.6878
Log Likelihood         .     -129.5709             .
```

OUTPUT

La déviance est très élevée par rapport au nombre de degrés de liberté. Si on suppose que le modèle est adéquat, la déviance est significativement trop grande.
On ne dispose d'aucune autre variable explicative supplémentaire qui permettrait d'améliorer le modèle. On pourrait estimer une proportion pour chacune des portées mais le modèle serait alors saturé (le nombre d'observations étant égal au nombre de paramètres).
On peut aussi se demander si cette déviance élevée peut être attribuée à quelques observations particulières.

PGM

```
Data Tmp;
 merge rats stat;
 h=1 - ((Resdev**2)/(StResdev**2)) ;
 ychap=pred*n;
 d=(h*reschi**2)/( 2* (1-h)**2);
 index=_n_;
run;
Proc print data=tmp;
 var treat y n ychap h stresdev d;
run;
```

PGM

La consultation du tableau des résidus ne confirme pas cette hypothèse.

OUTPUT

OBS	TREAT	Y	N	YCHAP	H	STRESDEV	D
1	traite	13	13	11.6835	0.082278	1.73921	0.07155
2	traite	12	12	10.7848	0.075949	1.66525	0.06013
3	traite	9	9	8.0886	0.056962	1.42756	0.03248
4	traite	9	9	8.0886	0.056962	1.42756	0.03248
5	traite	8	8	7.1899	0.050633	1.34142	0.02532
6	traite	8	8	7.1899	0.050633	1.34142	0.02532
7	traite	12	13	11.6835	0.082278	0.31580	0.00413
8	traite	11	12	10.7848	0.075949	0.22030	0.00189
9	traite	9	10	8.9873	0.063291	0.01374	0.00001
10	traite	9	10	8.9873	0.063291	0.01374	0.00001
11	traite	8	9	8.0886	0.056962	-0.09942	0.00031
12	traite	11	13	11.6835	0.082278	-0.61448	0.01929
13	traite	4	5	4.4937	0.031646	-0.66647	0.00904
14	traite	5	7	6.2911	0.044304	-1.39204	0.06346
15	traite	7	10	8.9873	0.063291	-1.79482	0.15651
16	traite	7	10	8.9873	0.063291	-1.79482	0.15651
17	contro	12	12	9.2690	0.082759	2.59939	0.17390
18	contro	11	11	8.4966	0.075862	2.47942	0.14395
19	contro	10	10	7.7241	0.068966	2.35526	0.11721
20	contro	9	9	6.9517	0.062069	2.22617	0.09355
21	contro	10	11	8.4966	0.075862	1.24096	0.05192
22	contro	9	10	7.7241	0.068966	1.09042	0.03684
23	contro	9	10	7.7241	0.068966	1.09042	0.03684

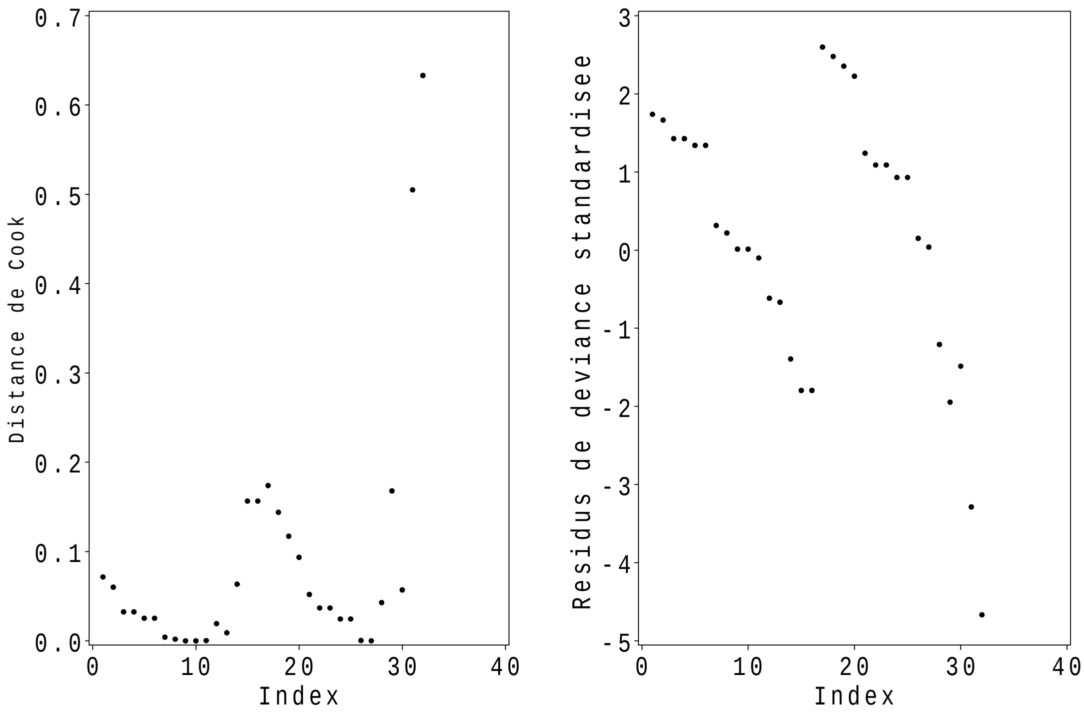


FIG. 5.13:

24	contro	8	9	6.9517	0.062069	0.93115	0.02450
25	contro	8	9	6.9517	0.062069	0.93115	0.02450
26	contro	4	5	3.8621	0.034483	0.15200	0.00040
27	contro	7	9	6.9517	0.062069	0.03974	0.00005
28	contro	4	7	5.4069	0.048276	-1.20700	0.04287
29	contro	5	10	7.7241	0.068966	-1.94487	0.16793
30	contro	3	6	4.6345	0.041379	-1.48466	0.05703
31	contro	3	10	7.7241	0.068966	-3.28635	0.50503
32	contro	0	7	5.4069	0.048276	-4.66630	0.63311

OUTPUT

Les observations 31 et 32 ont une grande valeur de la distance de Cook (D) mais le fait de les écarter ne diminue pas suffisamment la déviance.
Les représentations graphiques indiquent que les observations 15,16,28,29 et 30 ont des résidus supérieurs à 2 en valeur absolue (Figure 5.13).

PGM

```
Proc gplot data=tmp;
 symbol1 v=dot i=none c=black;
 axis1 label=(a=90 h=2 "Distance de Cook") order=0 to .7 by .1;
 axis2 label=("Index") order=0 to 40 by 10;
 axis3 label=(a=90 "Residus de deviance standardisee") order=-5 to 3 by 1;
 plot d*index=1/ frame haxis=axis2 vaxis=axis1 hm=0 vm=0;
 plot stresdev*index=1/ frame haxis=axis2 vaxis=axis3 hm=0 vm=0;
run;
quit;
```

PGM

Cependant on ne va pas continuer à exclure des observations jusqu'à obtenir satisfaction. Il faut conclure à l'existence d'une surdispersion.

5.7 Model checking for binary data

5.8 Summary and recommendations

5.9 A further example on the use of diagnostics

Deuxième partie

MISE EN ŒUVRE AVEC SPLUS

Chapitre 1

Introduction

1.1 Some examples

(1) *Exemple 1.1, p.2 : Smoked mice*

Des souris sont placées dans des cages dans lesquelles on injecte ou non de la fumée de cigarette. Après un certain temps, on sacrifie les animaux et on observe ou non le développement d'une tumeur.

```
> souris_c(21,2,19,13)
> souris_matrix(souris,ncol=2)
> dimnames(souris)_list(c("tumeur","pas tumeur"),c("cigarette","controle"))

> souris
           cigarette controle
    tumeur        21       19
pas tumeur         2       13
```

(2) *Example 1.2, p.2 : Plum root-stocks*

On cherche à comparer la proportion de plants qui "prennent" ou pas selon qu'ils sont plantés immédiatement ou au printemps et selon la taille des racines (courtes ou longues).

```
> prune_matrix(c(107,133,156,84,31,209,84,156),nrow=2)
> dimnames(prune)_list(c("vivant","mort"),
+         c("precode.court","precoce.long","tardif.court","tardif.long"))

> prune
       precode.court precoce.long tardif.court tardif.long
vivant           107          156           31          84
  mort           133           84          209         156
```

(3) *Example 1.3, p.3 : Germination of Orobanche*

```
y_c(10,23,23,26,17,5,53,55,32,46,10,8,10,8,23,0,3,22,15,32,3)
n_c(39,62,81,51,39,6,74,72,51,79,13,16,30,28,45,4,12,41,30,51,7)
orobanche_data.frame(y=y,n=n,variete=as.factor(c(rep(1:2,c(11,10)))),
                     extrait=as.factor(rep(c(1,2,1,2),c(5,6,5,5))))
levels(orobanche$variete)_c("O. aegyptiaca 75","O. aegyptiaca 75 73")
levels(orobanche$extrait)_c("haricot","concombre")
> orobanche
    y  n             variete   extrait
```

```
 1 10 39    O. aegyptiaca 75   haricot
 2 23 62    O. aegyptiaca 75   haricot
 3 23 81    O. aegyptiaca 75   haricot
 4 26 51    O. aegyptiaca 75   haricot
 5 17 39    O. aegyptiaca 75   haricot
 6  5  6    O. aegyptiaca 75 concombre
 7 53 74    O. aegyptiaca 75 concombre
 8 55 72    O. aegyptiaca 75 concombre
 9 32 51    O. aegyptiaca 75 concombre
10 46 79    O. aegyptiaca 75 concombre
11 10 13    O. aegyptiaca 75 concombre
12  8 16 O. aegyptiaca 75 73   haricot
13 10 30 O. aegyptiaca 75 73   haricot
14  8 28 O. aegyptiaca 75 73   haricot
15 23 45 O. aegyptiaca 75 73   haricot
16  0  4 O. aegyptiaca 75 73   haricot
17  3 12 O. aegyptiaca 75 73 concombre
18 22 41 O. aegyptiaca 75 73 concombre
19 15 30 O. aegyptiaca 75 73 concombre
20 32 51 O. aegyptiaca 75 73 concombre
21  3  7 O. aegyptiaca 75 73 concombre
```

(4) *Example 1.4, p.4 : A toxicological study*

On observe la proportion de rats survivants dans des portées, les rattes ayant été affectées à deux groupes dont les régimes alimentaires diffèrent.

```
> rats_data.frame(
+   y=c(13,12,9,9,8,8,12,11,9,9,8,11,4,5,7,7,12,11,
+                   10,9,10,9,9,8,8,4,7,4,5,3,3,0),
+   n=c(13,12,9,9,8,8,13,12,10,10,9,13,5,7,10,10,12,
+             11,10,9,11,10,10,9,9,5,9,7,10,6,10,7),
+   trait=factor(rep(1:2,c(16,16)),labels=c("traite","control"))
+   )
```

```
> rats
    y  n   trait
 1 13 13  traite
 2 12 12  traite
 3  9  9  traite
 4  9  9  traite
 5  8  8  traite
 6  8  8  traite
 7 12 13  traite
 8 11 12  traite
 9  9 10  traite
10  9 10  traite
11  8  9  traite
12 11 13  traite
13  4  5  traite
14  5  7  traite
15  7 10  traite
16  7 10  traite
17 12 12 control
18 11 11 control
19 10 10 control
20  9  9 control
21 10 11 control
22  9 10 control
23  9 10 control
24  8  9 control
25  8  9 control
26  4  5 control
27  7  9 control
28  4  7 control
29  5 10 control
30  3  6 control
31  3 10 control
32  0  7 control
```

(5) *Example 1.5, p.5 : Aircraft fasteners*

Dans cet exemple, on teste la charge maximale que peuvent supporter des rivets en alliage utilisés dans l'aviation. On teste des lots de rivets avec des charges croissantes.

```
> aircraft_data.frame(
+   load=c(2500,2700,2900,3100,3300,3500,3700,3900,4100,4300),
+   n=c(50,70,100,60,40,85,90,50,80,65),
+   k=c(10,17,30,21,18,43,54,33,60,51))

> aircraft$p_aircraft$k/aircraft$n

> aircraft
   load   n  k         p
 1 2500  50 10 0.2000000
 2 2700  70 17 0.2428571
 3 2900 100 30 0.3000000
 4 3100  60 21 0.3500000
 5 3300  40 18 0.4500000
 6 3500  85 43 0.5058824
 7 3700  90 54 0.6000000
 8 3900  50 33 0.6600000
 9 4100  80 60 0.7500000
10 4300  65 51 0.7846154
```

(6) *Example 1.6, p.6 : Antipneumococcus serum*

Cinq doses d'un serum anti-pneumocoque sont expérimentées. Pour chaque dose, on utilise un lot de 40 souris. Les souris succombent ou pas.

```
> serum_data.frame(dose=c(0.0028,0.0056,0.0112,0.0225,0.0450),
+                  y=c(35,21,9,6,1),
+                  n=rep(40,5))

> serum
    dose  y  n
1 0.0028 35 40
2 0.0056 21 40
3 0.0112  9 40
4 0.0225  6 40
5 0.0450  1 40
```

(7) *Example 1.7, p.8 : Determination of ESR*

Cet exemple concerne des données relatives au taux de sédimentation de globules rouges. Un taux de sédimentation inférieur à un certain seuil est un signe de bonne santé (Y=1), alors qu'un taux supérieur au même seuil est un indicateur d'un problème (Y=0).

```
> ESR_data.frame(fibrinogene=c(
+   2.52,2.56,2.19,2.18,3.41,2.46,3.22,2.21,3.15,2.60,2.29,2.35,5.06,3.34,2.38,
+   3.15,3.53,2.68,2.6,2.23,2.88,2.65,2.09,2.28,2.67,2.29,2.15,2.54,3.93,3.34,
+   2.99,3.32),
+   globuline=c(38,31,33,31,37,36,38,37,39,41,36,29,37,32,37,36,46,34,38,37,
+   30,46,44,36,39,31,31,28,32,30,36,35),
+   reponse=c(0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0))
```

```
> ESR                                           16      3.15      36      0
   fibrinogene globuline reponse                17      3.53      46      1
 1        2.52        38       0                18      2.68      34      0
 2        2.56        31       0                19      2.60      38      0
 3        2.19        33       0                20      2.23      37      0
 4        2.18        31       0                21      2.88      30      0
 5        3.41        37       0                22      2.65      46      0
 6        2.46        36       0                23      2.09      44      1
 7        3.22        38       0                24      2.28      36      0
 8        2.21        37       0                25      2.67      39      0
 9        3.15        39       0                26      2.29      31      0
10        2.60        41       0                27      2.15      31      0
11        2.29        36       0                28      2.54      28      0
12        2.35        29       0                29      3.93      32      1
13        5.06        37       1                30      3.34      30      0
14        3.34        32       1                31      2.99      36      0
15        2.38        37       1                32      3.32      35      0
```

(8) *Example 1.8, p.9 : Women's role in society*

L'exemple concerne le du rôle de la femme dans la société, ou plus précisément de l'attitude vis à vis de la femme au foyer. Ces données sont issues d'enquêtes d'opinion.

```
> role_data.frame(
+   educ=rep(0:20,2),
+   sexe=factor(rep(1:2,c(21,21)),labels=c("hommes","femmes")),
+   accord=c(4,2,4,6,5,13,25,27,75,29,32,36,115,31,28,9,15,3,1,2,3,4,1,0,6,
+   10,14,17,26,91,30,55,50,190,17,18,7,13,3,0,1,2),
+   desaccord=c(2,0,0,3,5,7,9,15,49,29,45,59,245,70,79,23,110,29,28,13,20,
+   2,0,0,1,0,7,5,16,36,35,67,62,403,92,81,34,115,28,21,2,4))
```

```
> role                                     21  20 hommes     3        20
   educ   sexe accord desaccord            22   0 femmes     4         2
 1    0 hommes      4         2            23   1 femmes     1         0
 2    1 hommes      2         0            24   2 femmes     0         0
 3    2 hommes      4         0            25   3 femmes     6         1
 4    3 hommes      6         3            26   4 femmes    10         0
 5    4 hommes      5         5            27   5 femmes    14         7
 6    5 hommes     13         7            28   6 femmes    17         5
 7    6 hommes     25         9            29   7 femmes    26        16
 8    7 hommes     27        15            30   8 femmes    91        36
 9    8 hommes     75        49            31   9 femmes    30        35
10    9 hommes     29        29            32  10 femmes    55        67
11   10 hommes     32        45            33  11 femmes    50        62
12   11 hommes     36        59            34  12 femmes   190       403
13   12 hommes    115       245            35  13 femmes    17        92
14   13 hommes     31        70            36  14 femmes    18        81
15   14 hommes     28        79            37  15 femmes     7        34
16   15 hommes      9        23            38  16 femmes    13       115
17   16 hommes     15       110            39  17 femmes     3        28
18   17 hommes      3        29            40  18 femmes     0        21
19   18 hommes      1        28            41  19 femmes     1         2
20   19 hommes      2        13            42  20 femmes     2         4
```

(9) *Example 1.9, p.11 : Prostatic cancer*

```
> prostate_data.frame(
```

```
+ age=c(66,68,66,56,58,60,65,60,50,49,61,58,51,67,67,51,56,60,52,56,67,63,
+   59,64,61,56,64,61,64,63,52,66,58,57,65,65,59,61,53,67,53,65,50,60,45,56,
+   46,67,63,57,51,64,68),
+   acide=c(.48,.56,.5,.52,.5,.49,.46,.62,.56,.55,.62,.71,.65,.67,.47,.49,
+   .5,.78,.83,.98,.52,.75,.99,1.87,1.36,.82,.4,.5,.5,.4,.55,.59,.48,.51,
+   .49,.48,.63,1.02,.76,.95,.66,.84,.81,.76,.7,.78,.7,.67,.82,.67,.72,.89,
+   1.26),
+   X=c(0,0,0,0,0,0,1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,
+   1,0,0,1,0,0,0,0,1,1,1,0,1,0,0,0,0,1,1,1),
+   taille=c(rep(0:1,c(26,27))),
+   grade=c(0,0,0,0,0,0,0,0,1,0,0,0,0,1,1,0,1,0,0,0,0,0,1,0,0,0,1,0,1,0,1,
+   1,0,1,0,1,1,0,0,0,1,1,1,1,1,1,0,0,0,1,0,0,1),
+   Y=c(0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0,1,
+   1,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1))

> prostate
   age acide X taille grade Y
 1  66  0.48 0      0     0 0
 2  68  0.56 0      0     0 0
 3  66  0.50 0      0     0 0
 4  56  0.52 0      0     0 0
 5  58  0.50 0      0     0 0
 6  60  0.49 0      0     0 0
 7  65  0.46 1      0     0 0
 8  60  0.62 1      0     0 0
 9  50  0.56 0      0     1 1
10  49  0.55 1      0     0 0
11  61  0.62 0      0     0 0
12  58  0.71 0      0     0 0
13  51  0.65 0      0     0 0
14  67  0.67 1      0     1 1
15  67  0.47 0      0     1 0
16  51  0.49 0      0     0 0
17  56  0.50 0      0     1 0
18  60  0.78 0      0     0 0
19  52  0.83 0      0     0 0
20  56  0.98 0      0     0 0
21  67  0.52 0      0     0 0
22  63  0.75 0      0     0 0
23  59  0.99 0      0     1 1
24  64  1.87 0      0     0 0
25  61  1.36 1      0     0 1
26  56  0.82 0      0     0 1
27  64  0.40 0      1     1 0
28  61  0.50 0      1     0 0
29  64  0.50 0      1     1 0
30  63  0.40 0      1     0 0
31  52  0.55 0      1     1 0
32  66  0.59 0      1     1 0
33  58  0.48 1      1     0 1
34  57  0.51 1      1     1 1
35  65  0.49 0      1     0 1
36  65  0.48 0      1     1 0
37  59  0.63 1      1     1 0
38  61  1.02 0      1     0 0
39  53  0.76 0      1     0 0
40  67  0.95 0      1     0 0
41  53  0.66 0      1     1 0
42  65  0.84 1      1     1 1
43  50  0.81 1      1     1 1
44  60  0.76 1      1     1 1
45  45  0.70 0      1     1 1
46  56  0.78 1      1     1 1
47  46  0.70 0      1     0 1
48  67  0.67 0      1     0 1
49  63  0.82 0      1     0 1
50  57  0.67 0      1     1 1
51  51  0.72 1      1     0 1
52  64  0.89 1      1     0 1
53  68  1.26 1      1     1 1
```

(10) *Example 1.10, p.13 : Diverticular disease*

Chapitre 2

Statistical inference for binary data

2.1 The binomial distribution

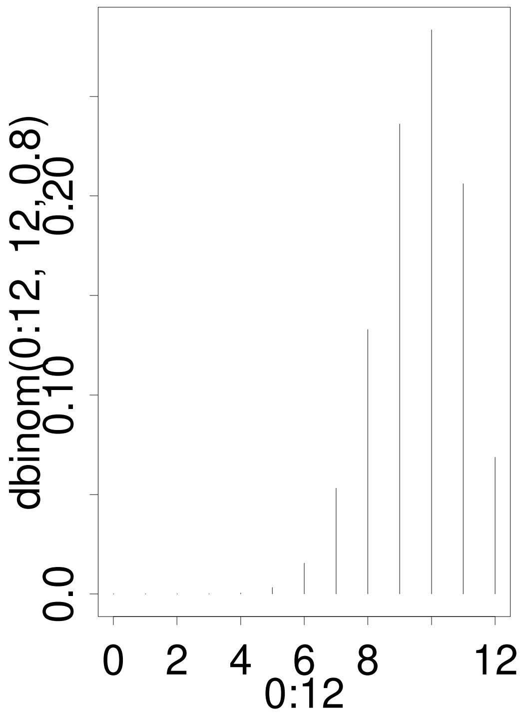
2.2 Inference about the success probability

Un marchand de graines annonce une probabilité de germination égale à 0.8. Sur un échantillon de 12 graines mises à germer, 7 graines ont germé. La probabilité estimée est $7/12 = 0.583$, ce qui est peu encourageant. Peut-on accepter l'hypothèse d'une probabilité égale à 0.8 ?
H0 : "$p = 0.8$"
H1 : "$p < 0.8$" (test unilatéral)

Test exact

```
> dbinom(0:12,12,0.8)                         # densite loi Binomiale(12,0.8)
 [1] 4.096000e-09 1.966080e-07 4.325376e-06 5.767168e-05 5.190451e-04
 [6] 3.321889e-03 1.550215e-02 5.315022e-02 1.328756e-01 2.362232e-01
[11] 2.834678e-01 2.061584e-01 6.871948e-02

> round(dbinom(0:12,12,0.8),3)
 [1] 0.000 0.000 0.000 0.000 0.001 0.003 0.016 0.053 0.133 0.236 0.283 0.206
[13] 0.069

> cbind(0:12,round(dbinom(0:12,12,0.8),3))
      [,1]  [,2]
 [1,]    0 0.000
 [2,]    1 0.000
 [3,]    2 0.000
 [4,]    3 0.000
 [5,]    4 0.001
 [6,]    5 0.003
 [7,]    6 0.016
 [8,]    7 0.053
 [9,]    8 0.133
[10,]    9 0.236
[11,]   10 0.283
[12,]   11 0.206
[13,]   12 0.069

> cbind(0:12,round(cumsum(dbinom(0:12,12,0.8)),3))    # densite cumulee
      [,1]  [,2]
 [1,]    0 0.000
```

```
 [2,]    1 0.000
 [3,]    2 0.000
 [4,]    3 0.000
 [5,]    4 0.001
 [6,]    5 0.004
 [7,]    6 0.019
 [8,]    7 0.073
 [9,]    8 0.205
[10,]    9 0.442
[11,]   10 0.725
[12,]   11 0.931
[13,]   12 1.000

> plot(0:12,dbinom(0:12,12,0.8),type="h")
```



Supposons que l'hypothèse nulle H_0 est vraie, c'est à dire que la probabilité de germer est réellement égale à 0.8. On suppose aussi implicitement que la loi de la variable aléatoire "nombre de graines qui germent parmi les 12 graines mises à germer" est binomiale, autrement dit que les graines sont bien indépendantes les unes des autres. Si je mets 12 graines à germer, la probabilité d'obtenir exactement 9 graines qui ont germé vaut 0.24, la probabilité d'obtenir 0, 1, 2, 3, 4, 5, 6, ou 7 graines qui germent vaut 0.07.

Donc, sous l'hypothèse H_0 , la probabilité d'obtenir un nombre de germinations aussi petit que 7, vaut 0.07. La "p-value" de mon test vaut 0.07. C'est une estimation du risque d'erreur si je rejette H_0. Si je me suis fixé un risque de première espèce α égal à 5%, ces données ne me permettent pas de rejeter H_0, elles ne me permettent pas de remettre en cause la parole du marchand. Par contre, si j'accepte un risque d'erreur égal à 10%, je remets en cause l'hypothèse nulle.

J'ai 7% de chances de me tromper si j'affirme que le marchand a surestimé sa probabilité de germination. On peut donc avoir des doutes, et soupçonner une surestimation de la probabilité de germination, sans qu'on puisse la mettre en évidence de façon certaine.

test approché (basé sur une approximation normale)

```
> Z_(7/12 - 0.8) / sqrt(0.8*(1-0.8)/12)   # statistique de test

> pnorm(Z)
[1] 0.03030098
```

La pvalue vaut 0.03. Je peux donc rejeter H_0 avec un risque d'erreur faible.

2.3 Comparison of two proportions

Un premier groupe de souris a été exposé à la fumée de cigarette, un deuxième groupe est un groupe contrôle. Sur chaque souris, après un an, on observe ou pas le développement d'une tumeur.

```
> souris
           cigarette controle
    tumeur        21       19
pas tumeur         2       13

> n1_23
> n2_32
> p1_21/n1
> p2_19/n2
```

H_0 : "$p_1 = p_2$"
H_1 : "$p_1 \neq p_2$" (test bilatéral)
On met en œuvre un test bilatéral, comme le fait Collett, mais un test unilatéral pourrait être choisi ici.

méthode approchée, test direct

```
> diff_p1-p2                                   # difference entre les proba
> se.diff_sqrt( p1*(1-p1)/n1 + p2*(1-p2)/n2 )  # ecart-type de la difference
> Z_diff/se.diff                               # statistique de test
> 2*(1-pnorm(Z))                               # pvalue 0.002
[1] 0.002320961
```

On calcule l'intervalle de confiance.

```
> z_qnorm(0.975)                               # 1.96
> borne.inf_diff-z*se.diff
> borne.sup_diff+z*se.diff                     # I.C. [0.11,0.52]
```

On peut aussi calculer l'écart-type de la différence des probabilités en remplaçant les probabilités de chacun des groupes par la probabilité commune.

```
> p_(21+19)/(n1+n2)
> se.diff_sqrt( p*(1-p)* (1/n1+1/n2))          # autre calcul de l'ecart-type
> Z_diff/se.diff
> 2*(1-pnorm(Z))                               # pvalue : 0.009
 [1] 0.008725761
```

méthode directe

```
> fisher.test(souris)

         Fisher's exact test
```

```
data:  souris
p-value = 0.013
alternative hypothesis: two.sided

> chisq.test(souris,correct=F)

         Pearson's chi-square test without Yates' continuity correction

data:  souris
X-squared = 6.8781, df = 1, p-value = 0.0087

> Z**2         # Z**2 est egal a la statistique  du Chi2.
[1] 6.878114
```

2.4 Comparison of two or more proportions

On cherche à comparer la proportion de plants de pruniers qui “prennent” ou pas selon qu’ils sont plantés immédiatement ou au printemps, et selon que les racines sont courtes ou longues.

```
> prune
       precode.court precoce.long tardif.court tardif.long
vivant           107          156           31          84
  mort           133           84          209         156

> chisq.test(prune)
         Pearson's chi-square test without Yates' continuity correction
data:  prune
X-squared = 141.0527, df = 3, p-value = 0
```

Collett présente un autre exemple (Collett, exemple 2.6, p.41) qui montre qu’il est important de ne pas “pooler” les données, c’est à dire de ne pas négliger un éventuel facteur de variation. Le risque est d’aboutir à la conclusion contraire à celle obtenue en considérant tous les facteurs de variation.

```
> enfants_matrix(c(3,176,4,293,17,197,2,23),nrow=2)
> dimnames(enfants)_list(c("mort","vivant"),
+                        paste(c("A","A","B","B"),rep(c("<1mois",">1mois"),2)))

> enfants
       A <1mois A >1mois B <1mois B >1mois
  mort        3        4       17        2
vivant      176      293      197       23

> chisq.test(enfants,correct=F)                    # cliniques A et B distinguees
         Pearson's chi-square test without Yates' continuity correction
data:  enfants
X-squared = 19.0964, df = 3, p-value = 3e-04

> chisq.test(enfants[,1:2],correct=F)              # clinique A
         Pearson's chi-square test without Yates' continuity correction
data:  enfants[, 1:2]
X-squared = 0.0835, df = 1, p-value = 0.7726

> chisq.test(enfants[,3:4],correct=F)              # clinique B
         Pearson's chi-square test without Yates' continuity correction
data:  enfants[, 3:4]
X-squared = 1e-04, df = 1, p-value = 0.9922
```

```
> chisq.test(enfants[,1:2]+enfants[,3:4],correct=F) # cliniques A et B poolees
         Pearson's chi-square test without Yates' continuity correction
data:  enfants[, 1:2] + enfants[, 3:4]
X-squared = 5.2555, df = 1, p-value = 0.0219
```

Pour chacune des cliniques, on ne détecte pas d'effet de la durée du soin prénatal sur la mortalité. Si on regroupait les données des deux cliniques, sans tenir compte du facteur clinique, on aboutirait à un effet de la durée du soin prénatal, et à une conclusion erronée. Ceci est vraisemblablement ? (c'est mon interprétation, Collett ne le dit pas, est-ce vrai ?) dû à une différence des probabilités entre les cliniques.
Le choix a été fait de ne pas utiliser la correction de continuité (correct=F) pour obtenir les mêmes résultats que Collett.

odds-ratio

```
> odds.ratio_function(tab,alpha=0.95)
+   {odr_(tab[1,1]*tab[2,2])/(tab[1,2]*tab[2,1])
+    se.log.odr_sqrt(sum(1/tab))
+    IC.log.odr_c(log(odr)-qnorm((1-alpha)/2)*se.log.odr,
+                 log(odr)+qnorm((1-alpha)/2)*se.log.odr)
+    IC.odr_exp(IC.log.odr)
+    list(odds.ratio=odr,IC.odds.ratio=IC.odr)
+   }


> odds.ratio(souris)
$odds.ratio:
[1] 7.184211

$IC.odds.ratio:
[1] 36.044384  1.431926
```

L'intervalle de confiance [1.4,36] ne contient pas la valeur 1. La probabilité de tumeur est donc plus élevée pour les souris exposées que pour les autres.

Chapitre 3

Models for binary and binomial data

3.1 Statistical modelling

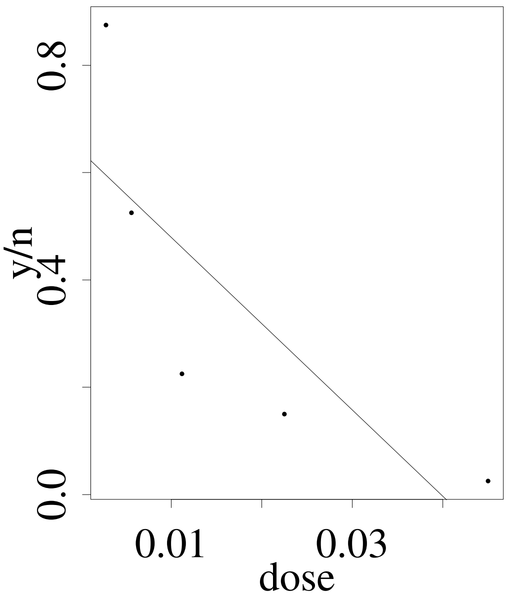
3.2 Linear models

3.3 Methods of estimation

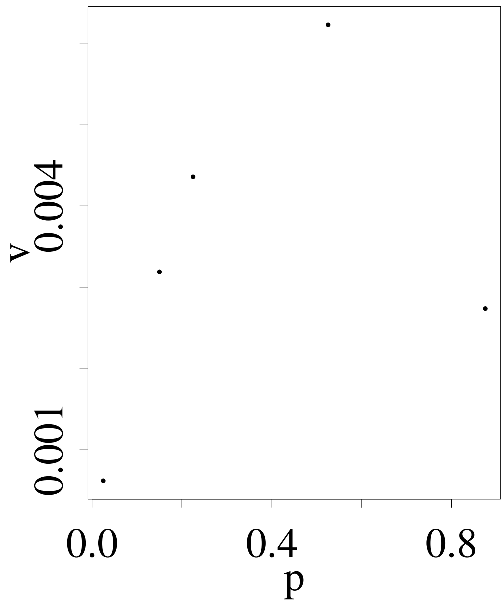
3.4 Fitting linear models to binomial data

L'exemple choisi est celui du sérum anti-pneumocoque (Collett, exemple 1.6, p.6). Cinq doses de sérum sont expérimentées. Chacune des doses est administrée à un lot de 40 souris. Les souris succombent ou pas.

```
> lm1_lm(y/n~dose,data=serum)

> coef(lm1)
 (Intercept)     dose
   0.6400247 -16.0749

> fitted(lm1)      # proba estimee negative, pour la dose la plus elevee.
        1         2         3         4           5
 0.595015 0.5500053 0.4599858 0.2783395 -0.08334561

> attach(serum)
> plot(dose,y/n)   # graphe des donnees
> abline(lm1)      # superposition de la droite estimee
> detach()
```




```
> serum$p_serum$y/serum$n
> attach(serum)
> v_p*(1-p)/n
> plot(p,v)        # graphe des variances en fonction des proba.
> detach()
```



L'utilisation du modèle linéaire pose deux problèmes : le modèle linéaire ne tient pas compte de l'hétérogénéité des variances, il ne tient pas compte non plus de la contrainte de l'appartenance à l'intervalle [0,1] des probabilités estimées.

3.5 Models for binomial response data

3.5.1 The logistic transformation

La transformation logistique d'une probabilité consiste à prendre $\log(p/(1-p))$.

3.5.2 The probit transformation

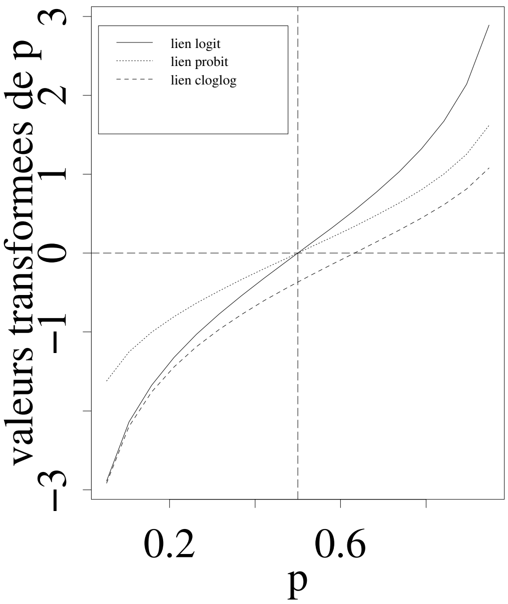
La transformation probit d'une probabilité p est définie comme étant la valeur ξ telle que:

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\xi} \exp(-\frac{1}{2}u^2)du = p$$

3.5.3 The complementary log-log transformation

La transformation cloglog d'une probabilité p est égale à $\log(-\log(1-p))$.

```
> p_seq(0,1,length=20)

> binomial()$link
function(mu)
log(mu/(1 - mu))

> binomial(link=logit)$link
function(mu)
log(mu/(1 - mu))

> binomial(link=probit)$link
function(mu)
qnorm(mu)

> binomial(link=cloglog)$link
function(mu)
log( - log(1 - mu))


> plot(p,binomial(link=logit)$link(p),type="l",
+                 ylab="valeurs transformees de p",xlab="p")
> lines(p,binomial(link=probit)$link(p),lty=2)
> lines(p,binomial(link=cloglog)$link(p),lty=3)
> abline(h=0,lty=4)
> abline(v=0.5,lty=4)
> legend(locator(2),lty=1:3,paste("lien",c("logit","probit","cloglog")))
```



Les fonctions de lien les plus couramment utilisées dans le cas de données binaires sont ici représentées. Elles "envoient" l'intervalle]0,1[sur l'intervalle $]-\infty,+\infty[$.

3.6 The linear logistic model

3.7 Fitting the linear logistic model to binomial data

L'exemple illustratif est l'exemple "Aircraft fasteners" (Collett, exemple 1.5, p.5) de la résistance des rivets en alliage utilisés dans l'aviation ("the compressive strength of an alloy fastener used in the construction of aircraft").

```
> glm1_glm(cbind(k,n-k)~load,family=binomial,data=aircraft)

> summary(glm1)
Call: glm(formula = cbind(k, n - k) ~ load, family = binomial, data = aircraft)
Deviance Residuals:
       Min         1Q     Median         3Q       Max
 -0.294752 -0.1112901 0.04161701 0.08846694 0.3501585

Coefficients:
                   Value   Std. Error   t value
(Intercept) -5.339711512 0.5456932048 -9.785190
       load  0.001548434 0.0001575379  9.828958

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 112.8321 on 9 degrees of freedom

Residual Deviance: 0.3719169 on 8 degrees of freedom

Number of Fisher Scoring Iterations: 3
```

```
Correlation of Coefficients:
     (Intercept)
load -0.9884009
```

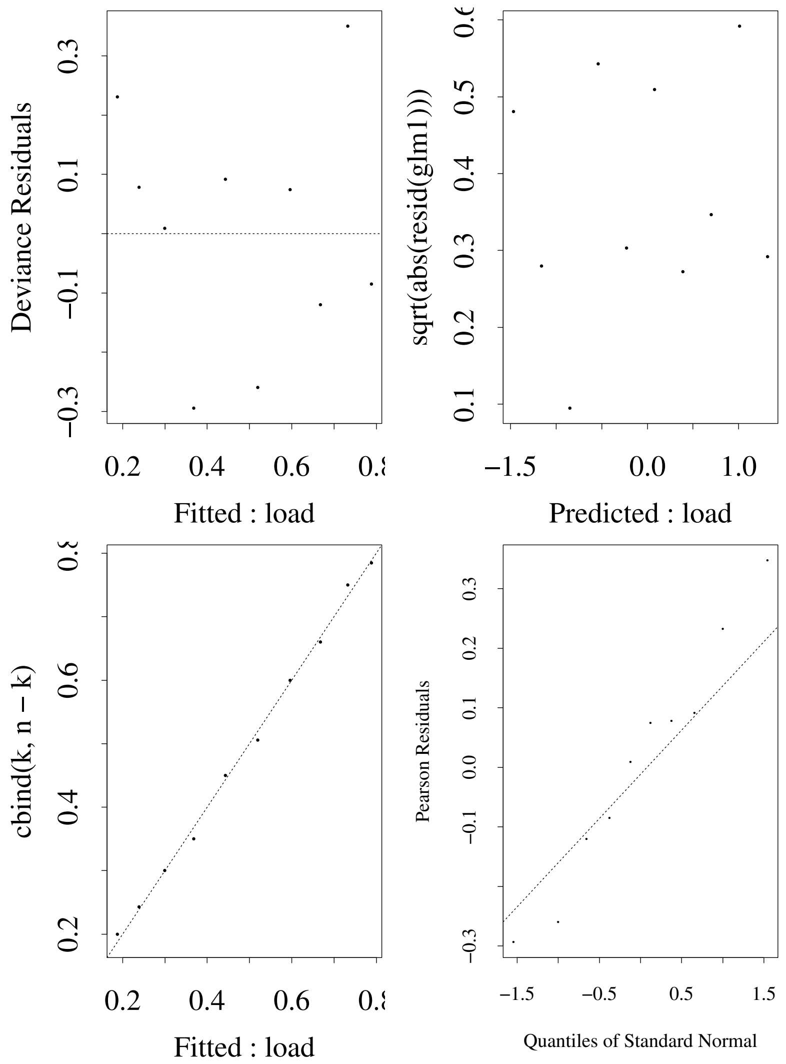
Le modèle ci-dessous est une autre paramétrisation du même modèle. La variable explicative a été centrée. Le terme constant correspond à l'ordonnée au centre du domaine de variation des abscisses, et non plus à l'ordonnée à l'origine, comme dans le modèle précédent. L'avantage est de rendre la corrélation entre les deux paramètres presque nulle. Les deux paramètres estimés peuvent donc être interprétés indépendamment l'un de l'autre.
Les modèles glm1 et glm1.1 sont équivalents : la déviance résiduelle est la même, les probabilités ajustées sont les mêmes.

```
> glm1.1_glm(cbind(k,n-k)~poly(load,1),family=binomial,data=aircraft)

> summary(glm1.1)
Call: glm(formula = cbind(k, n - k) ~ poly(load, 1), family = binomial,
        data =  aircraft)
Deviance Residuals:
       Min         1Q     Median         3Q       Max
 -0.294752 -0.1112901 0.04161701 0.08846694 0.3501585

Coefficients:
                    Value Std. Error    t value
  (Intercept) -0.07503716 0.08295692 -0.9045317
poly(load, 1)  2.81286938 0.28618185  9.8289581

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 112.8321 on 9 degrees of freedom

Residual Deviance: 0.3719169 on 8 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
              (Intercept)
poly(load, 1) -0.0450202

> attach(aircraft)
> predicteur_predict(glm1)       # dans l'echelle du lien, par defaut.
> proba.ajustee_exp(predicteur)/(1+exp(predicteur))

> predict(glm1,type="response")  # dans l'echelle initiale.
        1         2         3         4         5        6         7
 0.1871513 0.2388598 0.2995894 0.3682883 0.4427816 0.519941 0.5961606
        8         9        10
 0.6680059 0.7327982 0.7889409

> cbind(load,predicteur,proba.ajustee,n*proba.ajustee)
   load predicteur proba.ajustee
 1 2500 -1.4686274     0.1871513  9.357566
 2 2700 -1.1589407     0.2388598 16.720187
 3 2900 -0.8492540     0.2995894 29.958938
 4 3100 -0.5395673     0.3682883 22.097295
 5 3300 -0.2298805     0.4427816 17.711265
 6 3500  0.0798062     0.5199410 44.194982
```

```
 7 3700  0.3894929      0.5961606 53.654456
 8 3900  0.6991797      0.6680059 33.400293
 9 4100  1.0088664      0.7327982 58.623859
10 4300  1.3185531      0.7889409 51.281157
```

La fonction plot permet d'obtenir quelques représentations graphiques utiles pour valider un modèle. Elle reconnait la classe "glm" de l'objet fourni en argument et appelle la fonction plot.glm.

```
> par(mfrow=c(2,2))
> plot(glm1)
```

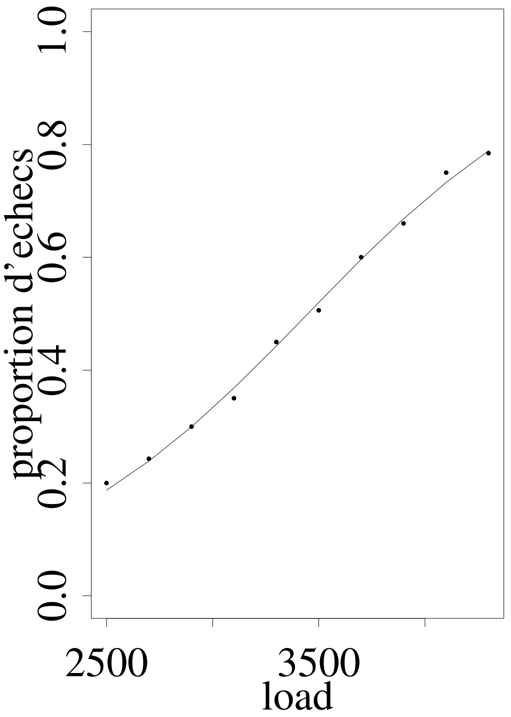



```
> class(glm1)
[1] "glm" "lm"
```

```
> ?plot.glm
```

On représente la courbe estimée et les observations.

```
> plot(load,k/n,ylim=c(0,1),ylab="proportion d'echecs",xlab="load")
> lines(load,fitted(glm1))
```



autre exemple : "les souris qui fument"

L'exemple "souris" a déjà été présenté au chapitre précédent (p. 105).

```
> souris1_data.frame(t(souris))
> souris1$groupe_factor(row.names(souris1))

> souris1
          tumeur pas.tumeur    groupe
cigarette     21          2 cigarette
 controle     19         13  controle
```

Le modèle ci-dessous suppose l'égalité des probabilités de tumeur entre les deux groupes "cigarette" et "contrôle". La probabilité commune, estimée par maximum de vraisemblance, est équivalente à la proportion observée.

```
> glm2_glm(cbind(tumeur,pas.tumeur)~1,family=binomial,data=souris1)

> summary(glm2)
Call: glm(formula = cbind(tumeur, pas.tumeur) ~ 1, family = binomial, data = souris1)
Deviance Residuals:
 cigarette   controle
 0.4654117 -0.2879272

Coefficients:
                Value Std. Error  t value
(Intercept) 0.9808293  0.3310611 2.962684

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 7.634852 on 1 degrees of freedom
```

```
Residual Deviance: 7.634852 on 1 degrees of freedom

Number of Fisher Scoring Iterations: 0

> attach(souris1)

> cbind(tumeur,pas.tumeur,tumeur/(tumeur+pas.tumeur),predict(glm2),fitted(glm2))
          tumeur pas.tumeur
cigarette     21          2 0.9130435 0.9808293 0.7272727
 controle     19         13 0.5937500 0.9808293 0.7272727

> (21+19)/(23+32)
[1] 0.7272727
```

Le modèle suivant tient compte de l'effet groupe. Si l'on compare les paramètres estimés avec ceux que donne Collett, on ne trouve pas les mêmes valeurs. Pourquoi ?

```
> glm3_glm(cbind(tumeur,pas.tumeur)~groupe,family=binomial,data=souris1)

> summary(glm3)
Call: glm(formula = cbind(tumeur, pas.tumeur) ~ groupe, family = binomial,
           data =  souris1)

Coefficients:
                 Value Std. Error   t value
(Intercept)  1.3654324  0.4114528  3.318564
     groupe -0.9859428  0.4114528 -2.396248

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 7.634852 on 1 degrees of freedom

Residual Deviance: 0 on 0 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
       (Intercept)
groupe -0.6173665
```

Parce que les contrastes utilisés sont ici ceux que Splus choisit par défaut, les contrastes de Helmert, pour les facteurs dont les niveaux ne sont pas ordonnés. Ces contrastes comparent chaque niveau à la moyenne des précédents, ils ont l'avantage d'être orthogonaux.

```
> options()$contrasts
          factor      ordered
 "contr.helmert" "contr.poly"

> glm3$contrasts
$groupe:
          [,1]
cigarette   -1
 controle    1
```

On choisit, comme Collett (et le logiciel GLIM) des contrastes de type "traitement". Le premier groupe (ici "cigarette") est considéré comme un groupe de référence. On obtient alors les mêmes estimations des paramètres. Ouf !

Le premier type de contrastes choisi par options(contrasts=...) est utilisé pour tous les facteurs dont les niveaux ne sont pas ordonnés, alors que le deuxième est utilisé pour les facteurs dont les niveaux sont déclarés ordonnés, cette déclaration peut être faite au moment de la création du facteur, ou plus tard.

```
> options(contrasts=c("contr.treatment","contr.poly"))
> glm3_glm(cbind(tumeur,pas.tumeur)~groupe,family=binomial,data=souris1)

> summary(glm3)
Call: glm(formula = cbind(tumeur, pas.tumeur) ~ groupe, family = binomial, data =
        souris1)

Coefficients:
                Value Std. Error   t value
(Intercept)  2.351375  0.7400129  3.177479
     groupe -1.971886  0.8229056 -2.396248

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 7.634852 on 1 degrees of freedom

Residual Deviance: 0 on 0 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
       (Intercept)
groupe -0.8992682

> glm3$contrasts
$groupe:
          controle
cigarette        0
 controle        1
```

Les probabilités estimées par maximum de vraisemblance sont égales aux proportions observées dans les deux groupes. Le modèle est saturé : on a autant de paramètres (2) que de traitements (2). Le terme traitement est pris au sens statistique de combinaison de niveaux de facteurs. C'est parce que le modèle est saturé que la déviance (résiduelle) est nulle.

```
> cbind(tumeur,pas.tumeur,tumeur/(tumeur+pas.tumeur),predict(glm3),fitted(glm3))
          tumeur pas.tumeur
cigarette     21          2 0.9130435 2.3513753 0.9130435
 controle     19         13 0.5937500 0.3794896 0.5937500

> 21/23
[1] 0.9130435

> 19/32
[1] 0.59375
```

Voici le même modèle, mais avec une autre paramétrisation : le terme constant est enlevé. L'avantage de cette paramétrisation est dans l'interprétation plus direct des paramètres estimés. Il n'y a plus de terme constant mais il y a autant de paramètres que de niveaux du facteur. Les paramètres estimés sont alors les transformées logit des $\widehat{p}$. Les $\widehat{p}$ sont donc obtenus en appliquant la transformation réciproque de la fonction de lien aux paramètres estimés.

```
> glm4_glm(cbind(tumeur,pas.tumeur)~-1+groupe,family=binomial,data=souris1)
```

```
> summary(glm4)
Call: glm(formula = cbind(tumeur, pas.tumeur) ~ -1 + groupe, family = binomial,
 data = souris1)

Coefficients:
                    Value Std. Error  t value
groupecigarette 2.3513753  0.7400129 3.177479
 groupecontrole 0.3794896  0.3599370 1.054322

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 19.42625 on 2 degrees of freedom

Residual Deviance: 0 on 0 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
               groupecigarette
groupecontrole 0

> binomial()$inverse(2.35)
[1] 0.9129342

> binomial()$inverse(0.38)
[1] 0.5938731

> exp(2.35)/(1+exp(2.35))
[1] 0.9129342

> exp(0.38)/(1+exp(0.38))
[1] 0.5938731
```

3.8 Goodness of fit of a linear logistic model

L'exemple est l'exemple "Aircraft fasteners" (p. 112). On a 10 observations binomiales, un modèle à 2 paramètres, donc 8 degrés de liberté. La déviance vaut 0.37. Elle suit sous l'hypothèse H_0 d'un modèle adéquat une loi de χ^2 à 8 degrés de liberté. Que penser de cette valeur 0.37 ?

```
> summary(glm1)
Call: glm(formula = cbind(k, n - k) ~ load, family = binomial, data = aircraft)
Deviance Residuals:
       Min         1Q     Median         3Q       Max
 -0.294752 -0.1112901 0.04161701 0.08846694 0.3501585

Coefficients:
                   Value   Std. Error   t value
(Intercept) -5.339711512 0.5456932048 -9.785190
       load  0.001548434 0.0001575379  9.828958

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 112.8321 on 9 degrees of freedom

Residual Deviance: 0.3719169 on 8 degrees of freedom
```

```
Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
     (Intercept)
load -0.9884009

> glm1$deviance
[1] 0.3719169

> glm1$df
[1] 8
```

C'est une valeur bien petite ! en effet, l'espérance d'un χ^2, c'est son degré de liberté. Approximativement, le modèle est adéquat lorsque la déviance est de l'ordre de grandeur du degré de liberté. Une déviance "trop grande" par rapport au degré de liberté est un indicateur d'un défaut d'adéquation du modèle.
Ici, la déviance est "suspecte". La probabilité d'obtenir 0.37 pour une variable aléatoire qui suit une loi de $\chi^2(8)$ vaut 10^{-5}. Le $1-\alpha$ quantile d'un $\chi^2(8)$ vaut 15.5 (avec $\alpha = 0.05$), ce qui signifie qu'une déviance inférieure ou égale à 15.5 conduit à accepter l'hypothèse d'adéquation du modèle.
Deux explications sont possibles : soit les données sont artificielles, soit les observations binaires sont corrélées négativement (ce qui conduirait à une sous-dispersion). Soit encore l'expérimentateur a un manque de chance évident ... mais à un tel point ! ... cette hypothèse est écartée.

```
> pchisq(0.37,8)
1] 4.211147e-05

> qchisq(.95,8)
[1] 15.50731
```

déviance et "scaled deviance"

Dans les exemples qui précèdent, déviance et "scaled deviance" sont identiques, car le paramètre d'échelle ("scale parameter" ou "dispersion parameter") est fixé, égal à 1. Ce choix correspond au choix de la distribution binomiale, qui suppose l'indépendance entre les observations binaires.
Dans certains cas, on n'a pas indépendance entre les observations binaires. Les observations peuvent être corrélées positivement (c'est le cas le plus courant) ou négativement. Une façon de résoudre ce problème est d'introduire un paramètre de dispersion ϕ tel que $Var(\mu) = \phi Var(Y)$, c'est à dire $Var(p) = \phi p(1-p)$, dans le cas d'une extension de la loi binomiale à des observations corrélées. Si $\phi > 1$ on parle de surdispersion, alors que si $\phi < 1$ on parle de sous-dispersion. Des corrélations positives induisent une surdispersion, des corrélations négatives induisent une sousdispersion (ce point est développé par Collett au chapitre 6).
La statistique du χ^2 de Pearson peut être obtenue en calculant la somme des carrés des résidus de Pearson. Dans cet exemple, elle est proche de la déviance. Des valeurs très différentes de ces deux statistiques (Chi2 de Pearson et déviance) sont des indicateurs de la mauvaise qualité de l'approximation du Chi2 (on est "loin" de l'asymptotique). On utilise la déviance pour aprécier la qualité de l'ajustement (lorsque les données sont groupées), parce que cette statistique est cohérente avec la méthode d'estimation (maximum de vraisemblance).

```
> sum(residuals(glm1,type="pearson")^2)
[1] 0.370663
```

3.9 Comparing linear logistics models

Comparer des modèles emboîtés - exemple prunier (p. 106)

```
> prune1_data.frame(t(prune))
```

```
> prune1$periode_factor(c(1,1,2,2),labels=c("precoce","tardif"))
> prune1$longueur_factor(c(1,2,1,2),labels=c("court","long"))
> glm5_glm(cbind(vivant,mort)~periode*longueur,binomial,prune1)
```

L'ajout de l'un ou l'autre des facteurs diminue significativement la déviance. La prise en compte des deux facteurs dans un modèle additif améliore encore le modèle par rapport à un modèle qui ne prendrait en compte qu'un facteur. L'ajout du terme d'interaction n'apport pas d'amélioration significative. Le modèle interactif complet est saturé (quatre paramètres sont ajustés pour quatre observations binomiales). On choisit le modèle additif.

```
> anova(glm5,test="Chi")
Analysis of Deviance Table
Binomial model
Response: cbind(vivant, mort)
Terms added sequentially (first to last)
                 Df Deviance Resid. Df Resid. Dev   Pr(Chi)
            NULL                     3   151.0193
         periode  1 97.57890         2    53.4404 0.0000000
        longueur  1 51.14657         1     2.2938 0.0000000
periode:longueur  1  2.29384         0     0.0000 0.1298883

> glm6_glm(cbind(vivant,mort)~1,binomial,prune1)

> anova(glm6,update(glm6,~.+longueur),test="Chi")
Analysis of Deviance Table
Response: cbind(vivant, mort)
     Terms Resid. Df Resid. Dev Test Df Deviance      Pr(Chi)
1        1         3   151.0193
2 longueur         2   105.1824       1  45.8369 1.285194e-11

> anova(glm6,update(glm6,~.+periode),
+            update(glm6,~.+longueur+periode),update(glm6,~.+longueur*periode),
+       test="Chi")
Analysis of Deviance Table
Response: cbind(vivant, mort)
                                  Terms Resid. Df Resid. Dev              Test
1                                     1         3   151.0193
2                               periode         2    53.4404
3                    longueur + periode         1     2.2938         +longueur
4 longueur + periode + longueur:periode         0     0.0000 +longueur:periode
  Df Deviance   Pr(Chi)
1
2  1 97.57890 0.0000000
3  1 51.14657 0.0000000
4  1  2.29384 0.1298883
```

Ce plan est équilibré : le nombre de répétitions par combinaison vaut 240. En modèle linéaire, un plan équilibré est un plan orthogonal : on a orthogonalité entre les sous-espaces vectoriels dans lesquels on projette le vecteur des observations. Les résultats (les sommes de carrés) ne dépendent pas de l'ordre d'introduction des termes dans le modèle, ils ne dépendent pas de l'ordre des projections successives. En GLM, ça n'est pas le cas. Même si un plan est équilibré, on n'a pas orthogonalité. Les déviances dépendent de l'ordre d'introduction des termes dans le modèle.
L'effet longueur, en ignorant l'effet période, est associé à une déviance égale à 45.84. L'effet longueur, ajusté par rapport à l'effet période, est associé à une déviance égale à 51.15.

```
> apply(prune,2,sum)
 precode.court precoce.long tardif.court tardif.long
```

```
          240          240          240         240

> anova(glm6,update(glm6,~.+longueur))
Analysis of Deviance Table
Response: cbind(vivant, mort)
     Terms Resid. Df Resid. Dev Test Df Deviance
1        1         3   151.0193
2 longueur         2   105.1824       1  45.8369

> anova(glm6,update(glm6,~.+periode))
Analysis of Deviance Table
Response: cbind(vivant, mort)
    Terms Resid. Df Resid. Dev Test Df Deviance
1       1         3   151.0193
2 periode         2    53.4404       1  97.5789

> anova(update(glm6,~.+periode),update(glm6,.~+periode+longueur))
Analysis of Deviance Table
Response: cbind(vivant, mort)
               Terms Resid. Df Resid. Dev      Test Df Deviance
1            periode         2   53.44041
2 periode + longueur         1    2.29384 +longueur  1 51.14657

> anova(update(glm6,~.+longueur),update(glm6,.~+longueur+periode))
Analysis of Deviance Table
Response: cbind(vivant, mort)
               Terms Resid. Df Resid. Dev     Test Df Deviance
1           longueur         2   105.1824
2 longueur + periode         1     2.2938 +periode  1 102.8886

> anova(update(glm6,~.+longueur+periode),update(glm6,.~+longueur*periode))
Analysis of Deviance Table
Response: cbind(vivant, mort)
                                  Terms Resid. Df Resid. Dev              Test
1                    longueur + periode         1   2.293839
2 longueur + periode + longueur:periode         0   0.000000 +longueur:periode
  Df Deviance
1
2  1 2.293839

> options(contrasts=c("contr.treatment","contr.poly"))
> glm7_glm(cbind(vivant,mort)~longueur+periode,binomial,prune1)

> summary(glm7)
Call: glm(formula = cbind(vivant, mort) ~ longueur + periode, family = binomial, data
        = prune1)
Deviance Residuals:
 precode.court precoce.long tardif.court tardif.long
     0.6641748   -0.6965749   -0.9392965   0.6965749

Coefficients:
                 Value Std. Error   t value
(Intercept) -0.3039203  0.1171647 -2.593959
   longueur  1.0176915  0.1454798  6.995416
    periode -1.4275424  0.1464621 -9.746835
```

```
(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 151.0193 on 3 degrees of freedom

Residual Deviance: 2.293839 on 1 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
         (Intercept)  longueur
longueur -0.5892778
 periode -0.4292804  -0.1576479

> predict(glm7)
 precode.court precoce.long tardif.court tardif.long
    -0.3039203    0.7137712    -1.731463  -0.7137712

> cbind(prune1,fitted(glm7))
              vivant mort periode longueur        X2
precode.court    107  133 precoce    court 0.4245994
 precoce.long    156   84 precoce     long 0.6712339
 tardif.court     31  209  tardif    court 0.1504006
  tardif.long     84  156  tardif     long 0.3287661
```

Comparer des modèles emboîtés - exemple ESR

L'exemple ESR est choisi (Collett, exemple 1.7, p. 8). ESR signifie “Erythrocyte sedimentation rate”. Il s'agit du taux de sédimentation des globules rouges. Un taux de sédimentation inférieur à un certain seuil est un signe de bonne santé (Y=1), alors qu'un taux supérieur au même seuil est un indicateur d'un problème (Y=0). On cherche à savoir si la variable Y ainsi définie peut être expliquée par les variables fibrinogène et γ-globuline.
Ici, les données sont binaires, la déviance ne peut pas être utilisée pour apprécier l'adéquation du modèle. La probabilité que le taux de sédimentation dépasse le seuil fixé ne semble dépendre que de la variable fibrinogène.

```
> glm8_glm(reponse~fibrinogene+globuline,binomial,ESR)
> glm8.1_glm(reponse~globuline+fibrinogene,binomial,ESR)

> anova(glm8,test="Chi")
Analysis of Deviance Table
Binomial model
Response: reponse

Terms added sequentially (first to last)
            Df Deviance Resid. Df Resid. Dev   Pr(Chi)
       NULL                    31   30.88496
fibrinogene  1 6.044608        30   24.84036 0.0139488
  globuline  1 1.869242        29   22.97111 0.1715619

> anova(glm8.1,test="Chi")
Analysis of Deviance Table
Binomial model
Response: reponse

Terms added sequentially (first to last)
            Df Deviance Resid. Df Resid. Dev   Pr(Chi)
```

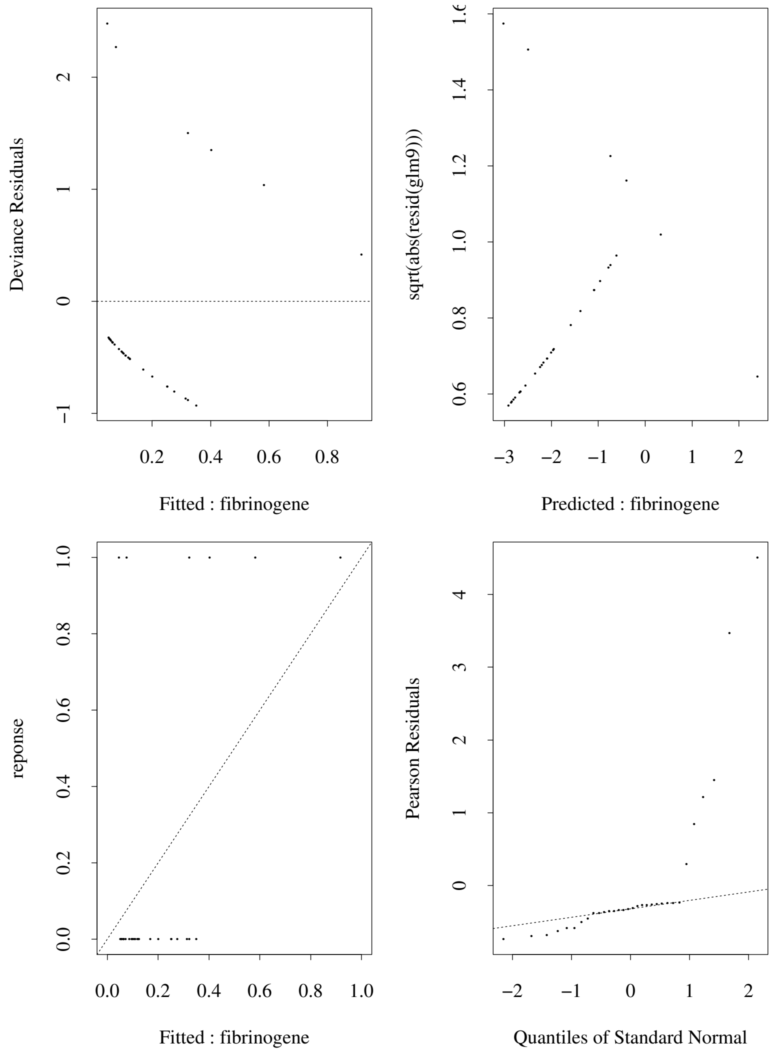
```
       NULL                   31   30.88496
  globuline  1  1.93953       30   28.94543 0.1637196
fibrinogene  1  5.97432       29   22.97111 0.0145157
```

Le modèle logistique ne faisant intervenir que la variable fibrinogène est ajusté.

```
> glm9_glm(reponse~fibrinogene,binomial,ESR)

> summary(glm9)
Call: glm(formula = reponse ~ fibrinogene, family = binomial, data = ESR)
Deviance Residuals:
        Min         1Q     Median         3Q      Max
 -0.9298272 -0.5399198 -0.4381977 -0.3356489 2.479406

Coefficients:
                Value Std. Error   t value
(Intercept) -6.844796  2.7543984 -2.485042
fibrinogene  1.826996  0.8960538  2.038935

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 30.88496 on 31 degrees of freedom

Residual Deviance: 24.84036 on 30 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
            (Intercept)
fibrinogene -0.9826252
```

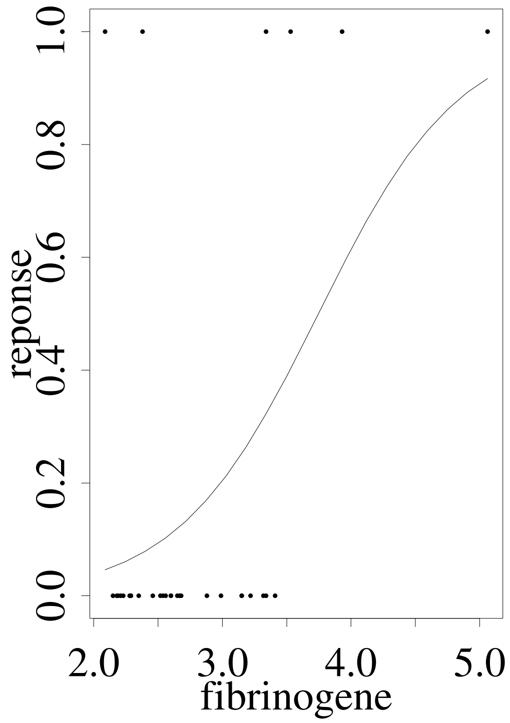
Les représentations graphiques obtenues à l'aide de la commande plot sont présentées. Elles ne sont pas informatives dans le cas de données binaires.

```
> par(mfrow=c(2,2))
> plot(glm9)
```



La fonction estimée est représentée graphiquement, ainsi que les observations binaires.

```
> attach(ESR)
> plot(fibrinogene,reponse)
> x_seq(min(fibrinogene),max(fibrinogene),length=20)
> lines(x,predict(glm9,data.frame(fibrinogene=x),type="response"))
> detach()
```



3.10 Linear trends in proportions

Exemple “toxicity of cypermethrin to moths”, résistance d’un papillon de nuit au pyrethroide trans-cypermethrine (la larve est un ravageur des cultures de coton) (Collett, exemple 3.7, p.75).

```
> insecte_data.frame(
> dose=rep(c(1,2,4,8,16,32),2),
> sexe=factor(rep(1:2,c(6,6)),labels=c("male","femelle")),
> k=c(1,4,9,13,18,20,0,2,6,10,12,16),
> n=rep(20,12))

> insecte
   dose    sexe  k  n
 1    1    male  1 20
 2    2    male  4 20
 3    4    male  9 20
 4    8    male 13 20
 5   16    male 18 20
 6   32    male 20 20
 7    1 femelle  0 20
 8    2 femelle  2 20
 9    4 femelle  6 20
10    8 femelle 10 20
11   16 femelle 12 20
12   32 femelle 16 20

> glm10_glm(cbind(k,n-k)~sexe+factor(dose),binomial,insecte)
> glm10.1_glm(cbind(k,n-k)~factor(dose)+sexe,binomial,insecte)

> anova(glm10,test="Chi")
```

```
Analysis of Deviance Table
Binomial model
Response: cbind(k, n - k)

Terms added sequentially (first to last)
             Df Deviance Resid. Df Resid. Dev    Pr(Chi)
        NULL                    11   124.8756
        sexe  1   6.0770        10   118.7986 0.01369548
factor(dose)  5 113.7858         5     5.0128 0.00000000

> anova(glm10.1,test="Chi")
Analysis of Deviance Table
Binomial model
Response: cbind(k, n - k)

Terms added sequentially (first to last)
             Df Deviance Resid. Df Resid. Dev     Pr(Chi)
        NULL                    11   124.8756
factor(dose)  5 109.7232         6    15.1524 0.000000000
        sexe  1  10.1395         5     5.0128 0.001451218
```

On a considéré la dose comme un facteur dont les niveaux sont non ordonnés. C'est dommage : on n'a pas tenu compte de toute l'information dont on dispose, du fait que les niveaux sont ordonnés. La dose est en fait une variable explicative quantitative. On peut inclure dans le modèle des effets linéaires (dans l'échelle du lien), quadratique, ou d'ordre plus élevé. Dans cet exemple, il faut aller jusqu'à un polynôme de degré 3.

```
> glm11.1_glm(cbind(k,n-k)~sexe+poly(dose,1),binomial,insecte)
> glm11.2_glm(cbind(k,n-k)~sexe+poly(dose,2),binomial,insecte)
> glm11.3_glm(cbind(k,n-k)~sexe+poly(dose,3),binomial,insecte)
> glm11.4_glm(cbind(k,n-k)~sexe+poly(dose,4),binomial,insecte)
> glm11.5_glm(cbind(k,n-k)~sexe+poly(dose,5),binomial,insecte)

> anova(glm11.1,glm11.2,glm11.3,glm11.4,glm11.5,test="Chi")
Analysis of Deviance Table
Response: cbind(k, n - k)
                 Terms Resid. Df Resid. Dev    Test Df Deviance   Pr(Chi)
1 sexe + poly(dose, 1)         9   27.96797
2 sexe + poly(dose, 2)         8   14.83517 1 vs. 2  1 13.13280 0.0002902
3 sexe + poly(dose, 3)         7    8.18010 2 vs. 3  1  6.65507 0.0098874
4 sexe + poly(dose, 4)         6    5.51521 3 vs. 4  1  2.66490 0.1025845
5 sexe + poly(dose, 5)         5    5.01282 4 vs. 5  1  0.50239 0.4784535
```

Les doses sont des puissances de 2. Si on transforme la variable explicative dose, en considérant le logarithme base 2 de la dose, un polynôme de degré 1 est suffisant. L'écart à la linéarité de la relation est non significatif.

```
> glm12.1_glm(cbind(k,n-k)~sexe+poly(log(dose,2),1),binomial,insecte)
> glm12.5_glm(cbind(k,n-k)~sexe+poly(log(dose,2),5),binomial,insecte)

> anova(glm12.1,glm12.5,test="Chi")
Analysis of Deviance Table
Response: cbind(k, n - k)
                         Terms Resid. Df Resid. Dev    Test Df Deviance
1 sexe + poly(log(dose, 2), 1)         9   6.757064
2 sexe + poly(log(dose, 2), 5)         5   5.012820 1 vs. 2  4 1.744244
    Pr(Chi)
```

```
1
2 0.7826658
```

L'effet linéaire de la dose est significatif.

```
> anova(glm12.1,test="Chi")
Analysis of Deviance Table
Binomial model
Response: cbind(k, n - k)

Terms added sequentially (first to last)
                      Df Deviance Resid. Df Resid. Dev    Pr(Chi)
                 NULL                    11   124.8756
                 sexe  1   6.0770        10   118.7986 0.01369548
poly(log(dose, 2), 1)  1 112.0415         9     6.7571 0.00000000
```

L'effet sexe, ajusté par rapport à l'effet dose est significatif.

```
> anova(glm12.5,update(glm12.5,~.-sexe),test="Chi")
Analysis of Deviance Table
Response: cbind(k, n - k)
                         Terms Resid. Df Resid. Dev  Test Df  Deviance
1 sexe + poly(log(dose, 2), 5)         5    5.01282
2        poly(log(dose, 2), 5)         6   15.15237 -sexe -1 -10.13955
      Pr(Chi)
1
2 0.001451218
```

L'effet dose ajusté par rapport à l'effet sexe est significatif.

```
> anova(glm12.5,update(glm12.5,~.-poly(log(dose,2),5)),test="Chi")
Analysis of Deviance Table
Response: cbind(k, n - k)
                         Terms Resid. Df Resid. Dev                   Test Df
1 sexe + poly(log(dose, 2), 5)         5     5.0128
2                         sexe        10   118.7986 -poly(log(dose, 2), 5) -5
   Deviance Pr(Chi)
1
2 -113.7858       0
```

3.11 Comparing stimulus-response relationships

L'exemple utilisé est celui du rôle de la femme dans la société, ou plus précisément de l'attitude vis à vis de la femme au foyer (Collett, exemple 1.8, p.10). Ces données sont issues d'enquêtes d'opinion. Chaque individu a répondu à la question “êtes vous d'accord ou pas avec l'assertion suivante : women should take care of running their homes and leave running the country up to men”.

```
> options(contrasts=c("contr.treatment","contr.poly"))
> glm13_glm(cbind(accord,desaccord)~-1+sexe/educ,binomial,role)

> summary(glm13)
Deviance Residuals:
       Min         1Q     Median        3Q      Max
 -2.390982 -0.8806163 0.01532345 0.7278331 2.452614

Coefficients:
                    Value Std. Error    t value
```

```
    sexehommes  2.0981950 0.23544332   8.911678
    sexefemmes  3.0029230 0.27201011  11.039748
sexehommeseduc -0.2340270 0.02018265 -11.595455
sexefemmeseduc -0.3154099 0.02361296 -13.357491

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 695.7281 on 42 degrees of freedom

Residual Deviance: 57.10269 on 37 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
               sexehommes sexefemmes sexehommeseduc
    sexefemmes  0.0000000
sexehommeseduc -0.9651050  0.0000000
sexefemmeseduc  0.0000000 -0.9778134  0.0000000
Warning messages:
  1 rows with zero weights not counted in: summary(glm13)
```

Bien que la déviance soit grande par rapport au degré de liberté, il est intéressant de la comparer à celle que l'on obtient en imposant l'hypothèse d'égalité des pentes entre hommes et femmes.

```
> glm14_glm(cbind(accord,desaccord)~-1+sexe+educ,binomial,role)

> summary(glm14)

Call: glm(formula = cbind(accord, desaccord) ~ -1 + sexe + educ, family = binomial,
        data = role)
Deviance Residuals:
       Min         1Q     Median       3Q      Max
 -2.725449 -0.8630174 -0.0652494 0.843397 3.133162

Coefficients:
                Value Std. Error   t value
sexehommes  2.5093670 0.18376256  13.65549
sexefemmes  2.4979200 0.18264773  13.67616
      educ -0.2706204 0.01539917 -17.57370

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 695.7281 on 42 degrees of freedom

Residual Deviance: 64.00664 on 38 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
           sexehommes sexefemmes
sexefemmes  0.8945960
      educ -0.9399417 -0.9517569
Warning messages:
  1 rows with zero weights not counted in: summary(glm14)
```

La différence entre les deux modèles est hautement significative, il est nécessaire de distinguer les deux pentes.

```
> anova(glm13,glm14,test="Chi")
```

```
Analysis of Deviance Table
Response: cbind(accord, desaccord)
             Terms Resid. Df Resid. Dev    Test Df  Deviance    Pr(Chi)
1   -1 + sexe/educ        37   57.10269
2 -1 + sexe + educ        38   64.00664 1 vs. 2 -1 -6.903948 0.00860056
```

La déviance du modèle glm13 est élevée. On recherche donc un défaut d'ajustement.

```
> 1-pchisq(glm13$deviance,glm13$df)
[1] 0.01847406
```

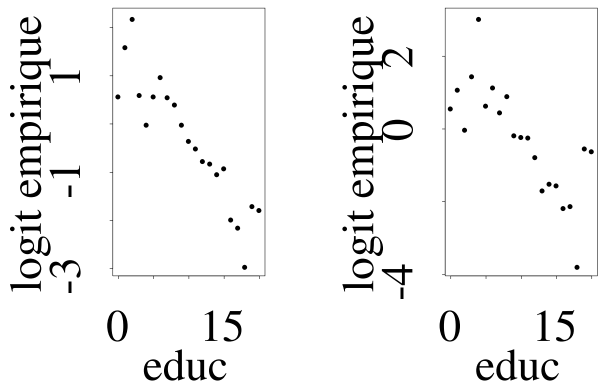
Le modèle est adéquat pour les hommes, il l'est beaucoup moins pour les femmes.

```
> glm14.h_glm(cbind(accord,desaccord)~educ,binomial,role,subset=sexe=="hommes")
> glm14.f_glm(cbind(accord,desaccord)~educ,binomial,role,subset=sexe=="femmes")

> summary(glm14.h)
Call: glm(formula = cbind(accord, desaccord) ~ educ, family = binomial, data = role,
        subset = sexe == "hommes")
Deviance Residuals:
       Min         1Q     Median        3Q     Max
 -1.789321 -0.6751499 0.03064691 0.7592683 1.19633

Coefficients:
                Value Std. Error    t value
(Intercept)  2.098195 0.23544332   8.911678
       educ -0.234027 0.02018265 -11.595455

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 182.9219 on 20 degrees of freedom

Residual Deviance: 18.94502 on 19 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
     (Intercept)
educ -0.965105

> summary(glm14.f)
Call: glm(formula = cbind(accord, desaccord) ~ educ, family = binomial, data = role,
        subset = sexe == "femmes")
Deviance Residuals:
       Min         1Q Median        3Q      Max
 -2.390982 -0.9606345      0 0.3628623 2.452614

Coefficients:
                 Value Std. Error   t value
(Intercept)  3.0029230 0.27201011  11.03975
       educ -0.3154099 0.02361296 -13.35749

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 268.7886 on 20 degrees of freedom

Residual Deviance: 38.15767 on 18 degrees of freedom
```

```
Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
     (Intercept)
educ -0.9778134
Warning messages:
  1 rows with zero weights not counted in: summary(glm14.f)
```

Pour mieux comprendre et tenter d'expliquer le fait que le modèle soit adéquat pour les hommes, et qu'il le soit beaucoup moins pour les femmes, on représente graphiquement les données dans l'échelle logit, en appliquant la transformation logit empirique aux réponses. En toute logique, ce graphique aurait dû être réalisé avant le premier ajustement.
L'hypothèse de linéarité de l'effet "nombre d'années d'éducation" semble plausible, pour les hommes comme pour les femmes. Les graphiques ne suggèrent aucune modification de la variable explicative. La valeur élevée de la déviance, pour les femmes, est vraisemblablement due aux faibles et aux fortes valeurs du nombre d'années d'éducation. Ces observations correspondent à des effectifs faibles. Cela suggère une explication possible : l'approximation Chi2 de la déviance n'est peut-être pas très bonne. Dans ce cas, la déviance ne serait pas une bonne mesure de la qualité d'ajustement. On conserve donc le modèle glm13, avec une pente qui dépend du sexe.

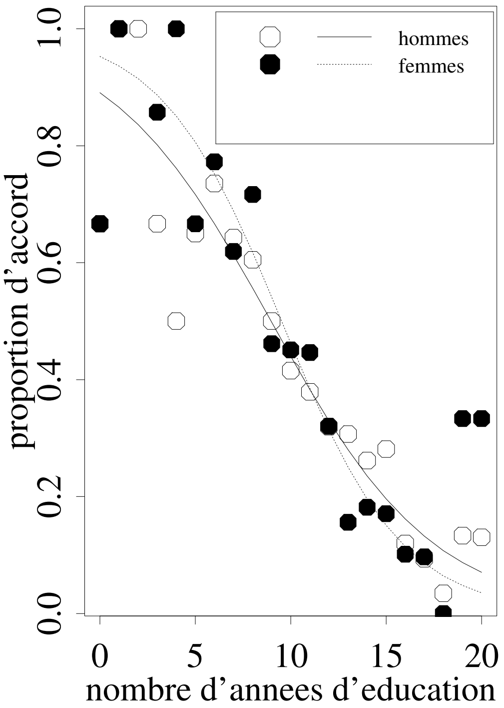
```
> role.h_role[role$sexe=="hommes",]
> role.f_role[role$sexe=="femmes",]

> par(mfrow=c(2,2))
> attach(role.h)
> plot(educ,log((accord+0.5)/(desaccord+0.5)),ylab="logit empirique",
                                                   sub="hommes")
> detach()
> attach(role.f)
> plot(educ,log((accord+0.5)/(desaccord+0.5)),ylab="logit empirique",
                                                   sub="femmes")
> detach()
```



On représente graphiquement les données, et le modèle retenu.

```
> par(mfrow=c(1,1))
> attach(role.h)
> plot(educ,accord/(accord+desaccord),type="n",
+      xlab="nombre d'annees d'education",ylab="proportion d'accord")
> points(educ,accord/(accord+desaccord),pch=1)
> lines(educ,fitted(glm14.h))
> detach()
> attach(role.f)
> points(educ,accord/(accord+desaccord),pch=16)
```

```
> lines(educ,fitted(glm14.f),lty=2)
> detach()
> legend(locator(2),lty=1:2,marks=c(1,16),c("hommes","femmes"))
```



3.12 Non-convergence and overfitting

On rencontre ce problème lorsque l'une au moins des proportions observées est exactement égale à 0 ou à 1. Le problème se pose donc lorsque les données sont "creuses" (sparse), lorsque le modèle est trop complexe par rapport à la quantité de données, lorsque le modèle est surparamétré. Les symptômes sont un message de non convergence, des écarts-type pour certains paramètres très grands. Cela suggère un modèle trop complexe.

```
> essai_data.frame(fact=factor(1:2),k=c(0,5),n=c(4,6))

> glm.essai_glm(cbind(k,n-k)~fact,binomial,essai)
Warning messages:
  linear convergence not obtained in  10  iterations. in: glm.fitter(x = X, y =
        Y, w = w, start = start, offset = offset, family = family, ....

> glm.essai_glm(cbind(k,n-k)~fact,binomial,essai,maxit=50)

> summary(glm.essai)
Call: glm(formula = cbind(k, n - k) ~ fact, family = binomial, data = essai, maxit =
        50)

Coefficients:
                Value Std. Error      t value
(Intercept) -21.15949   11927.44 -0.001774018
       fact  22.76893   11927.44  0.001908954

(Dispersion Parameter for Binomial family taken to be 1 )
```

```
    Null Deviance: 8.456209 on 1 degrees of freedom

Residual Deviance: 0 on 0 degrees of freedom

Number of Fisher Scoring Iterations: 19

Correlation of Coefficients:
     (Intercept)
fact -1

> fitted(glm.essai)
            1         2
 6.464743e-10 0.8333333
```

3.13 A futher example on model selection

L'exemple choisi est l'exemple du cancer de la prostate (Collett, exemple 1.9, p.11)
La distribution dissymétrique de la variable acide suggère l'utilisation de log(acide) à la place de la variable acide, comme régresseur.

```
> stem(prostate$acide)
N = 53   Median = 0.65
Quartiles = 0.5, 0.78

Decimal point is 1 place to the left of the colon

    4 : 0067888999
    5 : 0000012255669
    6 : 22356777
    7 : 001256688
    8 : 122349
    9 : 589
   10 : 2
   11 :
   12 : 6

High: 1.36 1.87

> stem(log(prostate$acide))
N = 53   Median = -0.430783
Quartiles = -0.693147, -0.2484614

Decimal point is 1 place to the left of the colon

  -9 : 22
  -8 :
  -7 : 86333111
  -6 : 9999975500
  -5 : 883
  -4 : 88632000
  -3 : 6643
  -2 : 97755100
  -1 : 972
  -0 : 521
   0 : 2
```

```
 1 :
 2 : 3
 3 : 1
 4 :
 5 :
 6 : 3
```

Le choix du logarithme de la variable acide comme régresseur est confirmé par la comparaison des déviances des modèles associés.

```
> glm15_glm(Y~acide,binomial,prostate)
> glm16_glm(Y~log(acide),binomial,prostate)

> summary(glm15)
Call: glm(formula = Y ~ acide, family = binomial, data = prostate)
Deviance Residuals:
       Min         1Q     Median       3Q      Max
 -2.014383 -0.9120228 -0.8164711 1.287079 1.597101

Coefficients:
                Value Std. Error   t value
(Intercept) -1.927027  0.9201944 -2.094152
      acide  2.040069  1.2551845  1.625314

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 70.25215 on 52 degrees of freedom

Residual Deviance: 67.11595 on 51 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
      (Intercept)
acide -0.9482855

> summary(glm16)
Call: glm(formula = Y ~ log(acide), family = binomial, data = prostate)
Deviance Residuals:
       Min         1Q     Median       3Q      Max
 -1.980231 -0.9094845 -0.7265506 1.195062 1.730228

Coefficients:
                Value Std. Error   t value
(Intercept) 0.4039863  0.5085514 0.7943864
 log(acide) 2.2445617  1.0385344 2.1612781

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 70.25215 on 52 degrees of freedom

Residual Deviance: 64.81272 on 51 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
           (Intercept)
```

```
log(acide) 0.8093297
```

Si on supprime la 24^e observation, l'intérêt de la transformation log pour la variable acide n'est plus évident.

```
> attach(prostate)
> stem(acide[Y==0],fence=10)
N = 33   Median = 0.55
Quartiles = 0.5, 0.71

Decimal point is 1 place to the left of the colon

    4 : 00678899
    5 : 00000225569
    6 : 22356
    7 : 1568
    8 : 3
    9 : 58
   10 : 2
   11 :
   12 :
   13 :
   14 :
   15 :
   16 :
   17 :
   18 : 7

> stem(acide[Y==1],fence=10)
N = 20   Median = 0.74
Quartiles = 0.67, 0.83

Decimal point is 1 place to the left of the colon

    4 : 89
    5 : 16
    6 : 777
    7 : 00268
    8 : 12249
    9 : 9
   10 :
   11 :
   12 : 6
   13 : 6

> (1:nrow(prostate)) [acide==1.87]
[1] 24
> detach()

> glm15.1_glm(Y~acide,binomial,prostate,subset=-24)
> glm16.1_glm(Y~log(acide),binomial,prostate,subset=-24)

> summary(glm15.1)
Call: glm(formula = Y ~ acide, family = binomial, data = prostate, subset = -24)
Deviance Residuals:
       Min         1Q     Median       3Q      Max
 -1.682713 -0.8437144 -0.6700625 1.099994 1.801655
```

```
Coefficients:
                Value Std. Error   t value
(Intercept) -3.661778   1.261624 -2.902432
      acide  4.705382   1.807927  2.602639

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 69.29296 on 51 degrees of freedom

Residual Deviance: 60.45243 on 50 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
      (Intercept)
acide -0.9691488

> summary(glm16.1)
Call: glm(formula = Y ~ log(acide), family = binomial, data = prostate, subset = -24)
Deviance Residuals:
       Min         1Q     Median       3Q      Max
 -1.648338 -0.8591557 -0.6246797 1.057664 1.876202

Coefficients:
                Value Std. Error  t value
(Intercept) 0.9922279  0.6014291 1.649784
 log(acide) 3.4926710  1.2608097 2.770181

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 69.29296 on 51 degrees of freedom

Residual Deviance: 59.7608 on 50 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
           (Intercept)
log(acide) 0.8537003
```

On n'a aucune raison médicale de supprimer le patient 24. De plus, la distribution de la variable acide, pour la population qui présente un nodule, semble dissymétrique, même sans l'individu 24. On conserve donc le patient 24, et on choisit log(acide) comme régresseur.
Il existe 32 modèles possibles pour ces données. On choisit une méthode ascendante de construction du modèle pas à pas. On introduit d'abord les effets principaux.

```
> glm17_glm(Y~1,binomial,prostate)
> glm17.1_step(glm17,~age+log(acide)+X+taille+grade,direction="forward")
 Start:  AIC= 72.2522
 Y ~ 1

Single term additions

Model:
Y ~ 1
```

```
scale:  1

           Df Sum of Sq      RSS       Cp
    <none>              53.00000 55.00000
       age  1   1.09362 51.90638 55.90638
log(acide)  1   5.32427 47.67573 51.67573
         X  1  11.28314 41.71686 45.71686
    taille  1   7.43833 45.56167 49.56167
     grade  1   4.07465 48.92535 52.92535

Step:  AIC= 63.0008
 Y ~ X

Single term additions

Model:
Y ~ X

scale:  1

           Df Sum of Sq      RSS       Cp
    <none>              52.92342 56.92342
       age  1  1.350338 51.57309 57.57309
log(acide)  1  3.791654 49.13177 55.13177
    taille  1  5.631195 47.29223 53.29223
     grade  1  2.367732 50.55569 56.55569

Step:  AIC= 59.3534
 Y ~ X + taille

Single term additions

Model:
Y ~ X + taille

scale:  1

           Df Sum of Sq      RSS       Cp
    <none>              53.98513 59.98513
       age  1  1.267386 52.71775 60.71775
log(acide)  1  4.578922 49.40621 57.40621
     grade  1  0.583726 53.40141 61.40141

Step:  AIC= 56.986
 Y ~ X + taille + log(acide)

Single term additions

Model:
Y ~ X + taille + log(acide)

scale:  1

       Df Sum of Sq      RSS       Cp
<none>              50.11233 58.11233
```

```
   age  1  1.298730 48.81360 58.81360
 grade  1  1.229871 48.88245 58.88245

> anova(glm17.1,test="Chi")
Analysis of Deviance Table
Binomial model
Response: Y

Terms added sequentially (first to last)
           Df Deviance Resid. Df Resid. Dev    Pr(Chi)
      NULL                    52   70.25215
         X  1 11.25135        51   59.00080 0.00079565
    taille  1  5.64738        50   53.35342 0.01748143
log(acide)  1  4.36741        49   48.98601 0.03663272

> anova(glm17.1,update(glm17.1,~.+grade),test="Chi")
Analysis of Deviance Table
Response: Y
                            Terms Resid. Df Resid. Dev   Test Df Deviance
1         X + taille + log(acide)        49   48.98601
2 X + taille + log(acide) + grade        48   47.77568 +grade  1 1.210329
    Pr(Chi)
1
2 0.2712669

> anova(glm17.1,update(glm17.1,~.+age),test="Chi")
Analysis of Deviance Table
Response: Y
                          Terms Resid. Df Resid. Dev Test Df Deviance  Pr(Chi)
1       X + taille + log(acide)        49   48.98601
2 X + taille + log(acide) + age        48   47.68068 +age  1  1.30533 0.253242
```

L'ajout de termes de degré 2, pour les variables âge et acide est-il nécessaire? non.

```
> glm18_glm(Y~age+log(acide)+X+taille+grade,binomial,prostate)
> glm19_glm(Y~poly(age,2)+poly(log(acide),2)+X+taille+grade,binomial,prostate)

> anova(glm18,glm19,test="Chi")
Analysis of Deviance Table
Response: Y
                                                  Terms Resid. Df Resid. Dev
1                   age + log(acide) + X + taille + grade        47   46.55990
2 poly(age, 2) + poly(log(acide), 2) + X + taille + grade        45   43.62121
     Test Df Deviance  Pr(Chi)
1
2 1 vs. 2  2 2.938683 0.230077
```

Est-il nécessaire d'introduire des termes d'interaction? oui, on trouve qu'il est intéressant d'introduire log(acide):grade et taille:grade.

```
> glm18.1_step(glm18,~.^2,direction="forward")
Start:  AIC= 58.5599
 Y ~ age + log(acide) + X + taille + grade

Single term additions

Model:
```

```
Y ~ age + log(acide) + X + taille + grade

scale:  1

                  Df Sum of Sq      RSS       Cp
           <none>              45.80816 57.80816
   age:log(acide)  1  0.516182 45.29198 59.29198
            age:X  1  1.038663 44.76950 58.76950
       age:taille  1  0.074456 45.73370 59.73370
        age:grade  1  0.633770 45.17439 59.17439
     log(acide):X  1  0.799075 45.00908 59.00908
log(acide):taille  1  0.019945 45.78821 59.78821
 log(acide):grade  1  3.425803 42.38236 56.38236
         X:taille  1  0.251874 45.55628 59.55628
          X:grade  1  0.045586 45.76257 59.76257
     taille:grade  1  8.010737 37.79742 51.79742

Step:  AIC= 52.4309
 Y ~ age + log(acide) + X + taille + grade + taille:grade

Single term additions

Model:
Y ~ age + log(acide) + X + taille + grade + taille:grade

scale:  1

                  Df Sum of Sq      RSS       Cp
           <none>              44.67801 58.67801
   age:log(acide)  1  0.983027 43.69498 59.69498
            age:X  1  0.535120 44.14289 60.14289
       age:taille  1  0.110387 44.56762 60.56762
        age:grade  1  0.835681 43.84232 59.84232
     log(acide):X  1  1.019920 43.65809 59.65809
log(acide):taille  1  1.191890 43.48612 59.48612
 log(acide):grade  1  2.205279 42.47273 58.47273
         X:taille  1  0.124882 44.55312 60.55312
          X:grade  1  0.004938 44.67307 60.67307

Step:  AIC= 51.3376
 Y ~ age + log(acide) + X + taille + grade + taille:grade + log(acide):grade

Single term additions

Model:
Y ~ age + log(acide) + X + taille + grade + taille:grade + log(acide):grade

scale:  1

                  Df Sum of Sq      RSS       Cp
           <none>              38.74654 54.74654
   age:log(acide)  1  0.631386 38.11515 56.11515
            age:X  1  0.183428 38.56311 56.56311
       age:taille  1  0.067424 38.67911 56.67911
        age:grade  1  0.843397 37.90314 55.90314
     log(acide):X  1  0.769580 37.97696 55.97696
```

```
log(acide):taille  1  2.351827 36.39471 54.39471
         X:taille  1  0.245158 38.50138 56.50138
          X:grade  1  0.000634 38.74590 56.74590

Step:  AIC= 51.0356
 Y ~ age + log(acide) + X + taille + grade + taille:grade +
     log(acide):grade + log(acide):taille

Single term additions

Model:
Y ~ age + log(acide) + X + taille + grade + taille:grade + log(acide):grade +
    log(acide):taille

scale:  1

               Df Sum of Sq      RSS       Cp
        <none>              38.20219 56.20219
age:log(acide)  1 0.5888726 37.61332 57.61332
         age:X  1 0.1173725 38.08482 58.08482
    age:taille  1 0.4245754 37.77762 57.77762
     age:grade  1 0.6677960 37.53440 57.53440
  log(acide):X  1 0.2601022 37.94209 57.94209
      X:taille  1 0.1395746 38.06262 58.06262
       X:grade  1 0.0223189 38.17987 58.17987
```

Chacun de ces termes apporte une explication significative.

```
> anova(glm18,update(glm18,~.+taille:grade),
+             update(glm18,~.+taille:grade+log(acide):grade),test="Chi")
Analysis of Deviance Table
Response: Y
                                                                    Terms
1                                   age + log(acide) + X + taille + grade
2                    age + log(acide) + X + taille + grade + taille:grade
3 age + log(acide) + X + taille + grade + taille:grade + log(acide):grade
  Resid. Df Resid. Dev              Test Df Deviance    Pr(Chi)
1        47   46.55990
2        46   38.43095     +taille:grade  1 8.128951 0.00435639
3        45   35.33757 +log(acide):grade  1 3.093379 0.07861140

> anova(glm18,update(glm18,~.+log(acide):grade),
+             update(glm18,~.+log(acide):grade+taille:grade),test="Chi")
Analysis of Deviance Table
Response: Y
                                                                    Terms
1                                   age + log(acide) + X + taille + grade
2                age + log(acide) + X + taille + grade + log(acide):grade
3 age + log(acide) + X + taille + grade + log(acide):grade + taille:grade
  Resid. Df Resid. Dev              Test Df Deviance    Pr(Chi)
1        47   46.55990
2        46   42.51803 +log(acide):grade  1 4.041868 0.04438467
3        45   35.33757     +taille:grade  1 7.180463 0.00737017
```

Faut-il ajouter d'autres interactions d'ordre 2? d'ordre 3? non.

```
> glm20_glm(Y~age+log(acide)+X+taille+grade+taille:grade+log(acide):grade,
```

```
+           binomial,prostate)

> anova(glm20,update(glm18,~.^2),test="Chi")[,-1]
Analysis of Deviance Table
Response: Y
  Resid. Df Resid. Dev
1        45   35.33757
2        37   30.17233
                                                                           Test
1
2 +age:log(acide)+age:X+age:taille+age:grade+log(acide):X+log(acide):taille+
                                                              X:taille+X:grade
  Df Deviance   Pr(Chi)
1
2  8 5.165236 0.7397783

> anova(glm20,update(glm18,~.^3),test="Chi")[,-1]
Analysis of Deviance Table
Response: Y
  Resid. Df Resid. Dev
1        45   35.33757
2        27   18.04626
                                                                          Test
1
2 +age:log(acide)+age:X+age:taille+age:grade+log(acide):X+log(acide):taille+
  X:taille+X:grade+age:log(acide):X+age:log(acide):taille+age:log(acide):grade+
  age:X:taille+age:X:grade+age:taille:grade+log(acide):X:taille+
  log(acide):X:grade+log(acide):taille:grade+X:taille:grade
  Df Deviance   Pr(Chi)
1
2 18 17.29131 0.5031699
There were 14 warnings (use warnings() to see them)

> anova(glm20,update(glm18,~.^2+grade:taille:log(acide)),test="Chi")[,-1]
Analysis of Deviance Table
Response: Y
  Resid. Df Resid. Dev
1        45   35.33757
2        36   28.50358
                                                                          Test
1
2 +age:log(acide)+age:X+age:taille+age:grade+log(acide):X+log(acide):taille+X:taille+
                                                     X:grade+grade:taille:log(acide)
  Df Deviance   Pr(Chi)
1
2  9 6.833985 0.6543991
There were 10 warnings (use warnings() to see them)
```

Est-il possible de supprimer des effets principaux qui n'interviennent dans aucun terme d'interaction ? oui, on peut enlever la variable âge.

```
> anova(glm20,update(glm20,~.-X),test="Chi")
Analysis of Deviance Table
Response: Y
  Resid. Df Resid. Dev Test Df Deviance    Pr(Chi)
1        45   35.33757
2        46   41.10404   -X -1 -5.76647 0.01633482
```

```
> anova(glm20,update(glm20,~.-age),test="Chi")
Analysis of Deviance Table
Response: Y
  Resid. Df Resid. Dev Test Df   Deviance   Pr(Chi)
1        45   35.33757
2        46   36.28715 -age -1 -0.9495802 0.3298262
```

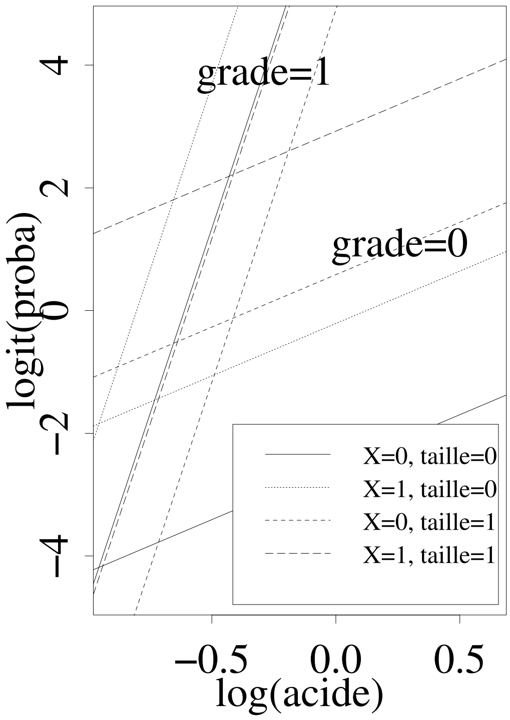
On aboutit au modèle final. L'effet principal taille est non significatif, si l'on se reporte à l'analyse de type 3 obtenue avec SAS. Mais il intervient dans un terme d'interaction, on le conserve donc dans le modèle. Avec l'analyse de type 1 affichée ci-dessous, c'est l'effet grade qui est non significatif, alors que l'effet taille est significatif. Certaines des corrélations entre les paramètres sont donc non négligeables. Ceci est lié au problème général de non orthogonalité en GLM. Conclusion pratique : pour interpréter, utiliser l'analyse de type 3, ou bien faire des tests de sous-modèle.

```
> options(contrasts=c("contr.treatment","contr.poly"))
> glm21_glm(Y~log(acide)+X+taille+grade+taille:grade+log(acide):grade,
+            binomial,prostate)

> anova(glm21,test="Chi")
Analysis of Deviance Table
Binomial model
Response: Y

Terms added sequentially (first to last)
                 Df Deviance Resid. Df Resid. Dev   Pr(Chi)
            NULL                   52   70.25215
      log(acide)  1 5.439431        51   64.81272 0.0196871
               X  1 9.541128        50   55.27159 0.0020092
          taille  1 6.285581        49   48.98601 0.0121724
           grade  1 1.210329        48   47.77568 0.2712669
    taille:grade  1 7.321379        47   40.45430 0.0068139
log(acide):grade  1 4.167158        46   36.28715 0.0412149

> summary(glm21)
Call: glm(formula = Y ~ log(acide) + X + taille + grade + taille:grade +
        log(acide): grade, family = binomial, data = prostate)
Deviance Residuals:
       Min         1Q     Median        3Q      Max
 -1.903177 -0.5651972 -0.2170102 0.1993139 2.427247

Coefficients:
                      Value Std. Error   t value
     (Intercept)  -2.552712   1.039471 -2.455779
      log(acide)   1.708528   1.420283  1.202949
               X   2.340447   1.084180  2.158727
          taille   3.137952   1.170442  2.680999
           grade   9.961154   4.663554  2.135958
    taille:grade  -5.647738   2.433498 -2.320831
log(acide):grade  10.425979   6.635917  1.571144

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 70.25215 on 52 degrees of freedom

Residual Deviance: 36.28715 on 46 degrees of freedom
```

```
Number of Fisher Scoring Iterations: 6

Correlation of Coefficients:
                 (Intercept) log(acide)          X     taille      grade
      log(acide)  0.2725927
               X -0.5053036   0.1116480
          taille -0.7036441   0.2284770  0.3977329
           grade -0.3048436  -0.0426516  0.2748103  0.2213421
    taille:grade  0.4428530  -0.1329628 -0.3979473 -0.5631621 -0.7486291
log(acide):grade -0.1480102  -0.1942174  0.1535562  0.0216778  0.9248190
                 taille:grade
      log(acide)
               X
          taille
           grade
    taille:grade
log(acide):grade -0.5054655
```

On représente graphiquement le modèle obtenu.

```
> limx_log(range(prostate$acide))
> limy_binomial()$link(c(0.01,0.99))
> b_coef(glm21)
> int_b["(Intercept)"]
> acid_b["log(acide)"]
> X_b["X"]
> taille_b["taille"]
> gr_b["grade"]
> taille.gr_b["taille:grade"]
> acid.gr_b["log(acide):grade"]
```



3.14 Predicting a binary response probability

L'exemple est l'exemple prostate (p. 132). On calcule la probabilité et un intervalle de confiance associés à un individu dont les caractéristiques sont X=0, taille=1, grade=1, acide=0.7.

```
> vcov_function(glm)
+   {result_summary(glm)
+   result$dispersion*result$cov.unscaled}

> new_data.frame(X=0,taille=1,grade=1,acide=0.7)

> predict(glm21,new, type="link")
        1
 0.5705819

> predict(glm21,new, type="response")
        1
 0.6388974

> vcov(glm21)
                 (Intercept) log(acide)          X     taille       grade
     (Intercept)   1.0805007  0.4024404 -0.5694638 -0.8560819  -1.4777695
      log(acide)   0.4024404  2.0172029  0.1719203  0.3798105  -0.2825055
               X  -0.5694638  0.1719203  1.1754453  0.5047107   1.3894768
          taille  -0.8560819  0.3798105  0.5047107  1.3699334   1.2081777
           grade  -1.4777695 -0.2825055  1.3894768  1.2081777  21.7487368
    taille:grade   1.1202194 -0.4595532 -1.0499237 -1.6040361  -8.4960034
log(acide):grade  -1.0209512 -1.8304754  1.1047643  0.1683702  28.6203358
                 taille:grade log(acide):grade
     (Intercept)    1.1202194       -1.0209512
      log(acide)   -0.4595532       -1.8304754
               X   -1.0499237        1.1047643
          taille   -1.6040361        0.1683702
           grade   -8.4960034       28.6203358
    taille:grade    5.9219117       -8.1625036
log(acide):grade   -8.1625036       44.0353936
```

Soient X une matrice d'incidence, θ un vecteur de paramètres. $X\theta$ est un vecteur de combinaisons linéaires des paramètres. Une valeur prédite, dans l'échelle du lien, est une combinaison linéaire des paramètres. Elle peut donc s'écrire sous la forme $X\theta$ où X est une matrice à une ligne contenant les coefficients de la combinaison linéaire à étudier. Le calcul de la variance associée à une valeur prédite (dans l'échelle du lien) se ramène donc au calcul de la variance de $X\theta$.
Or on a $Var(X\theta) = X\ Var(\theta)\ X'$ (Var se lit “matrice de variance covariance de ...”). On obtient une matrice de variance covariance. On en extrait la diagonale, et on en prend la racine carrée pour obtenir l'écart-type recherché. On utilise le $1-\alpha$ quantile d'une loi normale pour calculer un intervalle de confiance dans l'échelle du lien. On en déduit l'intervalle de confiance dans l'échelle initiale en appliquant la réciproque de la fonction de lien aux bornes de l'intervalle de confiance.

```
> new1_matrix(c(1,log(0.7),0,1,1,1,log(0.7)),nrow=1)
> et1_sqrt(diag(new1 %*% vcov(glm21) %*% t(new1)))
> b_predict(glm21,new1, type="link")
> IC1_c(b-qnorm(.975)*et1,b,b+qnorm(.975)*et1)
> binomial()$inverse(IC1)
[1] 0.1604868 0.6388974 0.9424471
```

L'intervalle de confiance obtenu est un peu différent de celui qui est donné par Collett. Cela vient de la matrice de variance covariance. L'intervalle de confiance est trop large pour que l'on puisse classer l'individu dans un groupe ou dans l'autre.

On calcule l'intervalle de confiance pour un nouvel individu. L'intervalle de confiance est plus étroit, on peut classer ce nouvel individu.

```
> new2_matrix(c(1,log(0.7),1,1,1,1,log(0.7)),nrow=1)
> et2_sqrt(diag(new2 %*% vcov(glm21) %*% t(new2)))
> b_predict(glm21,new2, type="link")
> IC2_c(b-qnorm(.975)*et2,b,b+qnorm(.975)*et2)
> binomial()$inverse(IC2)
[1] 0.5171316 0.9483890 0.9968384
```

On voudrait déterminer une règle d'affectation d'un individu à l'un des deux groupes (avec ou sans nodule). La règle la plus simple consiste à affecter un individu au groupe "sans nodule" si la probabilité estimée est inférieure à 0.5, et au groupe "avec nodule" dans le cas contraire. La règle d'affectation consiste à choisir une valeur seuil (0.5 ici).
Un choix plus astucieux de la valeur seuil consiste à étudier les proportions de mauvais classement, de faux positifs et de faux négatifs en fonction du seuil, et à choisir le seuil en tenant compte des probabilités de conclure à un faux positif ou à un faux négatif, ainsi que des conséquences d'une décision erronée. Ici, il est important de minimiser la probabilité associée à un faux négatif.

```
> seuil_c(0.25,0.35,0.45,0.50,0.55,0.65)
> faux.pos_NULL
> faux.neg_NULL
> mal.classes_NULL
> for (i in(1:length(seuil)))
+   {
+   p.chapeau_binomial()$inverse(predict(glm21))
+   b_table(p.chapeau>seuil[i],prostate$Y)
+   faux.pos_c(faux.pos,b[2,1]/sum(b[2,]))      # proportion de faux positifs
+   faux.neg_c(faux.neg,b[1,2]/sum(b[1,]))      # proportion de faux negatifs
+   mal.classes_c(mal.classes,(b[1,2]+b[2,1])/sum(b))  # proportion de
}                                               # mauvais classement
> round(cbind(seuil,faux.pos,faux.neg,mal.classes),3)
     seuil faux.pos faux.neg mal.classes
[1,]  0.25    0.321    0.040       0.189
[2,]  0.35    0.261    0.100       0.170
[3,]  0.45    0.227    0.097       0.151
[4,]  0.50    0.211    0.147       0.170
[5,]  0.55    0.176    0.167       0.170
[6,]  0.65    0.167    0.244       0.226
```

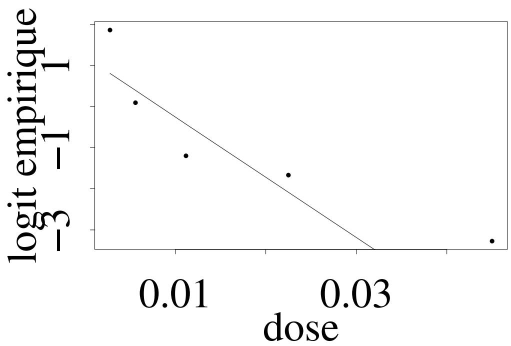
Chapitre 4

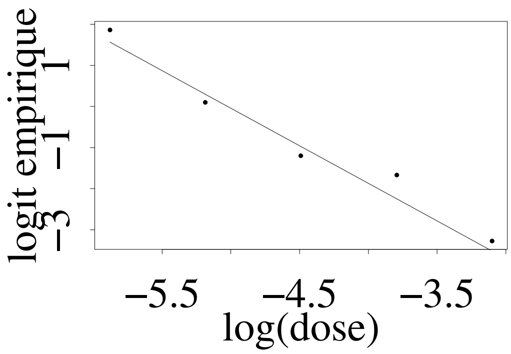
Bioassay and some other applications

4.1 The tolerance distribution

L'exemple utilisé est l'exemple sérum (p. 109). Les représentations graphiques des logit empiriques des proportions en fonction de la dose, ou de son logarithme montrent que l'hypothèse de linéarité est plus vraisemblable lorsqu'on choisit le log(dose) comme variable explicative.

```
> par(mfrow=c(2,2))
> attach(serum)
> plot(dose,log((y+0.5)/(n-y+0.5)),ylab="logit empirique")
> lines(dose,predict(glm(cbind(y,n-y)~dose,binomial)))
> plot(log(dose),log((y+0.5)/(n-y+0.5)),ylab="logit empirique")
> lines(log(dose),predict(glm(cbind(y,n-y)~log(dose),binomial)))
> detach()
```





On le confirme en ajustant chacun des deux modèles avec dose ou log(dose) comme régresseur. Avec log(dose), la déviance est de l'ordre de grandeur du degré de liberté résiduel, tandis qu'avec dose, la déviance est bien supérieure au degré de liberté. C'est donc log(dose) qu'il faut choisir.

```
> glm30_glm(cbind(y,n-y)~dose,binomial,serum)
> summary(glm30)
Call: glm(formula = cbind(y, n - y) ~ dose, family = binomial, data = serum)
Deviance Residuals:
        1          2         3        4        5
 2.727804 -0.9319343 -2.289251 0.754554 1.334382

Coefficients:
                  Value Std. Error   t value
(Intercept)    1.217898  0.2928912  4.158192
       dose -146.692648 26.3457216 -5.567987

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 90.05155 on 4 degrees of freedom

Residual Deviance: 15.90002 on 3 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
     (Intercept)
dose -0.7961891

> glm31_glm(cbind(y,n-y)~log(dose),binomial,serum)
> summary(glm31)
Call: glm(formula = cbind(y, n - y) ~ log(dose), family = binomial, data = serum)
Deviance Residuals:
         1          2          3        4          5
 0.8344325 -0.6209563 -0.7185586 1.090099 -0.1496342

Coefficients:
                Value Std. Error   t value
(Intercept) -9.189388  1.2543464 -7.326037
  log(dose) -1.829621  0.2544044 -7.191780

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 90.05155 on 4 degrees of freedom

Residual Deviance: 2.808898 on 3 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
          (Intercept)
log(dose) 0.9886115

> 1-pchisq(glm31$dev,glm31$df)
[1] 0.4220373

> 1-pchisq(glm30$dev,glm30$df)
[1] 0.001188781
```

On réalise de nouveau les deux ajustements, mais avec le lien probit. Les déviances obtenues sont légèrement supérieures aux déviances des modèles logit correspondant, mais la différence est très petite. On choisit un modèle logit.

```
> glm30.1_glm(cbind(y,n-y)~dose,binomial(link=probit),serum)
> summary(glm30.1)
Call: glm(formula = cbind(y, n - y) ~ dose, family = binomial(link = probit),
        data = serum)
Deviance Residuals:
        1          2         3         4        5
 3.041476 -0.7907873 -2.600867 0.0518275 1.564578

Coefficients:
                  Value Std. Error   t value
(Intercept)   0.6400596  0.1672923  3.825996
       dose -75.0692077 13.0584333 -5.748715

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 90.05155 on 4 degrees of freedom

Residual Deviance: 19.09102 on 3 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
     (Intercept)
dose -0.7823391

> glm31.1_glm(cbind(y,n-y)~log(dose),binomial(link=probit),serum)
> summary(glm31.1)
Call: glm(formula = cbind(y, n - y) ~ log(dose), family=binomial(link=probit),
        data = serum)
Deviance Residuals:
         1          2          3         4         5
 0.9214679 -0.6454021 -0.9907507 0.9663833 0.1093774

Coefficients:
                Value Std. Error   t value
(Intercept) -5.276683  0.6483279 -8.138911
  log(dose) -1.054481  0.1331760 -7.917952

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 90.05155 on 4 degrees of freedom

Residual Deviance: 3.193094 on 3 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
          (Intercept)
log(dose) 0.9859963
```

4.2 Estimating an effective dose

On souhaite estimer la DL_{50} ou DE_{50} (dose efficace ou dose léthale). Il s'agit de la dose qui provoque une réponse chez 50% des individus.
C'est encore l'exemple serum qui est utilisé (p. 109). Quelques nouvelles fonctions sont utilisées dans ce chapitre. Leur définition est donnée en annexe (p.195). Ces fonctions sont crées par la commande "source" qui suit.

```
> source("fcts.chap4")

> b_coef(glm31)
> DE50_exp(-b[1]/b[2])
> vcov(glm31)
            (Intercept) log(dose)
(Intercept)    1.573385 0.3154770
  log(dose)    0.315477 0.0647216
```

On calcule la DE_{50}, puis les DE_{50} et DE_{90} simultanément.

```
> dose.p(glm31)
              Dose        SE
p = 0.5: -5.022565 0.1052105

> dose.p(glm31,c(0.5,0.9))
              Dose        SE
p = 0.5: -5.022565 0.1052105
p = 0.9: -6.223483 0.2140911
```

On calcule deux intervalles de confiance pour la DE_{50}. Le premier est le résultat d'une approximation normale.

```
> IC.ED.approx(glm31)
             dose.inf        dose    dose.sup    p
p = 0.25: 0.009476657 0.012008918 0.015217824 0.25
p = 0.50: 0.005360095 0.006587609 0.008096235 0.50
p = 0.75: 0.002711414 0.003613697 0.004816234 0.75

> IC1.ED_IC.ED.approx(glm31,p=0.5)
> IC1.ED
            dose.inf        dose    dose.sup   p
p = 0.5: 0.005360095 0.006587609 0.008096235 0.5
```

Le deuxième est obtenu par la méthode du maximum de vraisemblance.

```
> IC2.ED_IC.ED.vrais(glm31,IC1.ED,p=0.5)
> IC2.ED
           dose.inf        dose    dose.sup   p
p = 0.5 0.005287773 0.006587609 0.008052915 0.5
```

4.3 Relative potency

L'exemple choisi est l'exemple "beetle" (Collett, exemple 4.3, p.103). Des lots d'environ 50 insectes sont exposés à trois types d'insecticides à différentes doses. Le premier insecticide est à base de DDT, le deuxième à base de BHC et le troisième insecticide est un mélange des deux premiers.

```
> y_c(3,5,19,19,24,35,2,14,20,27,41,40,28,37,46,48,48,50)
> n_c(50,49,47,50,49,50,50,49,50,50,50,50,50,50,50,50,50,50)
```

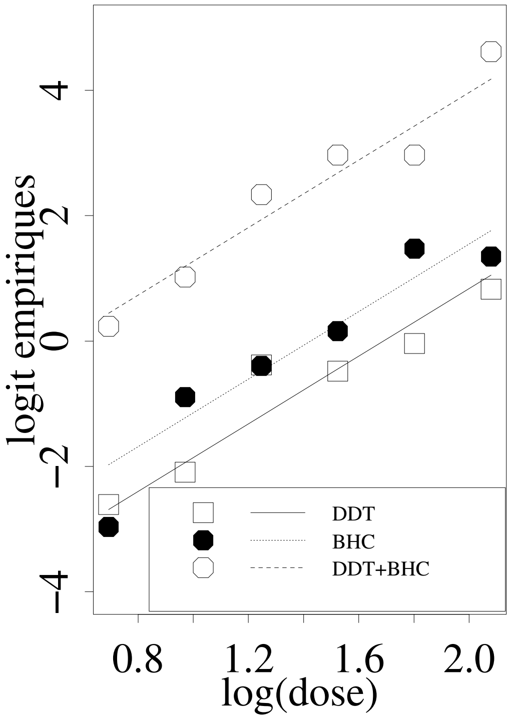
```
> dose_rep(c(2,2.64,3.48,4.59,6.06,8),3)
> insect_factor(rep(1:3,rep(6,3)))
> levels(insect)_c("DDT","BHC","DDT+BHC")

> beetles1_data.frame(insecticide=insect,dose=dose,y=y,n=n)
> rm(y,n,dose,insect)

> options(contrasts=c("contr.treatment","contr.treatment"))
> glm32_glm(cbind(y,n-y)~-1+insecticide+insecticide:log(dose),family=binomial,
                                                              data=beetles1)
> glm33_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=binomial,data=beetles1)
> glm34_glm(cbind(y,n-y)~log(dose),family=binomial,data=beetles1)

> summary(glm32)
Call: glm(formula = cbind(y, n - y) ~ -1 + insecticide + insecticide:log(dose),
        family = binomial, data = beetles1)
Deviance Residuals:
       Min        1Q      Median        3Q      Max
 -1.824537 -0.788704 -0.05548544 0.5942066 2.240953

Coefficients:
                                Value Std. Error   t value
             insecticideDDT -3.914985  0.5092150 -7.688275
             insecticideBHC -4.042812  0.4966308 -8.140478
         insecticideDDT+BHC -2.120668  0.5898309 -3.595382
    insecticideDDTlog(dose)  2.277869  0.3223113  7.067294
    insecticideBHClog(dose)  2.838064  0.3388185  8.376355
insecticideDDT+BHClog(dose)  3.388343  0.5719040  5.924671

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 429.2681 on 18 degrees of freedom

Residual Deviance: 17.8897 on 12 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
                            insecticideDDT insecticideBHC insecticideDDT+BHC
             insecticideBHC  0.0000000
         insecticideDDT+BHC  0.0000000      0.0000000
    insecticideDDTlog(dose) -0.9636210      0.0000000      0.0000000
    insecticideBHClog(dose)  0.0000000     -0.9607265      0.0000000
insecticideDDT+BHClog(dose)  0.0000000      0.0000000     -0.9503750
                            insecticideDDTlog(dose) insecticideBHClog(dose)
             insecticideBHC
         insecticideDDT+BHC
    insecticideDDTlog(dose)
    insecticideBHClog(dose)  0.0000000
insecticideDDT+BHClog(dose)  0.0000000               0.0000000

> summary(glm33)
Call: glm(formula = cbind(y, n - y) ~ -1 + insecticide + log(dose), family = binomial,
        data = beetles1)
Deviance Residuals:
       Min         1Q     Median        3Q      Max
```

```
 -2.026993 -0.6160165 -0.2422096 0.9016205 2.611961

Coefficients:
                       Value Std. Error    t value
    insecticideDDT -4.555254  0.3608836 -12.622501
    insecticideBHC -3.842464  0.3324548 -11.557853
insecticideDDT+BHC -1.424754  0.2849601  -4.999837
         log(dose)  2.695766  0.2155460  12.506687

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 429.2681 on 18 degrees of freedom

Residual Deviance: 21.28191 on 14 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
                   insecticideDDT insecticideBHC insecticideDDT+BHC
    insecticideBHC  0.8399421
insecticideDDT+BHC  0.7191820      0.7129846
         log(dose) -0.9204581     -0.9125262     -0.7813305

> summary(glm34)
Call: glm(formula = cbind(y, n - y) ~ log(dose), family = binomial, data = beetles1)
Deviance Residuals:
      Min        1Q    Median       3Q      Max
 -4.33605 -3.380553 -1.383809 4.023977 6.300754

Coefficients:
                Value Std. Error   t value
(Intercept) -2.465346  0.2386178 -10.33178
  log(dose)  2.010579  0.1702394  11.81030

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 413.6476 on 17 degrees of freedom

Residual Deviance: 246.8321 on 16 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
          (Intercept)
log(dose) -0.9501774
```

L'augmentation de déviance lorsqu'on impose l'égalité des pentes n'est pas significative. Par contre, l'augmentation de la déviance est significative lorsqu'on impose l'égalité des ordonnées à l'origine. Les droites (dans l'échelle du lien, ce sont des droites) sont donc parallèles, mais avec des ordonnées à l'origine différentes. On utilise le modèle avec des droites parallèles pour comparer l'efficacité des trois insecticides.

```
> anova(glm32,glm33,glm34,test="Chi")
Analysis of Deviance Table
 Response: cbind(y, n - y)
                                      Terms Resid. Df Resid. Dev         Test
1 -1 + insecticide + insecticide:log(dose)        12    17.8897
```

```
2            -1 + insecticide + log(dose)        14    21.2819      1 vs. 2
3                             log(dose)        16   246.8321 -insecticide
  Df  Deviance   Pr(Chi)
1
2 -2   -3.3922 0.1833962
3 -2 -225.5502 0.0000000

> summary(glm33)
Call: glm(formula = cbind(y, n - y) ~ -1 + insecticide + log(dose), family = binomial,
        data = beetles1)
Deviance Residuals:
       Min         1Q     Median        3Q      Max
 -2.026993 -0.6160165 -0.2422096 0.9016205 2.611961

Coefficients:
                      Value Std. Error    t value
    insecticideDDT -4.555254  0.3608836 -12.622501
    insecticideBHC -3.842464  0.3324548 -11.557853
insecticideDDT+BHC -1.424754  0.2849601  -4.999837
         log(dose)  2.695766  0.2155460  12.506687

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 429.2681 on 18 degrees of freedom

Residual Deviance: 21.28191 on 14 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
                   insecticideDDT insecticideBHC insecticideDDT+BHC
    insecticideBHC  0.8399421
insecticideDDT+BHC  0.7191820      0.7129846
         log(dose) -0.9204581     -0.9125262     -0.7813305

> attach(beetles1)
> beetles1$logit.emp_log((y+0.5)/(n-y+0.5))
> detach()
> attach(beetles1)
> plot(log(dose),binomial()$link(y/n),type="n",ylim=c(-4,5),ylab="logit empiriques")
> points(log(dose)[insecticide=="DDT"],logit.emp[insecticide=="DDT"],pch=0)
> points(log(dose)[insecticide=="BHC"],logit.emp[insecticide=="BHC"],pch=16)
> points(log(dose)[insecticide=="DDT+BHC"],logit.emp[insecticide=="DDT+BHC"],pch=1)
> lines(log(dose)[insecticide=="DDT"],predict(glm33)[insecticide=="DDT"],lty=1)
> lines(log(dose)[insecticide=="BHC"],predict(glm33)[insecticide=="BHC"],lty=2)
> lines(log(dose)[insecticide=="DDT+BHC"],predict(glm33)[insecticide=="DDT+BHC"],lty=3)
> legend(locator(2),lty=1:3,marks=c(0,16,1),legend=c("DDT","BHC","DDT+BHC"))
> detach()
```



On calcule la DE_{50} pour chacun des insecticides.

```
> b_coef(glm33)
> DE50_exp(c(-b[1]/b[4],-b[2]/b[4],-b[3]/b[4]))
> DE50
 insecticideDDT insecticideBHC insecticideDDT+BHC
       5.418292       4.159396           1.696412
```

L'efficacité relative du mélange DDT+BHC, par rapport au DDT seul peut être estimée en calculant le rapport des DE_{50}. Le mélange est trois fois plus efficace que le DDT seul, et 2.5 fois plus efficace que le BHC seul.

4.4 Natural response

C'est l'exemple "beetle" qui est repris (p. 148). Dans ce jeu de données, il existe aussi un groupe contrôle qui n'a pas été exposé aux insecticides. Il permet d'estimer le taux de mortalité à 10% (on ne retrouve pas exactement les mêmes résultats que Collett, mais il y avait une erreur dans le tableau de données. On peut supposer que les tableaux de données utilisés ne sont pas strictement les mêmes).

"logit.my" est une fonction dont la définition est donnée en annexe.
On corrige les données.

```
beetles1$n[4]_50

> glm35_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my,
                                            data=beetles1,trace=T)
> summary(glm35)
Call: glm(formula = cbind(y, n - y) ~ -1 + insecticide + log(dose),
                     family = logit.my, data = beetles1, trace = T)
Deviance Residuals:
       Min         1Q     Median       3Q      Max
 -2.765817 -0.7483743 -0.2879456 0.682271 2.604959
```

```
Coefficients:
                      Value Std. Error   t value
    insecticideDDT -5.634955  0.7028792 -8.016961
    insecticideBHC -4.725055  0.6301888 -7.497841
insecticideDDT+BHC -1.997181  0.4671632 -4.275125
         log(dose)  3.113860  0.3865635  8.055236

(Dispersion Parameter for binomiale - logit - reponse naturelle family
                    taken to be 1.734437 )

    Null Deviance: 414.5739 on 18 degrees of freedom

Residual Deviance: 26.38924 on 14 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
                   insecticideDDT insecticideBHC insecticideDDT+BHC
    insecticideBHC  0.8859135
insecticideDDT+BHC  0.7975988      0.7954024
         log(dose) -0.9425284     -0.9399329     -0.8462332
```

La fonction de lien utilisée est une extension du lien logit, prenant en compte la mortalité naturelle supposée connue et égale à 1% (p. 196).
Les DE_{50} obtenues en tenant compte de la mortalité naturelle sont supérieures à celles que l'on obtient lorsqu'on n'en tient pas compte.

```
> b_coef(glm35)

> DE50.my_exp(c(-b[1]/b[4],-b[2]/b[4],-b[3]/b[4]))

> DE50.my[1]/DE50.my[3]
 insecticideDDT
       3.216366

> DE50.my[2]/DE50.my[3]
 insecticideBHC
       2.401378
```

On ajuste le même modèle avec différentes valeurs de la probabilité naturelle : 0, 0.02, 0.04, 0.06, 0.08, 0.1. La déviance la plus faible est celle obtenue avec la probabilité naturelle égale à 0. Cela signifie que le jeu de données ne contient pas suffisamment d'information pour justifier la prise en compte d'une mortalité naturelle non nulle.

```
> glm35.1_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my(0),
                                                        data=beetles1)
> glm35.2_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my(0.02),
                                                           data=beetles1)
> glm35.3_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my(0.04),
                                                           data=beetles1)
> glm35.4_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my(0.06),
                                                           data=beetles1)
> glm35.5_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my(0.08),
                                                           data=beetles1)
> glm35.6_glm(cbind(y,n-y)~-1+insecticide+log(dose),family=logit.my(0.1),
                                                          data=beetles1)
```

```
> c(glm35.1$dev,glm35.2$dev,glm35.3$dev,glm35.4$dev,glm35.5$dev,glm35.6$dev)
[1] 21.28191 21.75397 22.49161 23.50327 24.79843 26.38924
```

4.5 Non-linear logistic regression models

Modèles de régression logistique polynomiaux

L'exemple est un autre exemple "beetle" que l'on appellera "CS2" (Collett, exemple 4.5, p.109) . Des lots d'environ 30 insectes sont esposés à différentes doses de CS^2. On observe le nombre d'insectes tués. L'expérience est répétée deux fois.

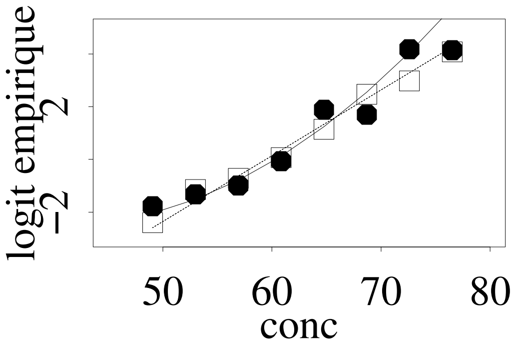
```
> CS2_data.frame(
  conc=rep(c(49.06,52.99,56.91,60.84,64.76,68.69,72.61,76.54),2),
  y=c(2,7,9,14,23,29,29,29,4,6,9,14,29,24,32,31),
  n=c(29,30,28,27,30,31,30,29,30,30,34,29,33,28,32,31),
  rep=rep(1:2,c(8,8))
  )
```

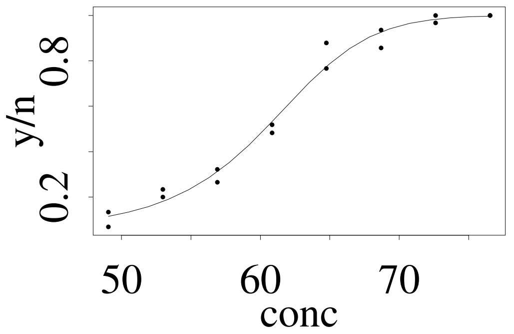
L'ajout d'un terme de degré 2 améliore significativement le modèle. L'ajout d'un terme de degré 3 au modèle de degré 2 n'apporte pas d'amélioration significative. La représentation graphique et le test de sous-modèle sont en faveur de l'introduction d'un terme de degré 2, alors que la déviance du modèle de degré 1 conduit à accepter l'hypothèse d'adéquation du modèle de degré 1.

```
> glm36_glm(cbind(y,n-y)~conc,binomial,CS2)
> glm37_glm(cbind(y,n-y)~conc+conc^2,binomial,CS2)
> glm38_glm(cbind(y,n-y)~conc+conc^2+conc^3,binomial,CS2)

> anova(glm36,glm37,glm38,test="Chi")
Analysis of Deviance Table
Response: cbind(y, n - y)
                   Terms Resid. Df Resid. Dev       Test Df Deviance   Pr(Chi)
1                   conc        14   12.50526
2          conc + conc^2        13    7.92733 +I(conc^2)  1 4.577934 0.0323862
3 conc + conc^2 + conc^3        12    7.88155 +I(conc^3)  1 0.045775 0.8305853

> summary(glm36)
Call: glm(formula = cbind(y, n - y) ~ conc, family = binomial, data = CS2)
Deviance Residuals:
       Min         1Q    Median        3Q      Max
 -1.129006 -0.4164031 0.2757867 0.9270647 1.534655

Coefficients:
                  Value Std. Error   t value
(Intercept) -14.8084253  1.2882400 -11.49508
       conc   0.2491701  0.0213588  11.66592

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 289.1413 on 15 degrees of freedom

Residual Deviance: 12.50526 on 14 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
```

```
     (Intercept)
conc -0.9948084

> summary(glm37)
Call: glm(formula = cbind(y, n - y) ~ conc + conc^2, family = binomial, data = CS2)
Deviance Residuals:
       Min         1Q    Median        3Q      Max
 -1.281047 -0.5417987 0.1436041 0.4310861 1.528116

Coefficients:
                   Value   Std. Error    t value
(Intercept)  7.968407076 11.034814021  0.7221152
       conc -0.516593243  0.373578176 -1.3828250
  I(conc^2)  0.006372126  0.003142401  2.0277891

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 289.1413 on 15 degrees of freedom

Residual Deviance: 7.927328 on 13 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
          (Intercept)       conc
     conc -0.9981309
I(conc^2)  0.9927660  -0.9982135
```

```
> par(mfrow=c(2,2))
> attach(CS2)
> CS2$logit.emp_log((y+0.5)/(n-y+0.5))
> detach()
> attach(CS2)
> plot(conc,binomial()$link(y/n),type="n",ylim=c(-3,5),ylab="logit empiriques")
> points(conc[rep==1],logit.emp[rep==1],pch=0)
> points(conc[rep==2],logit.emp[rep==2],pch=16)
> lines(conc,predict(glm36),lty=2)
> lines(sort(conc),sort(predict(glm37)),lty=1)

> plot(conc,y/n)
> b_seq(min(conc),max(conc),length=20)
> lines(b,predict(glm37,data.frame(conc=b),type="response"))
> detach()
```





On calcule la DE_{90}. On trouve une concentration égale à 67.69.

```
> b_coef(glm37)
> b[1]_b[1]-binomial()$link(0.9)

> round(polyroot(b),2)
 [1] 13.38+0i 67.69+0i
```

4.6 Applications of the complementary log-log model

4.6.1 A dilution assay

4.6.2 Analysis of grouped survival data

4.6.3 Serological testing

L'exemple utilisé est l'exemple “malaria” (Collett, exemple 4.6, p.116). On dispose de la proportion d'individus séropositifs par classe d'âge.

```
> malaria_data.frame(
> age=factor(1:7,labels=c("0-11mois","1-2ans","2-4ans","5-9ans",
                          "10-14ans","15-19ans",">20ans")),
  age.m=c(.5,1.5,3,7,12,17,30),
  y=c(3,1,5,39,31,8,91),
  n=c(10,10,29,69,51,15,108))

> glm38_glm(cbind(y,n-y)~offset(log(age.m)),binomial(link="cloglog"),malaria)

> summary(glm38)
Call: glm(formula = cbind(y, n - y) ~ offset(log(age.m)), family = binomial(link =
        "cloglog"), data = malaria)
Deviance Residuals:
```

```
      Min         1Q      Median       3Q      Max
 -1.512879 -0.9217521 -0.04939044 1.502618 2.865174

Coefficients:
                Value Std. Error   t value
(Intercept) -2.606822 0.08549524 -30.49084

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 20.25326 on 6 degrees of freedom

Residual Deviance: 20.25326 on 6 degrees of freedom

Number of Fisher Scoring Iterations: 4
```

On estime le taux d'infection annuel h par individu. On trouve 71 pour 1000 individus. On calcule l'écart-type associé, et on obtient un intervalle de confiance pour h.

```
> b_coef(glm38)
> h_1-exp(-exp(b))

> se.h_exp(b)*exp(-exp(b))*sqrt(vcov(glm38))
> c(h-qnorm(.975)*se.h,h,h+qnorm(.975)*se.h)
  (Intercept) (Intercept) (Intercept)
  0.05963121  0.07111339  0.08259557
```

Le modèle suppose un taux d'infection constant. Un écart à cette hypothèse conduit à un défaut d'adéquation du modèle. Le test d'adéquation, basé sur l'examen de la déviance revient donc à examiner dans quelle mesure les données sont compatibles avec l'hypothèse d'un taux d'infection constant. La déviance est significativement élevée. On rejette l'hypothèse.

```
> 1-pchisq(glm38$dev,glm38$df)
[1] 0.002496046
```

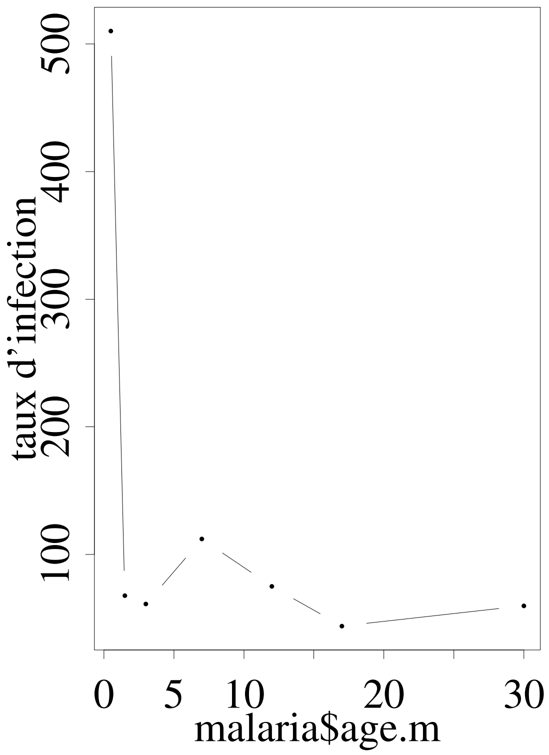
On envisage le modèle avec un taux d'infection par classe d'âge. Ce modèle est saturé.

```
> options(contrasts=c("contr.treatment","contr.treatment"))
> glm39_glm(cbind(y,n-y)~-1+offset(log(age.m))+age,binomial(link="cloglog"),malaria)

> summary(glm39)
Call: glm(formula = cbind(y, n - y) ~ -1 + offset(log(age.m)) + age, family =
        binomial(link = "cloglog"), data = malaria)

Coefficients:
                 Value Std. Error    t value
age0-11mois -0.3377833  0.5804155  -0.581968
  age1-2ans -2.6558324  1.0004623  -2.654605
  age2-4ans -2.7633410  0.4478812  -6.169808
  age5-9ans -2.1287409  0.1647971 -12.917346
age10-14ans -2.5509467  0.1862351 -13.697452
age15-19ans -3.1048383  0.3621726  -8.572815
  age>20ans -2.7865968  0.1204112 -23.142347

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 68.35809 on 7 degrees of freedom
```

```
Residual Deviance: 0 on 0 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
            age0-11mois age1-2ans age2-4ans age5-9ans age10-14ans age15-19ans
  age1-2ans 0
  age2-4ans 0           0
  age5-9ans 0           0         0
age10-14ans 0           0         0         0
age15-19ans 0           0         0         0         0
  age>20ans 0           0         0         0         0           0

> b_coef(glm39)
> h_1-exp(-exp(b))
> round(h*1000)
  age0-11mois age1-2ans age2-4ans age5-9ans age10-14ans age15-19ans age>20ans
        510        68        61       112          75          44        60
```

On représente les taux d'infection par classe d'âge.

```
> plot(malaria$age.m,h*1000,type="b",ylab="taux d'infection")
```



Chapitre 5

Model checking

5.1 Definition of residuals

L'exemple utilisé est l'exemple serum (p. 109). On calcule la fonction d'influence h (leverage), les résidus de Pearson, de déviance, de Pearson standardisés, et de déviance standardisés.

```
> glm40_glm(cbind(y,n-y)~log(dose),binomial,serum)
> attach(serum)
> y.chap_n*fitted(glm40)
> h_lm.influence(glm40)$h
> res.P_residuals(glm40,"pearson")
> res.D_residuals(glm40,"deviance")
> res.P.stand_res.P/sqrt((1-h))
> res.D.stand_res.D/sqrt((1-h))
> res.L.approx_sign(res.P)*sqrt(h*res.P.stand^2+(1-h)*res.D.stand^2)
> res.L_NULL
> for (i in 1:5)
+ res.L_c(res.L,sqrt(glm40$dev-glm(cbind(y,n-y)~log(dose),binomial,serum,
+                                                        subset=-i)$dev))
> res.L_sign(res.P)*res.L

> cbind(dose,y,y.chap,h,res.P,res.D,res.P.stand,res.D.stand,res.L.approx,res.L)
    dose  y    y.chap         h      res.P      res.D res.P.stand res.D.stand
1 0.0028 35 33.084904 0.5770961  0.8005511  0.8344325   1.2310288   1.2831292
2 0.0056 21 22.950061 0.4096647 -0.6234834 -0.6209563  -0.8114757  -0.8081866
3 0.0112  9 10.987071 0.3595737 -0.7037605 -0.7185586  -0.8794078  -0.8978992
4 0.0225  6  3.823068 0.3848125  1.1699336  1.0900992   1.4916175   1.3898319
5 0.0450  1  1.154904 0.2688530 -0.1460937 -0.1496342  -0.1708556  -0.1749962
  res.L.approx      res.L
1    1.2533265  1.2436609
2   -0.8095356 -0.8141408
3   -0.8912944 -0.8854118
4    1.4298582  1.4609623
5   -0.1738927 -0.1737757
> detach()
```

Exemple “CS2” (p. 154). On réalise un premier ajustement avec le lien canonique logit, avec un effet linéaire de la concentration. Les méthodes graphiques de validation montrent qu'il est nécessaire d'ajouter un terme quadratique.

```
> glm41_glm(cbind(y,n-y)~conc,binomial,CS2)
> attach(CS2)
> y.chap_n*fitted(glm41)
```

```
> h_lm.influence(glm41)$h
> res.P_residuals(glm41,"pearson")
> res.D_residuals(glm41,"deviance")
> res.P.stand_res.P/sqrt((1-h))
> res.D.stand_res.D/sqrt((1-h))
> res.L.approx_sign(res.P)*sqrt(h*res.P.stand^2+(1-h)*res.D.stand^2)
> res.L_NULL
> for (i in 1:nrow(CS2))
+ res.L_c(res.L,sqrt(glm41$dev-glm(cbind(y,n-y)~conc,binomial,CS2,subset=-i)$dev))
> res.L_sign(res.P)*res.L
> result_cbind(conc,y,y.chap,h,res.P,res.D,res.P.stand,res.D.stand,res.L.approx,res.L)
> result_result[order(result[,1]),]

> round(result,3)
    conc  y y.chap     h  res.P  res.D res.P.stand res.D.stand res.L.approx  res.L
 1 49.06  2  2.035 0.136 -0.025 -0.025      -0.027      -0.027       -0.027 -0.027
 9 49.06  4  2.105 0.141  1.353  1.216       1.459       1.311        1.333  1.336
 2 52.99  7  5.019 0.166  0.968  0.926       1.060       1.014        1.021  1.023
10 52.99  6  5.019 0.166  0.480  0.468       0.525       0.513        0.515  0.515
 3 56.91  9  9.742 0.137 -0.294 -0.296      -0.317      -0.319       -0.319 -0.319
11 56.91  9 11.830 0.166 -1.019 -1.042      -1.116      -1.141       -1.137 -1.135
 4 60.84 14 15.846 0.115 -0.721 -0.717      -0.767      -0.762       -0.762 -0.763
12 60.84 14 17.020 0.123 -1.139 -1.129      -1.216      -1.206       -1.207 -1.207
 5 64.76 23 23.714 0.137 -0.320 -0.316      -0.345      -0.340       -0.341 -0.341
13 64.76 29 26.086 0.151  1.246  1.328       1.352       1.441        1.428  1.426
 6 68.69 29 28.193 0.132  0.504  0.530       0.541       0.569        0.565  0.565
14 68.69 24 25.465 0.119 -0.964 -0.897      -1.027      -0.956       -0.965 -0.966
 7 72.61 29 28.916 0.094  0.082  0.083       0.086       0.087        0.087  0.087
15 72.61 32 30.844 0.100  1.093  1.535       1.152       1.618        1.577  1.577
 8 76.54 29 28.597 0.057  0.637  0.901       0.656       0.927        0.914  0.914
16 76.54 31 30.570 0.061  0.659  0.931       0.680       0.961        0.946  0.946
> detach()
```

5.2 Checking the form of the linear predictor

Exemple "lupin"

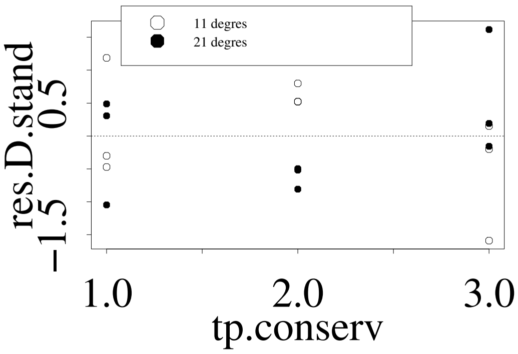
La probabilité de germination de graines de lupin est étudiée en fonction de la température de germination, du niveau d'humidité et de la température de stockage (Collett, exemple 5.3, p.127).

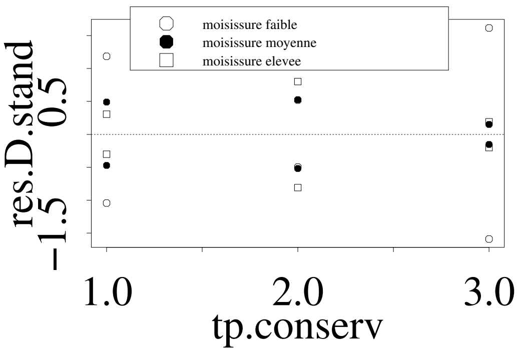
```
> lupin_data.frame(
+   y=c(98,96,62,94,79,,3,92,41,,1,94,93,65,94,71,,2,91,30,1),
+   tp.germ=factor(rep(c(11,21),c(9,9))),
+   moisissure=factor(rep(c(1:3,1:3),rep(3,6))),
+   tp.conserv=factor(rep(c(21,42,62),6)),
+   n=rep(100,18))

> glm42_glm(cbind(y,n-y)~tp.germ+moisissure+tp.conserv+tp.conserv:moisissure,
+   binomial,lupin)

> summary(glm42,corr=F)
Call: glm(formula = cbind(y, n - y) ~ tp.germ + moisissure + tp.conserv +
                  tp.conserv:moisissure, family = binomial, data = lupin)
Deviance Residuals:
       Min         1Q       Median        3Q       Max
 -0.986127 -0.3181911 -0.008164557 0.3350929 0.9674891
```

```
Coefficients:
                              Value Std. Error    t value
            (Intercept)  3.3250529  0.3703801  8.9774065
                tp.germ -0.2764638  0.1491711 -1.8533328
            moisissure2 -0.4268958  0.4678300 -0.9125020
            moisissure3 -0.8026318  0.4410270 -1.8199152
           tp.conserv42 -0.3344848  0.4758279 -0.7029532
           tp.conserv62 -2.6305106  0.3896588 -6.7508049
moisissure2tp.conserv42 -1.3220492  0.5847478 -2.2608878
moisissure3tp.conserv42 -2.6496109  0.5593322 -4.7370971
moisissure2tp.conserv62 -3.8020280  0.6677441 -5.6938395
moisissure3tp.conserv62 -4.3581292  0.8485202 -5.1361524

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 1230.86 on 17 degrees of freedom

Residual Deviance: 4.429755 on 8 degrees of freedom

Number of Fisher Scoring Iterations: 3
```

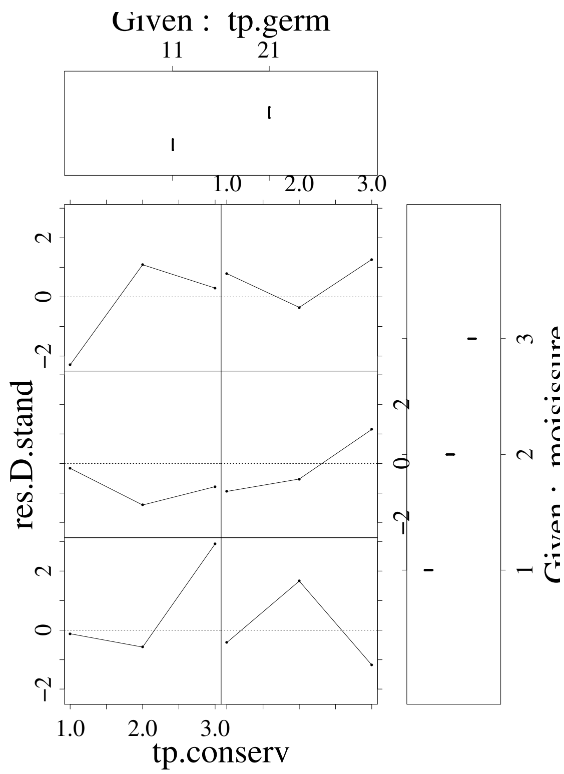
```
> res.D_residuals(glm42,"deviance")
> h_lm.influence(glm42)$h
> res.D.stand_res.D/sqrt((1-h))
```

```
> attach(lupin)
> x_as.numeric(tp.conserv)
> par(mfrow=c(2,2))
> plot(x,res.D.stand,type="n",xlab="tp.conserv")
> points(x[tp.germ==11],res.D.stand[tp.germ==11],pch=1)
> points(x[tp.germ==21],res.D.stand[tp.germ==21],pch=16)
> abline(h=0,lty=2)
> legend(locator(2),marks=c(1,16),legend=c("11 degres","21 degres"))
> plot(x,res.D.stand,type="n",xlab="tp.conserv")
> points(x[moisissure==1],res.D.stand[moisissure==1],pch=1)
> points(x[moisissure==2],res.D.stand[moisissure==2],pch=16)
> points(x[moisissure==3],res.D.stand[moisissure==3],pch=0)
> abline(h=0,lty=2)
> legend(locator(2),marks=c(1,16,0),
+        legend=c("moisissure faible","moisissure moyenne","moisissure elevee"))
> par(mfg=c(1,1,1,1))
> title(main="lupin")
```






```
>
> coplot(res.D.stand~tp.conserv|tp.germ:moisissure,
+        panel=function(x,y) {points(x,y);lines(x,y);abline(h=0,lty=2)})
```




```
> matrix(round(res.D.stand,1),ncol=3,byrow=T)
     [,1] [,2] [,3]
```

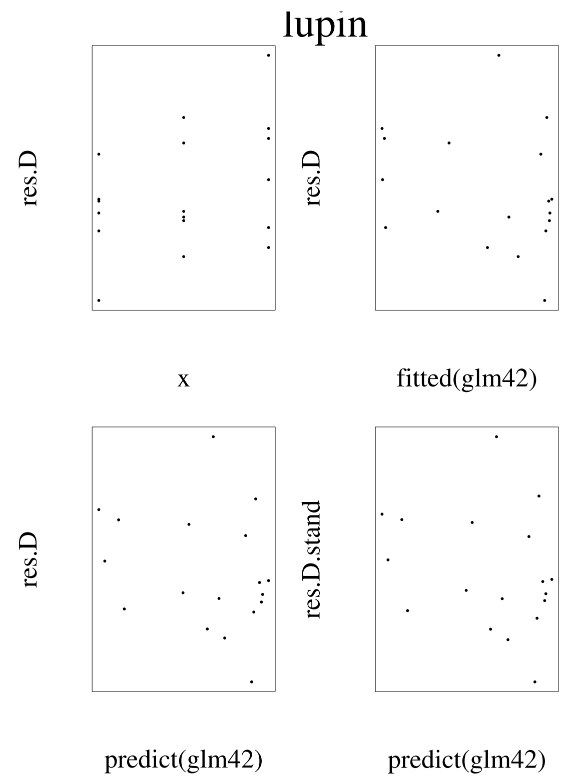
```
[1,]  1.2  0.5 -1.6
[2,] -0.5  0.5  0.2
[3,] -0.3  0.8 -0.2
[4,] -1.0 -0.5  1.6
[5,]  0.5 -0.5 -0.2
[6,]  0.3 -0.8  0.2
```

Les valeurs les plus mal ajustées sont celles correspondant à la température de conservation la plus élevée, 62°C, et le taux d'humidité le plus faible. Les résidus correspondant à une température de conservation égale à 42°C sont tous positifs lorsque la température de germination est de 11°C, et tous négatifs lorsque la température de germination est de 21°C . Cela suggère une interaction entre température de germination et température de conservation. Mais le modèle obtenu avec interaction n'est pas significativement meilleur.

```
> glm43_update(glm42,~.+tp.conserv:tp.germ)
> anova(glm42,glm43,test="Chi")[,-1]
Analysis of Deviance Table
Response: cbind(y, n - y)
  Resid. Df Resid. Dev                Test Df Deviance   Pr(Chi)
1         8   4.429755
2         6   1.892430 +tp.conserv:tp.germ  2 2.537325 0.2812075
```

On représente les résidus en fonctions des valeurs estimées (on peut encore les représenter en fonction du prédicteur ou en fonction des variables explicatives).

```
> par(mfrow=c(2,2))
> plot(x,res.D)
> plot(fitted(glm42),res.D)
> plot(predict(glm42),res.D)
> plot(predict(glm42),res.D.stand)
> par(mfg=c(1,1,1,1))
> mtext("lupin",cex=3)
```



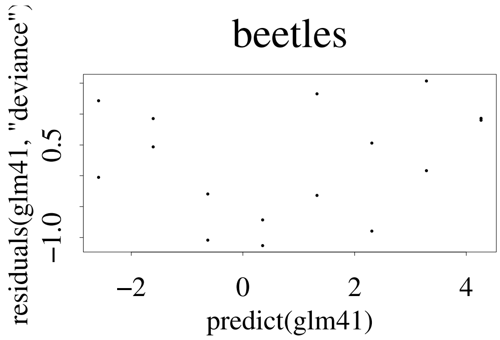
5.2.1 Plots of residuals

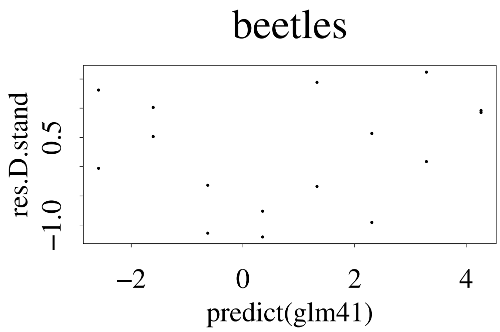
5.2.2 Half-normal plot of residuals

Exemple “CS2” (p. 154)

On représente les résidus de déviance standardisés en fonction du prédicteur. La représentation des résidus de déviance standardisés est très proche de celle des résidus de déviance non standardisés.

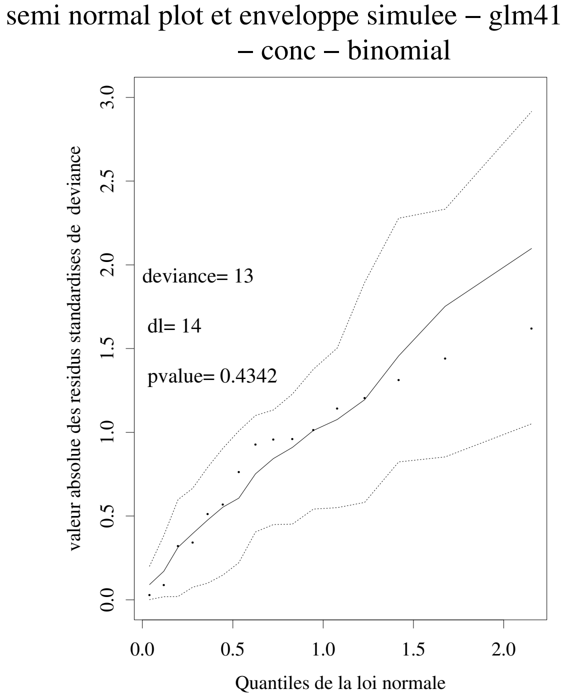
```
> res.D_residuals(glm41,"deviance")
> h_lm.influence(glm41)$h
> res.D.stand_res.D/sqrt(1-h)
> par(mfrow=c(2,2))
> plot(predict(glm41),residuals(glm41,"deviance"),main="beetles")
> plot(predict(glm41),res.D.stand,main="beetles")
```





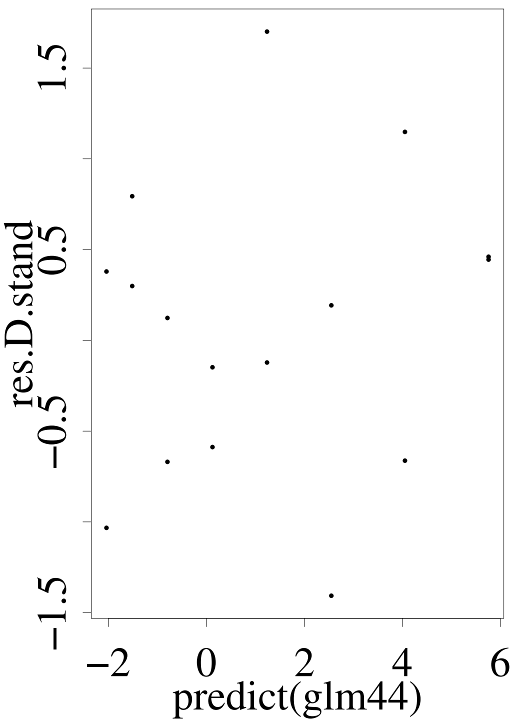
On représente aussi le graphique semi normal des résidus avec enveloppe simulée (half normal plot, Collett, p.129).

```
> source("fcts.chap5")

> par(mfrow=c(1,1))
> env41_enveloppe(glm41)
> enveloppe.plot(env41)
```

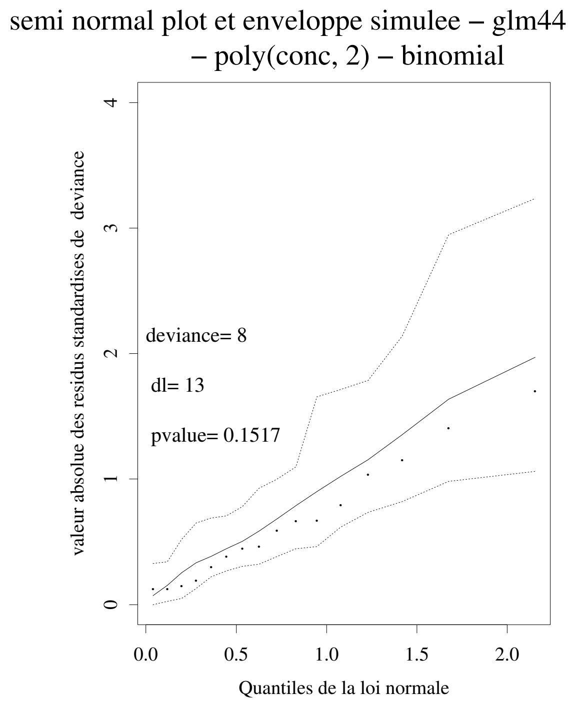


La représentation graphique des résidus en fonction du prédicteur suggère une structure quadratique des résidus, et donc l'ajout du régresseur conc^2. On ajuste un modèle avec effet quadratique. La structure des résidus est alors supprimée, mais l'enveloppe simulée suggère un modèle surparamétré, car tous les résidus, sauf un, sont inférieurs à la moyenne des résidus simulés.

```
> glm44_glm(cbind(y,n-y)~poly(conc,2),binomial,CS2)

> res.D_residuals(glm44,"deviance")
> h_lm.influence(glm44)$h
> res.D.stand_res.D/sqrt(1-h)
> par(mfrow=c(2,2))
> plot(predict(glm44),res.D.stand,main="beetles, effet quadratique")
```



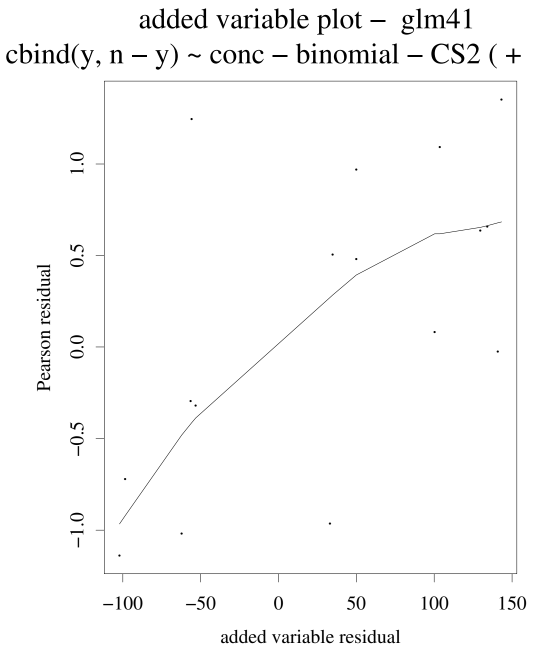

```
par(mfrow=c(1,1))
env44_enveloppe(glm44)
```



5.2.3 The added variable plot

La question posée est *Faut-il ajouter une nouvelle variable explicative ?* L'exemple support est l'exemple CS2 (p. 154). La question de l'ajout d'un terme quadratique se pose. Le graphique de la variable ajoutée (ou “added variable plot”) est un outil d'aide à la décision. Ce graphique suggère l'existence d'une structure des résidus de Pearson, explicable par la variable ajoutée conc^2. Il suggère donc l'ajout d'un terme quadratique.

```
> conc2_CS2$conc^2
> b_added.var.plot(glm41,conc2)
```

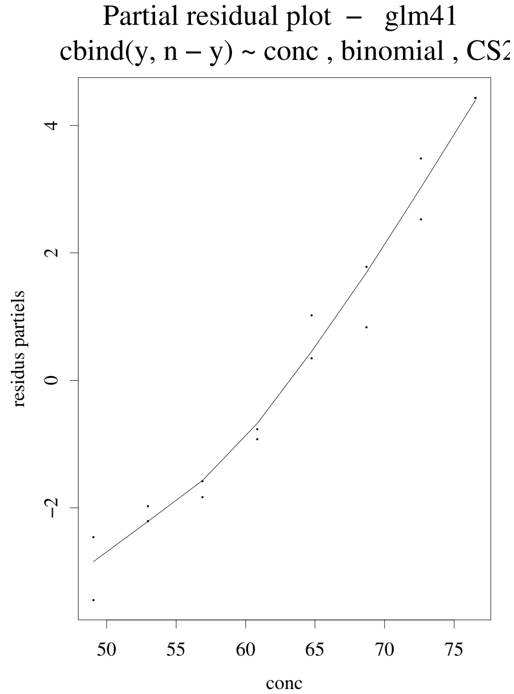



```
> anova(glm41,glm44,test="Chi")
Analysis of Deviance Table
Response: cbind(y, n - y)
          Terms Resid. Df Resid. Dev    Test Df Deviance    Pr(Chi)
1          conc        14   12.50526
2 poly(conc, 2)        13    7.92733 1 vs. 2  1 4.577934 0.03238624
```

5.2.4 The partial residual plot

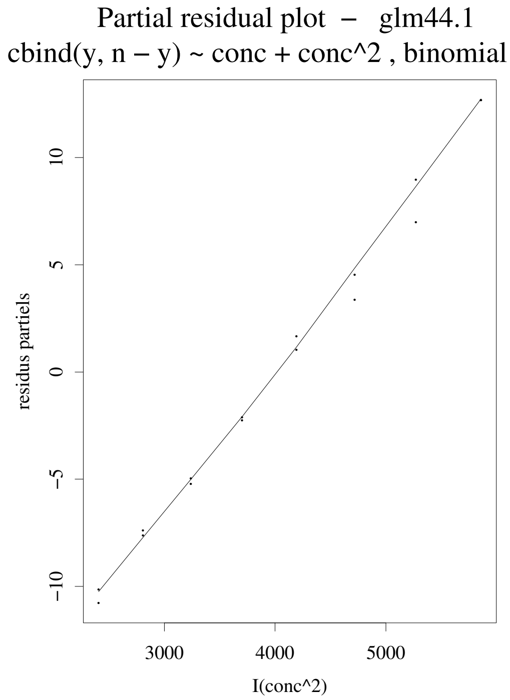
Faut-il transformer la variable explicative ? la courbure du graphique des résidus partiels obtenu avec le modèle à effet linéaire suggère l'ajout d'un terme quadratique.

```
> glm41$call
glm(formula = cbind(y, n - y) ~ conc, family = binomial, data = CS2)

> partial.res.plot(glm41,"conc")
```



L'absence de courbure, dans le graphique obtenu avec le modèle quadratique, suggère une prise en compte correcte de l'effet concentration.

```
> glm44.1_glm(cbind(y,n-y)~conc+conc^2,binomial,CS2)
> partial.res.plot(glm44.1,"I(conc^2)")
```

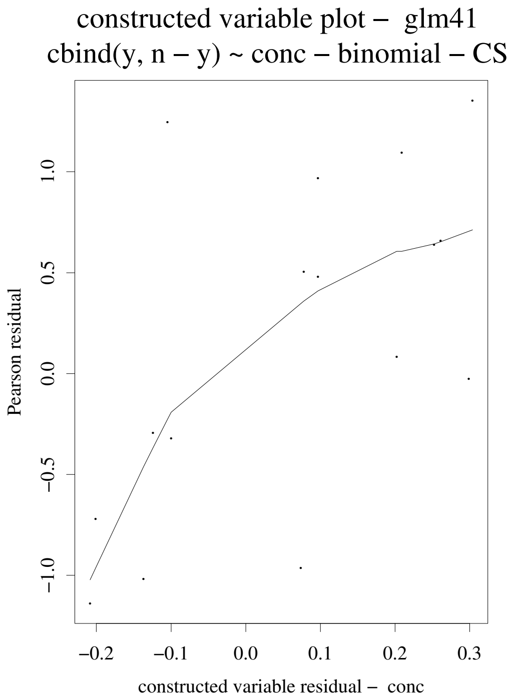


5.2.5 The constructed variable plot

La question de la nécessité de transformer la variable explicative, et du choix d'une tranformation se pose. La méthode graphique proposée s'appelle "constructed variable plot".
Il y a peut-être une erreur dans Collett p.137? "la pente estimée est égale à lambda - 1" serait à remplacer par "la pente estimée est égale à lambda + 1"? on trouverait alors les mêmes résultats.

Le graphique suggère l'introduction d'un terme quadratique en la concentration.

```
> attach(CS2)
> constr.var.plot(glm41,conc)
> detach()
 $pente:
        X
 2.755017

$lambda:
[1] "? pente - 1 ?"

$transfo:
[1] "(x^lambda-1)/lambda"
```



5.3 Checking the adequacy of the link function

Voyons si l'ajustement d'un modèle qui contient un terme de degré 1 seulement peut être amélioré par un changement de la fonction de lien. On ajoute donc au modèle logistique de degré 1, une variable contrsuite z. L'ajout de cette variable diminue significativement la déviance, ce qui laisse penser qu'une fonction de lien plus appropriée que le lien logit peut être trouvée. Le coefficient de z vaut 1.232, le paramètre estimé pour la fonction de lien générale vaut $\widehat{\alpha} = 1 - \widehat{\gamma} = -0.232$, il est proche de 0, ce qui suggère un lien cloglog.

```
> z_-(1+log(1-fitted(glm41))/fitted(glm41))
> glm41.1_update(glm41,~.+z)
> anova(glm41,glm41.1,test="Chi")

Response: cbind(y, n - y)

     Terms Resid. Df Resid. Dev Test Df Deviance    Pr(Chi)
1     conc        14   12.50526
2 conc + z        13    7.91456   +z  1 4.590701 0.03214587

> summary(glm41.1,corr=F)
Call: glm(formula = cbind(y, n - y) ~ conc + z, family = binomial, data = CS2)
Deviance Residuals:
       Min         1Q    Median        3Q     Max
 -1.321102 -0.5123128 0.1015754 0.4355183 1.56672

Coefficients:
                 Value Std. Error   t value
(Intercept) -8.8302896 3.00464245 -2.938882
       conc  0.1366023 0.05589114  2.444078
          z  1.2324223 0.61235241  2.012603

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 289.1413 on 15 degrees of freedom

Residual Deviance: 7.914561 on 13 degrees of freedom

Number of Fisher Scoring Iterations: 4

> 1-coef(glm41.1)["z"]
          z
 -0.2324223
```

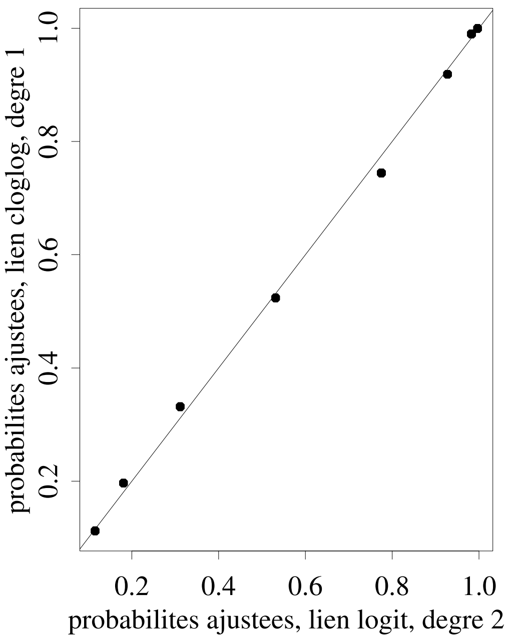
Effectivement, la déviance d'un modèle de degré 1 avec lien cloglog est très faible, elle vaut 8.67 pour 14 ddl. De plus, l'ajout d'un terme quadratique n'apporte pas d'explication significative.

```
> glm41.2_glm(cbind(y,n-y)~conc,family=binomial(link=cloglog),data=CS2)
> summary(glm41.2)

Call: glm(formula = cbind(y, n - y) ~ conc, family = binomial(link = cloglog), data
         = CS2)
Deviance Residuals:
       Min        1Q     Median        3Q      Max
 -1.092198 -0.527681 0.07698465 0.3512406 1.912148

Coefficients:
                 Value Std. Error   t value
(Intercept) -9.7550930 0.82404343 -11.83808
       conc  0.1554306 0.01301436  11.94301

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 289.1413 on 15 degrees of freedom
```

```
Residual Deviance: 8.671043 on 14 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
     (Intercept)
conc -0.9951996

> anova(glm41.2,update(glm41.2,~.+conc^2),test="Chi")
Response: cbind(y, n - y)

             Terms Resid. Df Resid. Dev       Test Df  Deviance   Pr(Chi)
1             conc        14   8.671043
2 conc + I(conc^2)        13   8.303091 +I(conc^2)  1 0.3679526 0.5441224
```

On représente graphiquement les probabilités ajustées avec le modèle logistique de degré 2, en fonction des probabilités ajustées avec le modèle à lien cloglog de degré 1. Ce graphique montre que les deux modèles sont très proches. Le modèle à lien cloglog a l'avantage d'être plus parcominieux.

```
> plot(fitted(glm44.1),fitted(glm41.2),xlab="probabilites ajustees, lien logit, degre 2",
+                                     ylab="probabilites ajustees, lien cloglog, degre 1")
> abline(c(0,1))
```



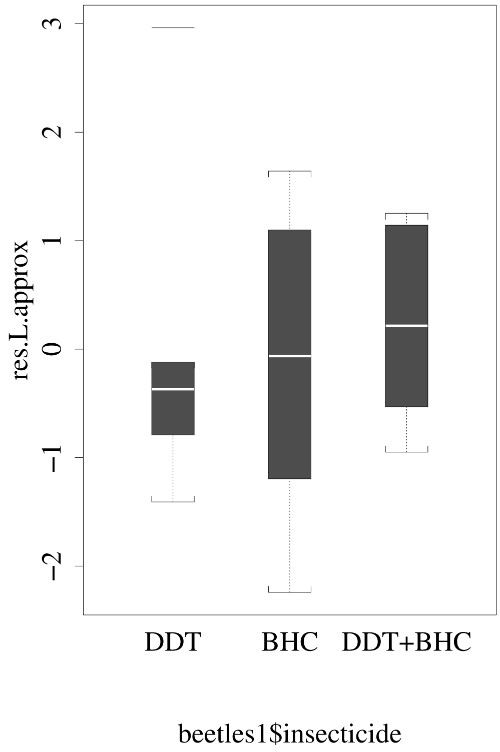
5.4 Identification of outlying observations

Exemple "beetle" (p. 148).

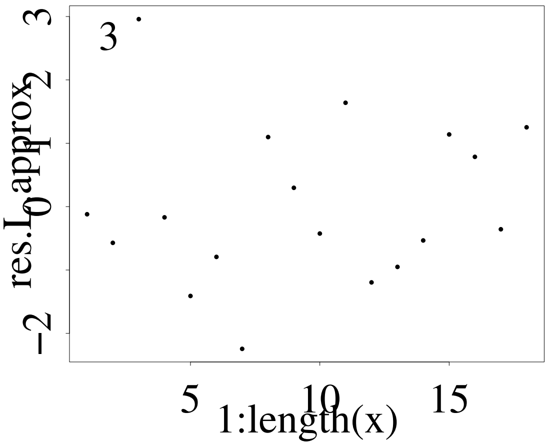
```
> h_lm.influence(glm33)$h
> res.P_residuals(glm33,"pearson")
```

```
> res.D_residuals(glm33,"deviance")
> res.P.stand_res.P/sqrt((1-h))
> res.D.stand_res.D/sqrt((1-h))
> res.L.approx_sign(res.P)*sqrt(h*res.P.stand^2+(1-h)*res.D.stand^2)

> round(cbind(res.P,res.D,res.P.stand,res.D.stand,res.L.approx),3)
    res.P  res.D res.P.stand res.D.stand res.L.approx
 1 -0.109 -0.110      -0.119      -0.120       -0.120
 2 -0.502 -0.517      -0.557      -0.574       -0.571
 3  2.784  2.612       3.114       2.922        2.961
 4 -0.145 -0.145      -0.166      -0.167       -0.166
 5 -1.205 -1.197      -1.416      -1.407       -1.409
 6 -0.659 -0.649      -0.800      -0.787       -0.791
 7 -1.770 -2.027      -2.012      -2.305       -2.242
 8  0.981  0.955       1.122       1.092        1.099
 9  0.261  0.260       0.298       0.297        0.297
10 -0.371 -0.370      -0.424      -0.423       -0.423
11  1.377  1.435       1.592       1.658        1.642
12 -1.072 -1.026      -1.235      -1.182       -1.195
13 -0.713 -0.708      -0.952      -0.946       -0.949
14 -0.454 -0.448      -0.537      -0.530       -0.532
15  0.979  1.041       1.085       1.154        1.142
16  0.692  0.741       0.741       0.794        0.787
17 -0.353 -0.339      -0.370      -0.355       -0.357
18  0.873  1.232       0.901       1.271        1.252

> plot(beetles1$insecticide,res.L.approx)
```

Pour l'insecticide DDT, tous les résidus de vraisemblance sont négatifs sauf 1.

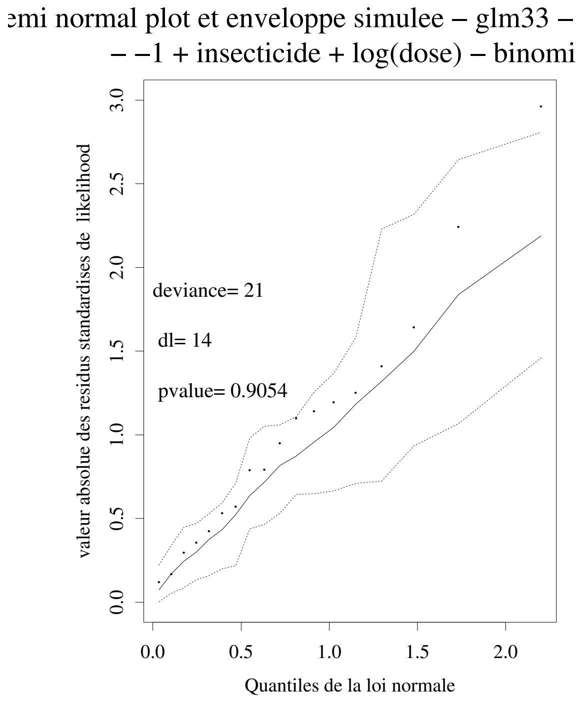



```
> index.plot(res.L.approx)
```



Le résidu correspondant à la troisième observation est particulièrement élevé.

```
> env33_enveloppe(glm33,type="likelihood",ylab="abs(res de vrais)")
```



Les enveloppes simulées confirment que ce résidu est particulièrement élevé.
Le résidu le plus grand (3^{e} observation) correspond-il à une observation discordante? on réalise un test du rapport de vraisemblance.

```
> glm33.1_update(glm33,subset=c(-3))
```

```
> anova(glm33,glm33.1,test="Chi")
Analysis of Deviance Table
Response: cbind(y, n - y)
                         Terms Resid. Df Resid. Dev Test Df Deviance
1 -1 + insecticide + log(dose)        14   21.28191
2 -1 + insecticide + log(dose)        13   12.44224    =   1 8.839671
      Pr(Chi)
1
2 0.002947522
```

Le résidu le plus grand correspond-il à une observation discordante ? on réalise un test approché.
$\alpha = 0.05$
Risque $\frac{\alpha}{2}$ pour un test bilatéral
Risque $\frac{\alpha}{2n}$ pour faire un ajustement du risque d'erreur, de type Bonferroni.
Les résidus supérieurs, en valeur absolue, à 2.99 sont déclarés discordants. Le 3e résidu correspond à une observation discordante (p=0.055).

```
> qnorm(1-0.05/(2*nrow(beetles1)))
[1] 2.991316
> 2*nrow(beetles1)*(1-pnorm(res.L.approx[3]))
         3
 0.05514435
```

Une observation est discordante. Que faire ? Soit le modèle est inadéquat, soit l'observation a une particularité.
Si c'est le modèle que l'on remet en cause, on peut l'améliorer en ajoutant d'autres variables explicatives. On peut aussi modifier certains régresseurs (dose ou log(dose) ...), autoriser des non-linéarités dans la liaison entre le logit et le régresseur (en remplaçant des effets linéaires par des effets quadratiques ..., en remplaçant un effet linéaire par un effet différent pour chaque niveau du facteur). On peut également remettre en cause le lien choisi.
Si la validité du modèle n'est pas remise en cause, s'il est considéré comme adéquat, c'est l'observation qui est suspecte. Dans cet exemple, le point extérieur detecté reflète le fait que le lot de 47 souris exposées à 3.48 $mg/10cm^2$ de DDT est plus sensible au poison que les autres lots.
Bien sûr, on peut rejeter la valeur extérieure et réajuster le modèle sur le nouveau jeu de données. Bien que cette démarche puisse être appropriée quand le modèle est considéré comme inviolable, cette procédure ne doit pas être adoptée comme une procédure de routine.
Si un ou des points extérieurs sont identifiés, l'analyse peut-être refaite en les excluant. Si les conclusions sont identiques, pas de problème. Si les conclusions sont différentes, la façon de considérer ces points extérieurs est cruciale. Ce ne sont pas des considérations statistiques qui détermineront l'attitude à adopter en présence de points extérieurs, ce sont des discussions avec les personnes qui ont recueilli les données qui permettront de décider du statut de ces données.

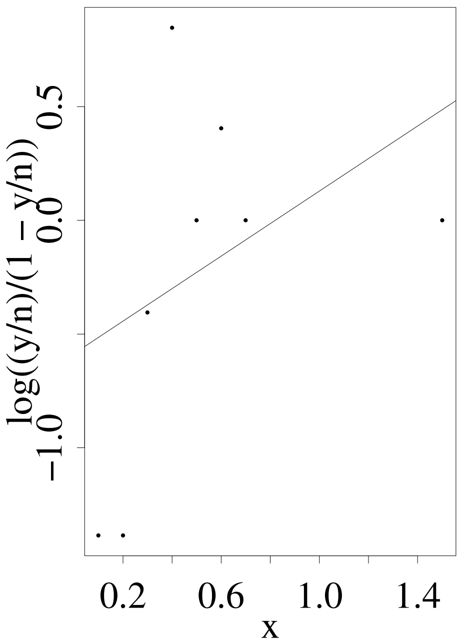
5.5 Identification of influential observations

Une observation influente n'est pas nécessairement détectée par l'examen des résidus. D'autres méthodes de détection doivent donc être mises en œuvre. Lorsqu'une observation est distante des autres en terme de valeurs prises par les prédicteurs, elle peut être influente. Cette distance peut être mesurée par la fonction d'influence (leverage) : h. La valeur de h dépend essentiellement des valeurs prises par les variables explicatives. Lorsque h est élevé, l'observation peut être influente, mais ne le sera pas forcément.
Fonction d'influence h (leverage)
C'est un jeu de données artificiel qui est utilisé (Collett, exemple 5.11, p.146).

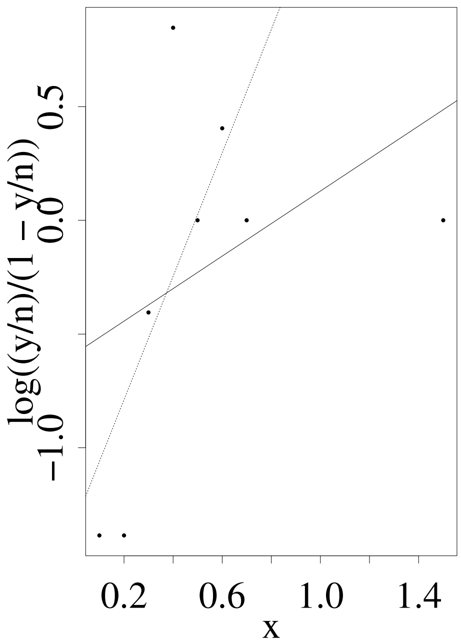
```
> artificial_data.frame(
+   x=c(seq(0.1,0.7,length=7),1.5),
+   y=c(2,2,4,7,5,6,5,5),
+   n=rep(10,8))
```

```
> glm45_glm(cbind(y,n-y)~x,binomial,artificial)

> summary(glm45)
Call: glm(formula = cbind(y, n - y) ~ x, family = binomial, data = artificial)
Deviance Residuals:
       Min         1Q     Median        3Q      Max
 -1.296788 -0.8719498 0.04188649 0.4918577 1.752778

Coefficients:
                 Value Std. Error   t value
(Intercept) -0.5865552  0.3779584 -1.551904
          x  0.7152902  0.5620748  1.272589

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 9.35953 on 7 degrees of freedom

Residual Deviance: 7.691394 on 6 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
  (Intercept)
x -0.7993348

> h_lm.influence(glm45)$h

> cbind(10*fitted(glm45),h,
+    residuals(glm45,"pearson")/sqrt(1-h),
+    residuals(glm45,"deviance")/sqrt(1-h))
1 3.740160 0.2623473 -1.32412822 -1.38777814
2 3.909080 0.2084960 -1.39063865 -1.45761349
3 4.080655 0.1676348 -0.05688162 -0.05694163
4 4.254501 0.1406823  1.89432541  1.89081895
5 4.430211 0.1283693  0.38852220  0.38725380
6 4.607357 0.1312183  0.94788910  0.94712111
7 4.785502 0.1495276  0.14723939  0.14717152
8 6.192533 0.8117244 -1.78982376 -1.76372683
```

```
> attach(artificial)
> plot(x,log((y/n)/(1-y/n)))
> abline(glm45$coef)
```

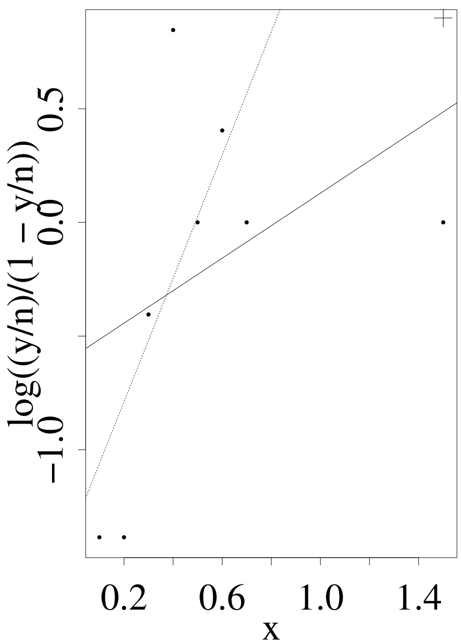



```
> glm45.1_glm(cbind(y,n-y)~x,binomial,artificial,subset=-8)

> summary(glm45.1)
Call: glm(formula = cbind(y, n - y) ~ x, family = binomial, data = artificial, subset
         = -8)
Deviance Residuals:
          1          2         3        4           5         6          7
 -0.4244501 -0.7983054 0.1734407 1.667863 -0.04017407 0.1685297 -0.8932203

Coefficients:
                Value Std. Error   t value
(Intercept) -1.332783  0.5837367 -2.283193
          x  2.716384  1.2809928  2.120530

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 9.244582 on 6 degrees of freedom

Residual Deviance: 4.457158 on 5 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
  (Intercept)
x -0.9043612

> abline(glm45.1$coef,lty=2)
> detach()
```



Si la 8e observation avait été 9/10 au lieu de 5/10, la valeur de h aurait toujours été élevée par rapport aux autres valeurs, mais l'observation aurait été moins influente.

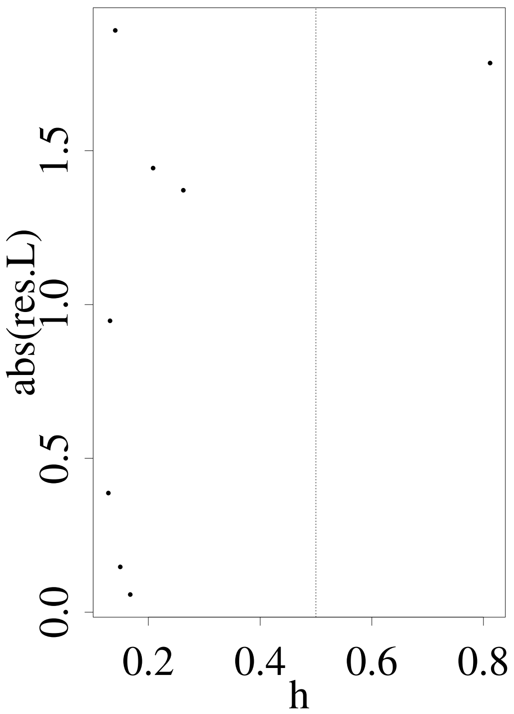
```
> artificial1_artificial
> artificial1[8,"y"]_9
> glm45.2_glm(cbind(y,n-y)~x,binomial,artificial1)
> points(1.5,9/10,pch=3)
```




```
lm.influence(glm45.2)$h
[1] 0.2892208 0.2275633 0.1781359 0.1499168 0.1490226 0.1769979 0.2302694
[8] 0.5988733
```

On peut faire un “index-plot” des valeurs de h. Les valeurs de h supérieures à 2p/n (p/n est la moyenne des éléments diagonaux de la matrice H) peuvent être considérées comme particulièrement grandes. Une représentation plus informative est le graphe des résidus (de vraisemblance par exemple) en fonction de h. Dans l'exemple artificiel, cette représentation montre qu'une observation a une valeur de h particulièrement élevée, et que le résidus correspondant est aussi assez grand.

```
> h_lm.influence(glm45)$h
> res.L_residuals.my(glm45,type="likelihood")
> plot(h,abs(res.L))
> abline(v=2*2/8,lty=2)
```

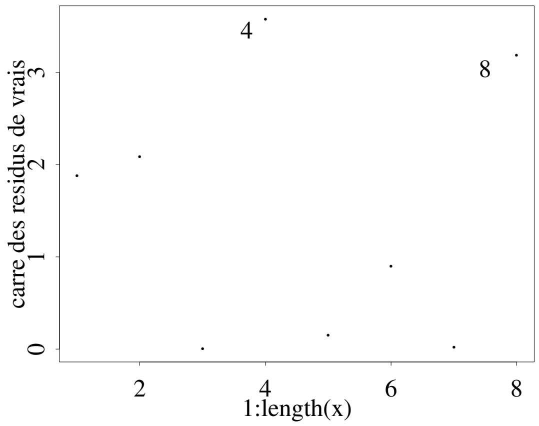


5.5.1 Influence of the ith observation on the overall fit of the model

carrés des résidus de vraisemblance

L'influence de la i[e] observation sur l'ajustement global peut être mesurée par la réduction de déviance, lorsqu'on omet cette observation. La différence de déviance lorsqu'on omet une observation peut être approchée par le carré du résidu de vraisemblance. Les résidus de vraisemblance sont plus faciles à calculer que les différences de déviances, qui nécessitent l'estimation d'autant de modèle qu'il y a d'observations. Un index plot des carrés des résidus de vraisemblance est un bon outil pour étudier la contribution de chaque observation à l'ajustement global.

```
> res.L_residuals.my(glm45,type="likelihood")
> index.plot(res.L^2,ylab="carre des residus de vrais")
```



5.5.2 Influence of the ith observation on the set of parameter estimates

distance de Cook, statistique D_i
Avec le jeu de données "artificial", la 8e observation a une forte influence sur l'ensemble des paramètres. Ce n'est plus le cas avec "artificial1" : la 8e observation n'est pas influente, en dépit de valeurs éloignées des variables explicatives.

```
> cook45_cook.calc(glm45)
> cook45.2_cook.calc(glm45.2)
> round(cbind(cook45,cook45.2),2)
  cook45 cook45.2
1   0.31     0.08
2   0.25     0.13
3   0.00     0.00
4   0.29     0.29
5   0.01     0.00
6   0.07     0.01
7   0.00     0.11
8   6.91     0.07
```

5.5.3 Influence of the ith observation on the jth parameter estimate

5.5.4 Influence of the ith observation on the comparison of two deviances

statistique L_i : différence des carrés des résidus de vraisemblance
Exemple "artificial". Quelle est l'influence des observations sur la différence de déviance entre les modèles incluant et n'incluant pas la variable explicative x ?
On ne trouve pas les même résidus de vraisemblance que Collett, pour le modèle glm45.0. Cela semble dû au calcul des résidus de déviance, lorsque le modèle est nul.

```
> glm45.0_glm(cbind(y,n-y)~1,binomial,artificial)
```

```
> res.L_residuals.my(glm45,"likelihood")
> res.L.0_ residuals.my(glm45.0,"likelihood")

> cbind(res.L.0,res.L,res.L^2 - res.L.0^2)
      res.L.0       res.L
1 -0.7974257 -1.37136564  1.244755915
2 -0.7974257 -1.44390586  1.448976333
3 -0.1568988 -0.05693158 -0.021376034
4  0.7846239  1.89131264  2.961428765
5  0.1564620  0.38741686  0.125611470
6  0.4693886  0.94722192  0.676903739
7  0.1564620  0.14718167 -0.002817906
8  0.1564620 -1.78493950  3.161528664
```

On peut calculer les résidus de vraisemblance exacts, on obtient alors des valeurs plus proches de celles de Collett. Quelle que soit la façon de calculer les résidus de vraisemblance, la conclusion est la même. La 8^e observation influence fortement la différence de déviance.

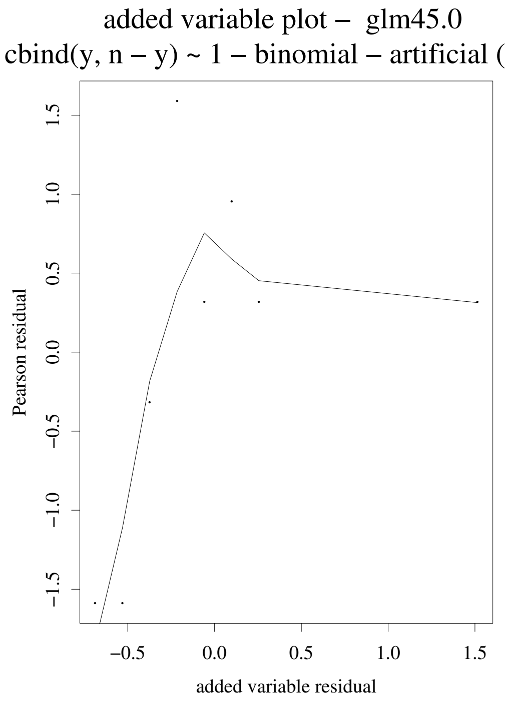
```
> res.L_NULL; res.L.0_NULL
> res.P_resid(glm45,type="pearson")
> res.P.0_resid(glm45.0,type="pearson")
> for (i in 1:8)
+   {res.L_c(res.L,sqrt(glm45$dev-update(glm45,subset=-i)$dev))
+    res.L.0_c(res.L.0,sqrt(glm45.0$dev-update(glm45.0,subset=-i)$dev)) }
+   res.L_sign(res.P)*res.L
+   res.L.0_sign(res.P.0)*res.L.0

> cbind(res.L.0,res.L,res.L^2 - res.L.0^2)
1 -1.7637197 -1.36962826 -1.23482550
2 -1.7637197 -1.44247508 -1.02997273
3 -0.3410116 -0.05692965 -0.11304790
4  1.7063581  1.89309773  0.67216113
5  0.3390395  0.38745775  0.03517574
6  1.0172719  0.94727375 -0.13751463
7  0.3390395  0.14717458 -0.09328742
8  0.3390395 -1.79839819  3.11928828
```

La différence de déviance, qui n'est pas significative avec toutes les observations, devient significative, lorsqu'on enlève la 8^e observation.

```
> anova(glm45.0,glm45,test="Chi")
Analysis of Deviance Table
Response: cbind(y, n - y)
  Terms Resid. Df Resid. Dev Test Df Deviance   Pr(Chi)
1     1         7   9.359530
2     x         6   7.691394       1 1.668136 0.1965084

> anova(update(glm45.0,subset=-8),update(glm45,subset=-8),test="Chi")
Analysis of Deviance Table
Response: cbind(y, n - y)
  Terms Resid. Df Resid. Dev Test Df Deviance    Pr(Chi)
1     1         6   9.244582
2     x         5   4.457158       1 4.787424 0.02866827
```

On peut utiliser un autre outil, pour juger de la nécessité de tenir compte de la variable explicative x dans le modèle : le "added variable plot". Ce graphe montre clairement que sans la 8^e observation, on conclue à la nécessité de prendre en compte la variable x, alors qu'avec la 8^e observation, l'ajout de la variable x n'est pas intéressant. La conclusion basée sur les différences des carrés des résidus de vraisemblance, est confirmée par le “added variable plot”.

```
> attach(artificial)
> added.var.plot(glm45.0,x)
> detach()
```



5.5.5 Influence of the ith observation on the goodness of link test

5.5.6 Influence of the ith observation on the choice of transformation of an explanatory variable

5.5.7 Influence of the ith observation on the fit at the ith point

5.5.8 Treatment of influential values

Quel sort donner aux observations influentes ? si une ou plusieurs observations influentes ont été détectées, rechercher d'abord une erreur éventuelle dans les données brutes. Se poser des questions sur les sources d'erreur possibles, lors du recueil des données. Si aucune erreur, ou aucune explication n'est trouvée, ajuster les modèles avec et sans observations influentes. C'est important, parce que plusieurs des méthodes de diagnostic proposées reposent sur des approximations. Si les observations influentes ont un impact important sur les résultats, présenter et discuter les deux types de résultats, avec et sans observations influentes. Ce n'est pas une bonne pratique statistique que d'écarter systématiquement les observations influentes.

5.6 Checking the assumption of a binomial distribution

surdispersion et sousdispersion
En cas de surdispersion, la déviance obtenue après ajustement d'un modèle adéquat est plus grande que ce qu'elle aurait dû être sous l'hypothèse d'une distribution binomiale.

L'exemple support est l'exemple "rats" (Collett, exemple 1.4, p.4). On observe la proportion de rats survivant dans des portées, les rattes ayant été affectées à deux groupes dont les régimes alimentaires diffèrent.

```
> glm47_glm(cbind(y,n-y)~trait,binomial,rats)

> summary(glm47)
Call: glm(formula = cbind(y, n - y) ~ trait, family = binomial, data = rats)
Deviance Residuals:
       Min         1Q    Median       3Q     Max
 -4.552266 -0.7861691 0.2573085 1.326936 2.48951

Coefficients:
                 Value Std. Error    t value
(Intercept)  1.7025332  0.1640053  10.380966
      trait -0.4805458  0.1640053  -2.930063

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 95.20312 on 31 degrees of freedom

Residual Deviance: 86.18708 on 30 degrees of freedom

Number of Fisher Scoring Iterations: 3

Correlation of Coefficients:
      (Intercept)
trait -0.2722357

> qchisq(.95,30)
[1] 43.77297

> 1-pchisq(glm47$dev,30)
[1] 2.455726e-07
```

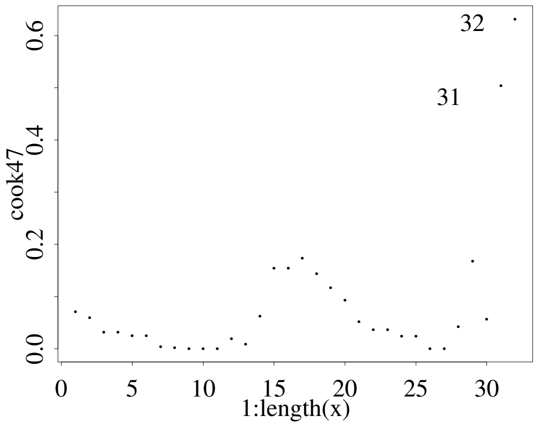
La déviance est très élevée par rapport au nombre de degrés de liberté. Si on suppose le modèle adéquat, la déviance est significativement trop grande. On ne dispose d'aucune variable explicative supplémentaire, qui permettrait d'améliorer le modèle. On pourait bien sûr estimer une proportion pour chacune des portées. Mais alors, le modèle serait saturé, le nombre d'observations étant égal au nombre de paramètres, et la déviance serait nulle. On peut aussi se demander si le fait que cette déviance est si élevée, peut être attribué à quelques observations particulières.

```
> fitted(glm47)[!duplicated(rats$trait)]
         1        17
 0.8987197 0.7724131
> h_lm.influence(glm47)$h
> res.D.stand_residuals(glm47,"deviance")/sqrt(1-h)
> cook47_cook.calc(glm47)
> attach(rats)
> b_cbind(trait,y,n,fitted(glm47)*n,h,res.D.stand,cook47)
> b_round(b,3)
> dimnames(b)[[2]]_c("trait","y","n","y.chap","h","res.D","Cook")
> b
   trait  y  n y.chap     h  res.D  Cook
 1     1 13 13 11.683 0.082  1.739 0.070
 2     1 12 12 10.785 0.076  1.665 0.059
 3     1  9  9  8.088 0.057  1.428 0.032
```

```
 4      1  9  9  8.088 0.057  1.428 0.032
 5      1  8  8  7.190 0.051  1.342 0.025
 6      1  8  8  7.190 0.051  1.342 0.025
 7      1 12 13 11.683 0.082  0.316 0.004
 8      1 11 12 10.785 0.076  0.220 0.002
 9      1  9 10  8.987 0.063  0.014 0.000
10      1  9 10  8.987 0.063  0.014 0.000
11      1  8  9  8.088 0.057 -0.099 0.000
12      1 11 13 11.683 0.082 -0.614 0.019
13      1  4  5  4.494 0.032 -0.666 0.009
14      1  5  7  6.291 0.044 -1.392 0.062
15      1  7 10  8.987 0.063 -1.795 0.154
16      1  7 10  8.987 0.063 -1.795 0.154
17      2 12 12  9.269 0.083  2.599 0.174
18      2 11 11  8.497 0.076  2.479 0.144
19      2 10 10  7.724 0.069  2.355 0.117
20      2  9  9  6.952 0.062  2.226 0.093
21      2 10 11  8.497 0.076  1.241 0.052
22      2  9 10  7.724 0.069  1.090 0.037
23      2  9 10  7.724 0.069  1.090 0.037
24      2  8  9  6.952 0.062  0.931 0.024
25      2  8  9  6.952 0.062  0.931 0.024
26      2  4  5  3.862 0.034  0.152 0.000
27      2  7  9  6.952 0.062  0.040 0.000
28      2  4  7  5.407 0.048 -1.207 0.043
29      2  5 10  7.724 0.069 -1.945 0.168
30      2  3  6  4.634 0.041 -1.485 0.057
31      2  3 10  7.724 0.069 -3.286 0.504
32      2  0  7  5.407 0.048 -4.666 0.632
> detach()
```

Les observations 31 et 32 ont une grande valeur de la distance de Cook D. Le fait de les écarter diminue la déviance, mais pas suffisamment pour que l'on puisse accepter l'hypothèse d'une distribution binomiale.

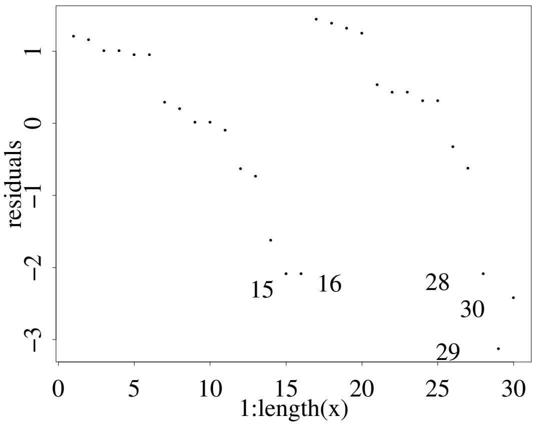
```
> index.plot(cook47)
```




```
> glm47.1_glm(cbind(y,n-y)~trait,binomial,rats,subset=-c(31,32))

> summary(glm47.1)
Call: glm(formula = cbind(y, n - y) ~ trait, family = binomial, data = rats, subset
         =  - c(31, 32))
Deviance Residuals:
       Min         1Q    Median       3Q      Max
 -2.611473 -0.5883817 0.3149099 1.366478 1.963766

Coefficients:
                 Value Std. Error    t value
(Intercept)  1.9650736  0.1811997  10.844794
      trait -0.2181647  0.1811997  -1.204001

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 51.8381 on 29 degrees of freedom

Residual Deviance: 50.38205 on 28 degrees of freedom

Number of Fisher Scoring Iterations: 4

Correlation of Coefficients:
      (Intercept)
trait -0.0588844

> 1-pchisq(glm47.1$dev,28)
[1] 0.005860943
```

Les observations 15,16,28,29,30 ont des résidus supérieurs à 2, en valeur absolue.

```
> index.plot(residuals(glm47.1,"pearson"))
```



Cependant, on ne va pas continuer à exclure des observations, jusqu'à obtenir satisfaction. Il faut conclure à une surdispersion, particulièrement dans le groupe contrôle. Cette surdispersion est vraisemblablement due à des corrélations entre les réponses binaires obtenues sur les individus d'une même portée.

5.7 Model checking for binary data

5.7.1 Checking the form of the linear predictor

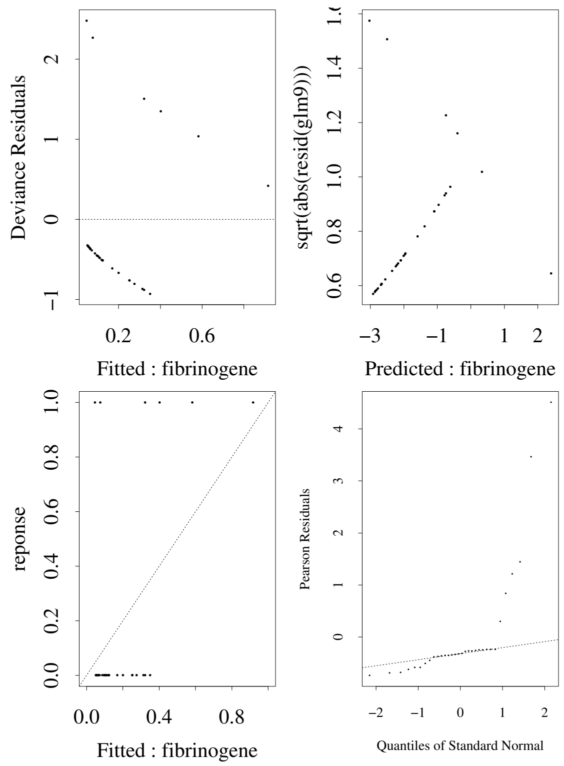
On utilise l'exemple ESR (p. 122).

```
> glm9_glm(reponse~fibrinogene,binomial,ESR)

> summary(glm9)
Call: glm(formula = reponse ~ fibrinogene, family = binomial, data = ESR)
Deviance Residuals:
        Min         1Q     Median         3Q      Max
 -0.9298272 -0.5399198 -0.4381977 -0.3356489 2.479406

Coefficients:
                Value Std. Error   t value
(Intercept) -6.844796  2.7543984 -2.485042
fibrinogene  1.826996  0.8960538  2.038935

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 30.88496 on 31 degrees of freedom

Residual Deviance: 24.84036 on 30 degrees of freedom

Number of Fisher Scoring Iterations: 4
```

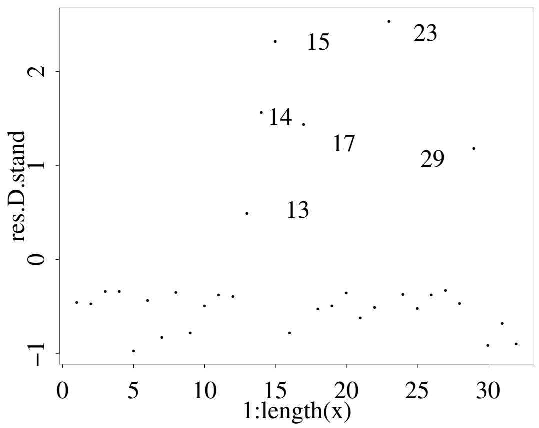
```
Correlation of Coefficients:
            (Intercept)
fibrinogene -0.9826252
```

```
> h_lm.influence(glm9)$h
> fit_fitted(glm9)
> res.D.stand_residuals(glm9,"deviance")/sqrt(1-h)

> round(cbind(fit,h,res.D.stand),3)
     fit     h res.D.stand
 1 0.096 0.041      -0.459
 2 0.103 0.040      -0.475
 3 0.055 0.043      -0.344
 4 0.054 0.043      -0.341
 5 0.351 0.087      -0.973
 6 0.087 0.041      -0.436
 7 0.277 0.059      -0.829
 8 0.057 0.043      -0.350
 9 0.252 0.052      -0.782
10 0.110 0.040      -0.492
11 0.065 0.043      -0.376
12 0.072 0.042      -0.396
13 0.917 0.280       0.491
14 0.322 0.075       1.564
15 0.076 0.042       2.319
16 0.252 0.052      -0.782
17 0.402 0.113       1.433
18 0.125 0.039      -0.527
19 0.110 0.040      -0.492
20 0.059 0.043      -0.356
21 0.170 0.039      -0.624
22 0.119 0.039      -0.513
23 0.046 0.043       2.534
24 0.064 0.043      -0.372
25 0.123 0.039      -0.522
26 0.065 0.043      -0.376
27 0.051 0.043      -0.332
28 0.099 0.040      -0.467
29 0.583 0.225       1.180
30 0.322 0.075      -0.917
31 0.201 0.042      -0.684
32 0.314 0.072      -0.902
```

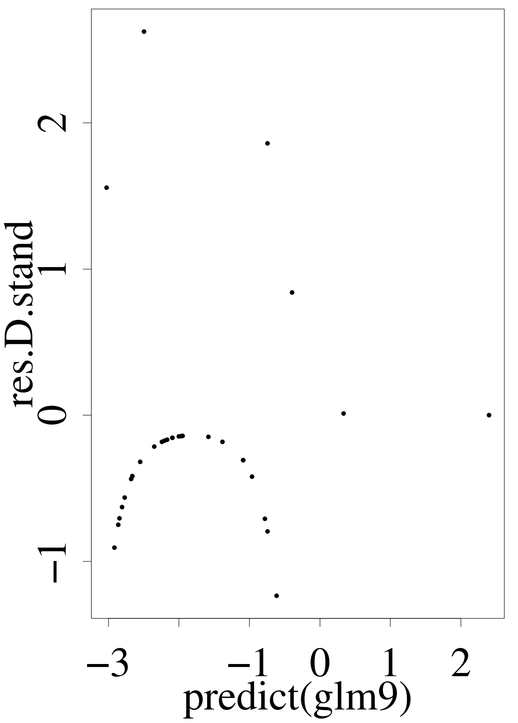
```
> par(mfrow=c(2,2))
> plot(glm9)
```



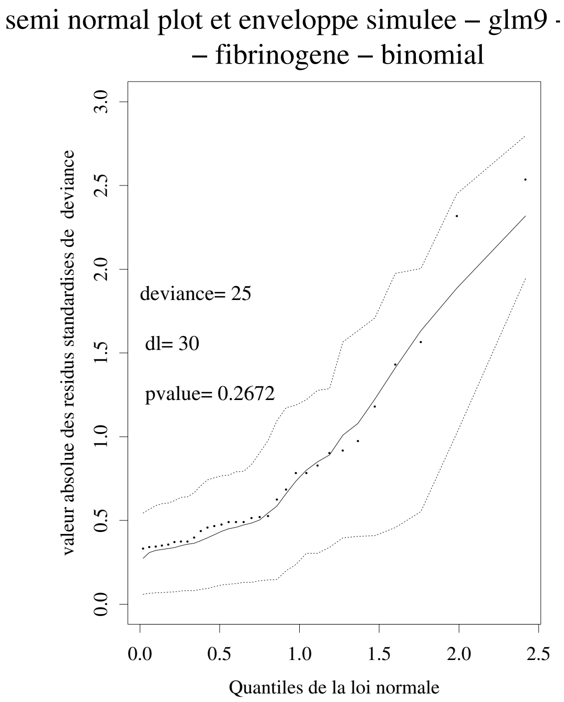

```
> index.plot(res.D.stand)
```



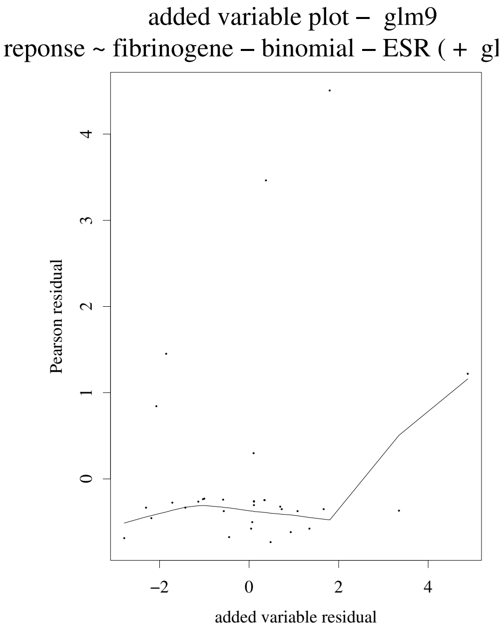

```
> plot(predict(glm9),res.D.stand)
```



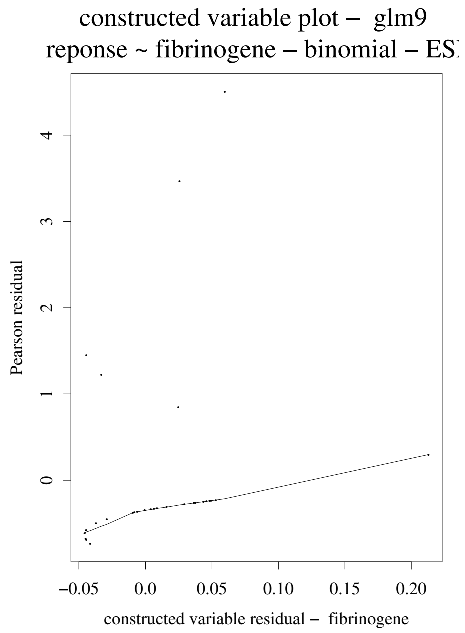

```
> ESR$n_rep(1,32)
> env9_enveloppe(glm9)
```




```
> attach(ESR)
> added.var.plot(glm9,globuline)
```



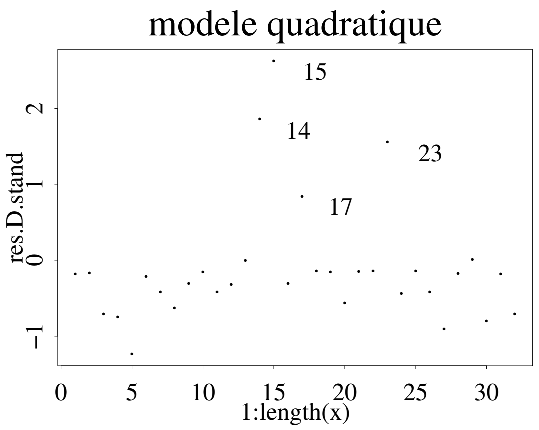

```
> constr.var.plot(glm9,fibrinogene)
$pente:
        X
 5.271917

$lambda:
[1] "? pente - 1 ?"

$transfo:
[1] "(x^lambda-1)/lambda"
> detach()
```



5.7.2 Checking the adequacy of the link function

Il y a deux façons d'améliorer la qualité de l'ajustement : introduire un effet quadratique de fibrinogène (pourquoi pas, on peut toujours essayer !), ou écarter les observations 15 et 23. L'ajout du terme quadratique conduit à un modèle instable, car surparamétré, et la déviance est peu réduite, et le point 15 reste un point extérieur (outlier).

```
> glm9.1_glm(reponse~poly(fibrinogene,2),binomial,ESR)

> anova(glm9,glm9.1,test="Chi")
Analysis of Deviance Table
Response: reponse
                 Terms Resid. Df Resid. Dev    Test Df Deviance     Pr(Chi)
1          fibrinogene        30   24.84036
2 poly(fibrinogene, 2)        29   16.96748 1 vs. 2  1 7.872873 0.005018187

> h_lm.influence(glm9.1)$h

> res.D.stand_residuals(glm9.1,"deviance")/sqrt(1-h)
Warning messages:
  fitted values close to 0 or 1 in: family$deviance(mu, y, w, residuals = T)

> index.plot(res.D.stand,main="modele quadratique")
```



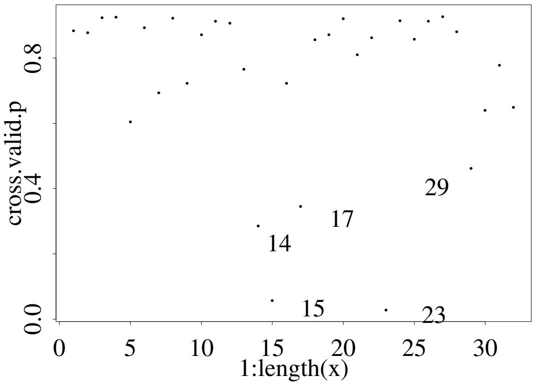

```
> glm9.2_glm(reponse~fibrinogene,binomial,ESR,subset=-c(15,23))
```

5.7.3 Identification of outliers

Dans le cas de données binaires (contrairement aux données binomiales), ne peuvent être "outliers" que des observations prenant des valeurs extrêmes pour les variables explicatives. Les "outliers" peuvent être détectés à partir d'un "index plot" des résidus de déviance standardisés, ou des résidus de vraisemblance, ou bien à l'aide d'un "half-normal plot" et de son enveloppe simulée.

Une autre méthode peut être utilisée : une méthode de validation croisée. La méthode consiste à calculer, pour chaque observation, la probabilité d'obtenir la valeur obtenue, étant données toutes les autres observations. Une approximation de cette probabilité se calcule facilement.

```
> res.L_residuals.my(glm9,"likelihood")
> h_lm.influence(glm9)$h
> cross.valid.p_exp(-(res.L^2)/2)*sqrt(1-h)
> index.plot(cross.valid.p)
```



On peut considérer les points extérieurs comme le résultat d'erreurs de transcription, et introduire une modélisation de l'erreur de transcription, qui revient à considérer une nouvelle fonction de lien, dans laquelle intervient la probabilité d'erreur de transcription γ. Le taux d'erreurs de transcription peut être estimé, et le nombre de points extérieurs peut être déduit.

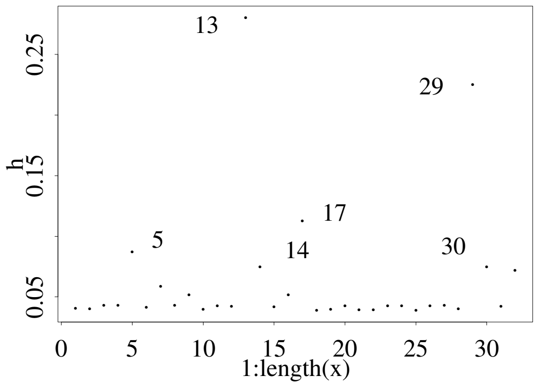
5.7.4 Identification of influential values

On utilise l'exemple ESR (p. 122).

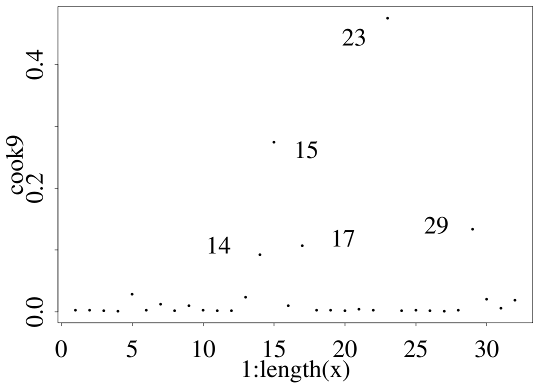
```
> h_lm.influence(glm9)$h
> cook9_cook.calc(glm9)
> L_residuals.my(glm9,"likelihood")^2 - residuals.my(glm9.1,"likelihood")^2
Warning messages:
  fitted values close to 0 or 1 in: family$deviance(mu, y, w, residuals = T)
```

Les observations 13 et 29 ont une valeur de h élevée, mais une valeur de D (distance de Cook) pas très élevée. Elles sont donc peu influentes malgré un h élevé. Par contre, les observations 15 et 23 sont influentes.

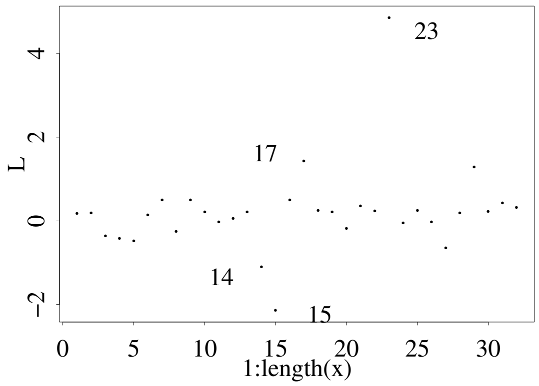
```
> index.plot(h)
```




```
> index.plot(cook9)
```




```
> index.plot(L)
```



5.7.5 Influence of the ith observation on the classification of future individuals

5.8 Summary and recommendations

Un tableau récapitulatif des méthodes de validation est donné par Collett (p.178).

5.9 A futher example on the use of diagnostics

Annexe 1

Quelques fonctions nécessaires pour le chapitre 4. Le texte qui suit, c'est à dire la définition de ces fonctions, est supposé stocké dans le fichier "fcts.chap4".

```
vcov_function(glm2,surdisp=F)
{result_summary(glm2)
if (surdisp) dispersion_glm2$deviance/glm2$df
else dispersion_result$dispersion
dispersion*result$cov.unscaled
}

########################################
dose.p_function(glm,p=0.5)
{
eta_family(glm)$link(p)
b_coef(glm)
dose_(eta-b[1])/b[2]
names(dose)_paste("p = ",format(p),":",sep="")
pd_cbind(1,dose)/b[2]
vr_(pd %*% vcov(glm) %*% t(pd))
attr(dose,"SE")_sqrt(diag(vr))
attr(dose,"p")_p
class(dose)_"glm.dose"
dose
}


########################################
IC.ED.approx_function(glm,p=c(0.25,0.5,0.75))
{
don_get(glm$call$data)
b_dose.p(glm,p)
IC_exp(cbind(b-qnorm(0.975)*attr(b,"SE"),b,
             b+qnorm(0.975)*attr(b,"SE")))
IC_cbind(IC,p)
dimnames(IC)[[2]]_c("dose.inf","dose","dose.sup","p")
IC
}

########################################
IC.ED.vrais_function(glm1,IC.approx,p=IC.approx[,"p"])
{
don_get(glm1$call$data)
IC_log(IC.approx[1,1:3])
grille_seq(IC[1]-diff(range(IC))/4,IC[3]+diff(range(IC))/4,length=20)
dev_NULL
```

```
offset0_rep(log((p)/(1-p)),nrow(don))
assign("offset0",offset0,frame=1)
for (i in 1:20)
{ldose_log(don$dose)-grille[i]; assign("ldose",ldose,frame=1)
 dev_c(dev,glm(cbind(y,n-y)~-1+offset(offset0)+ldose,family=binomial,
                                                data=don)$deviance) }
plot(grille,dev)
abline(h=glm1$deviance+qchisq(.95,1))
ldose.inf_approx(dev[1:10],grille[1:10],glm1$deviance+qchisq(.95,1))$y
ldose.sup_approx(dev[11:20],grille[11:20],glm1$deviance+qchisq(.95,1))$y
IC_cbind(exp(ldose.inf),IC.approx[1,2],exp(ldose.sup),IC.approx[1,"p"])
dimnames(IC)_list(paste("p =",p),c("dose.inf","dose","dose.sup","p"))
attr(IC,"p0")_attr(IC.approx,"p0")
IC
}

########################################
logit.my_function(mu0=0.10)
{
assign("mu0",mu0,frame=1)
        binom.link <- list(names =
                "Logit: reponse naturelle : log((mu-mu0)/(1-mu))", link =
                function(mu)
        {
                log((mu - mu0)/(1 - mu))
        }
        , inverse = function(eta)
        {
                junk <- care.exp(eta)
                (junk + mu0)/(1 + junk)
        }
        , deriv = function(mu)
        {
                d <- (mu - mu0) * (1 - mu)
                if(any(tiny <- (d < .Machine$double.eps))) {
                        warning("Model unstable; fitted probabilities of 0 or 1"
                                )
                        d[tiny] <- .Machine$double.eps
                }
                (1 - mu0)/d
        }
        , initialize = expression({
                if(is.matrix(y)) {
                        if(dim(y)[2] > 2)
                                stop(
                                  "only binomial response matrices (2 columns)"
                                  )
                        n <- drop(y %*% c(1, 1))
                        y <- y[, 1]
                }
                else {
                        if(is.category(y))
                                y <- y != levels(y)[1]
                        else y <- as.vector(y)
                        n <- rep(1, length(y))
                }
```

```
                w <- w * n
                n[n == 0] <- 1
                y <- y/n
                mu <- y + (0.5 - y)/n
                mu <- mu0 + (1 - mu0) * mu
        }
        ),  )
        binom.var <- list(names = "", variance = binomial()$variance, deviance
                 = binomial()$deviance)
        make.family("binomiale - logit - reponse naturelle", link = binom.link,
                variance = binom.var)
}
```

Annexe2

Quelques fonctions nécessaires pour le chapitre 5. Le texte qui suit, c'est à dire la définition de ces fonctions, est supposé stocké dans le fichier "fcts.chap5".

```
enveloppe_function(glm1, plot.it = T, labels = T,type="deviance",...)
{
# Representation graphique du half-normal plot des residus
# (de deviance par defaut) et d'une enveloppe simulee.
# Modelling binary data, Collett (p.129).

# glm1 : objet obtenu en sortie d'une commande glm.
# le data-frame des donnees (utilise pour creer glm1) doit contenir
# une variable n (contenant les ni).

# c'est  un peu long,  car cela necessite 19  ajustements du meme modele
# sur des donnees simulees. Il est donc conseille  de SAUVER LE RESULTAT
# de la  fonction    enveloppe, dans  un objet  S.    Si plot.if=T,  une
# representation graphique  du resultat  est   faite immediatement.   On
# pourra  a nouveau representer les   resultats en utilisant la fonction
# enveloppe.plot.

assign("glm1",glm1,frame=0)
don <- get(glm1$call$data)
n <- don$n
p <- fitted(glm1)
p[p == 1] <- 0.999999
p[p == 0] <- 0.000001
h_lm.influence(glm1)$h
binom.simul <- NULL
for(i in 1:length(n)) binom.simul <- rbind(binom.simul, rbinom(19, n[i], p[i]))
res.simul <- NULL
for(i in 1:19)
        {don.simul_cbind(binom.simul[, i], n - binom.simul[, i])
         glm.0 <- glm(don.simul ~ model.matrix(glm1),family = glm1$call$family)
         if (type=="likelihood")
                {res.D.stand_residuals(glm.0,"deviance")/sqrt(1-h)
                 res.P.stand_residuals(glm.0,"pearson")/sqrt(1-h)
                 residus_sign(res.P.stand)*sqrt(h*res.P.stand^2+(1-h)*res.D.stand^2)
                }
         else residus_residuals(glm.0, type = type)/sqrt(1-h)
         res.simul <- cbind(res.simul, residus)
        }
res.simul <- abs(res.simul)
for(i in 1:19) res.simul[, i] <- sort(res.simul[, i])
res.simul <- cbind(res.simul, apply(res.simul, 1, mean),
                              apply(res.simul, 1, min),
```

```
                              apply(res.simul, 1, max))
if (type=="likelihood")
        {res.D.stand_residuals(glm1,"deviance")/sqrt(1-h)
         res.P.stand_residuals(glm1,"pearson")/sqrt(1-h)
         residus_sign(res.P.stand)*sqrt(h*res.P.stand^2+(1-h)*res.D.stand^2)
        }
else residus <- residuals(glm1, type = type)/sqrt(1-h)
qqpl <- qqplot(qnorm((ppoints(residus) + 1)/2), abs(residus), plot = F)
if(plot.it && !is.null(dev.list()))
            {plot(qqpl, ylim = c(0,ceiling( max(c(res.simul[, 22],residus)))),
                  xlab = "", ylab = "")
             if(labels)
                title(xlab = "Quantiles de la loi normale",
                ylab = paste("valeur absolue des residus standardises de ",type),
                main = paste("semi normal plot et enveloppe simulee",
                             substitute(glm1),glm1$call$data,"\n",
                             glm1$call$formula[3],glm1$call$family,sep = " - "),
                ...)
             else  title(main = paste(substitute(glm1),glm1$call$data,"\n",
                                   glm1$call$formula[3],glm1$call$family,sep = " - "),
                 ...)
             lines(qqpl$x, res.simul[, 20], lty = 1)
             lines(qqpl$x, res.simul[, 21], lty = 2)
             lines(qqpl$x, res.simul[, 22], lty = 2)
             }
dev <- deviance(glm1)
dl <- glm1$df.res
invisible(list(type=type,qqpl = qqpl, res.simul = res.simul,
               modele = substitute(glm1), call = glm1$call, deviance = dev,
               dl = dl, pvalue = pchisq(dev, dl)))
}

########################################
enveloppe.plot_function(env, labels = T, ...)
{
qqpl <- env$qqpl
res.simul <- env$res.simul
plot(qqpl, ylim = c(0, ceiling(c(max(res.simul[, 22],qqpl$y)))), xlab = "", ylab = "")
par(...)
if(labels)
        title(xlab = "Quantiles de la loi normale",
              ylab = paste("valeur absolue des residus standardises de ",env$type),
              main = paste("semi normal plot et enveloppe simulee",
                           env$modele,env$call$data,"\n",
                           env$call$formula[3],env$call$family,sep = " - "))
else title(main = paste(env$modele,env$call$data,"\n",
                        env$call$formula[3],env$call$family,sep = " - "))
deviance <- paste("deviance=", round(env$dev), "\n\n", "dl=", env$dl,
                  "\n\n", "pvalue=", round(env$pvalue, 4))
text(x = 0, y = 2/3 * max(res.simul[, 22]), deviance, adj = 0,cex=1.5)
lines(qqpl$x, res.simul[, 20], lty = 1)
lines(qqpl$x, res.simul[, 21], lty = 2)
lines(qqpl$x, res.simul[, 22], lty = 2)
}

########################################
```

```
residuals.my_function(glm1,type="deviance")
{
if (type=="likelihood")
{h_lm.influence(glm1)$h
 res.D.stand_residuals(glm1,"deviance")/sqrt(1-h)
 res.P.stand_residuals(glm1,"pearson")/sqrt(1-h)
 result_sign(res.P.stand)*sqrt(h*res.P.stand^2+(1-h)*res.D.stand^2)
}
else result_residuals(glm1,type=type)
result
}

########################################
added.var.plot_function(glm1, cc2,...)
{
res.P <- residuals(glm1, type = "pearson")
fit <- fitted(lm(cc2 ~ -1 + model.matrix(glm1),weights = glm1$weights))
u.res <- (cc2 - fit) * sqrt(glm1$weight)
plot(u.res, res.P,
        main = paste("added variable plot - ", substitute(glm1), "\n",
                     paste(glm1$call[-1], collapse = " - "),
                     "( + ", substitute(cc2), ")"),
        ylab = "Pearson residual", xlab = "added variable residual",...)
lines(lowess(u.res, res.P))
invisible(list(pearson=res.P,"u-residual"=u.res))
}

########################################
partial.res.plot_function(glm1, variable, ...)
{
# variable : nom de la variable remise en cause
res.p <- predict(glm1, type = "terms",terms=variable) + glm1$residuals
x <- model.matrix(glm1)[,variable]
plot(x, res.p, ylab = "residus partiels",
               xlab = variable,
               main = paste("Partial residual plot  -  ", substitute(glm1),"\n",
                            paste(glm1$call[-1], collapse = " , ")), ...)
lines(lowess(x, res.p))
invisible()
}

########################################
constr.var.plot_function(glm1, cc, ...)
{
z <- coef(glm1)[paste(substitute(cc))] * cc * log(cc)
u.res <- (z - fitted(lm(z ~ -1+model.matrix(glm1), weights = glm1$weights)))
u.res <- u.res * sqrt(glm1$weight)
res.P <- residuals(glm1, type = "pearson")
plot(u.res, res.P,
     main = paste("constructed variable plot - ", substitute(glm1), "\n",
                  paste(glm1$call[-1], collapse = " - ")),
     ylab = "Pearson residual",
     xlab = paste("constructed variable residual - ", substitute(cc)), ...)
lines(lowess(u.res, res.P))
# lines(lowess(u.res, res.P,iter=0))
# abline(lsfit(u.res, res.P))
```

```
list(pente = lsfit(u.res, res.P)$coef[2], lambda = "? pente - 1 ?",
     transfo = "(x^lambda-1)/lambda")
}


########################################
index.plot_function(x,...)
{plot(1:length(x),x,...,ylab=paste(substitute(x)))
 identify(1:length(x),x)
 invisible()
}

########################################
cook.calc_function(glm1)
{
res.P_residuals(glm1,"pearson")
p_length(glm1$coef)
h_lm.influence(glm1)$h
cook_h*res.P^2/(p*(1-h)^2)
invisible(cook)
}

########################################
D.calc_function(glm1)
{cook_cook.calc(glm1)
invisible(cook)
}
```

Table des matières