VGAM Family Functions for Genetic Data

T. W. Yee

October 13, 2006

Beta Version 0.6-5

Department of Statistics,
University of Auckland,
New Zealand
yee@stat.auckland.ac.nz
http://www.stat.auckland.ac.nz/∼yee

Contents

[Important note: This document and code is not yet finished, but should be completed one day . . .]

1 Introduction

This document describes in detail VGAM family functions for modeling frequencies of genetic data. In particular, we look at gene combinations in the field of population genetics. There are many general references with statistical aspects; these include Elandt-Johnson (1971), Weir (1996), Lange (2002), Liu (1998). In this article we draw heavily on these. Other references include Ott (1999) and Thompson (2000). Many of VGAM's features come from `glm()` and `gam()` so that readers unfamiliar with these functions are referred to Chambers and Hastie (1993). Additionally, the VGAM *User Manual* should be consulted for general instructions about the software.

Table 1: *Models currently implemented by* VGAM *and their unique parameters.*

Family function name	Independent Parameters
`ABO`	p, q
`MNSs`	m_S, m_s, n_S
`AB.Ab.aB.ab`	p
`AB.Ab.aB.ab2`	p
`AA.Aa.aa`	p_A
`AAaa.nohw()`	p_A, f
`G1G2G3`	p_1, p_2, f

It should be noted that Fisher scoring is preferred to Newton-Raphson because it often converges in a larger parameter space, i.e., initial values don't have to be so close to the solution. All family functions in this article are based on a multinomial log-likelihood. With J classes, say,

$$\ell = \sum_{i=1}^{n} \sum_{j=1}^{J} y_{ij} \log p_{ij}$$

where y_{ij} are counts ($\sum_{j=1}^{J} y_{ij} = n_i$). For genetic family functions the input is often a 1 row matrix, so that $n = 1$. It may easily be shown that the score vector is

$$\frac{\partial \ell}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{n} \sum_{j=1}^{J} \frac{y_{ij}}{p_{ij}} \frac{\partial p_{ij}}{\partial \boldsymbol{\theta}}$$

and the expected information matrix is

$$\mathcal{I}_E(\boldsymbol{\theta}) \equiv -E\left[\frac{\partial^2 \ell}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}^T}\right] = \sum_{i=1}^{n} \sum_{j=1}^{J} \frac{1}{p_{ij}} \frac{\partial p_{ij}}{\partial \boldsymbol{\theta}} \frac{\partial p_{ij}}{\partial \boldsymbol{\theta}^T}.$$

Although these formulae can be used for *all* family functions, it is used for *most* of the family functions implemented by VGAM. A summary of all family functions currently implemented is in Table 1. In the next section we discuss each model in a little detail.

2 Models

2.1 The ABO Blood Group System

Table 2: *Probability table for the ABO blood group system. Nb.* $r = 1 - p - q$.

Genotype	AA	AO	BB	BO	AB	OO
Probability	p^2	$2pr$	q^2	$2qr$	$2pq$	r^2
Blood group	A	A	B	B	AB	O

In this example, the blood groups A, B and O form six possible combinations (genotypes) consisting of AA, AO, BB, BO, AB, OO (see Table 2). A and B are dominant over bloodtype O. Let p, q and r be the probabilities for A, B and O respectively (so that $p + q + r = 1$) for a given population. Our aim is to estimate p, q and r. The log-likelihood function is

$$\ell(p, q) \;=\; n_A \log(p^2 + 2pr) + n_B \log(q^2 + 2qr) + n_{AB} \log(2pq) + 2n_O \log(1 - p - q),$$

where $r = 1 - p - q$, $p \in (0, 1)$, $q \in (0, 1)$, $p + q < 1$. We let $\boldsymbol{\eta} = (g(p), g(r))^T$ where g is the link function. The four column input have columns corresponding to A-B-AB-O, respectively, and the family function is `ABO()`. For more information about the ABO blood group system see Chapter 14 of Elandt-Johnson (1971).

2.2 The MNS Blood Group System

2.2.1 The MNSs Blood Group System

This is described in Sections 14.13 and 14.14 of Elandt-Johnson (1971), and we will use the notation used there. There are three independent parameters: m_S, m_s, n_S, say, so that $n_s = 1 - m_S - m_s - n_S$. We let $\boldsymbol{\eta} = (g(m_S), g(m_s), g(n_S))^T$ where g is the link function.

Table 3: *Probability table for the combinations of MNSs blood group system.*

Genotype	MS	Ms	MNS	MNs	NS	Ns
Probability	$m_S^2 + 2m_S m_s$	m_s^2	$2(m_S m_S + m_s n_S + m_S n_s)$	$2m_s n_s$	$n_S^2 + 2n_S n_s$	n_s^2
Blood group	(MS)	(Ms)	(MNS)	(MNs)	(NS)	(Ns)

The six column input have columns corresponding to MS-Ms-MNS-MNs-NS-Ns, respectively. The family function `MNSs()` implements the expected information matrix, and is easily computed using Table 14.4 of Elandt-Johnson (1971) (which contains three wrong entries).

2.3 The `AB.Ab.aB.ab` Blood Group

This one parameter model is given in Table 4. One has

$$\ell(p) \;=\; n_{AB}\,\log((2+p^2)/4) + (n_{Ab}+n_{aB})\,\log((1-p^2)/4) + n_{ab}\,\log(p^2/4)$$

and

$$\begin{aligned}
\frac{\partial \ell}{\partial p} &= 8p\left\{\frac{n_{AB}}{2+p^2} - \frac{n_{Ab}+n_{aB}}{1-p^2} + \frac{n_{ab}}{p^2}\right\},\\
-\frac{\partial^2 \ell}{\partial p^2} &= -8\left\{\frac{n_{AB}}{2+p^2} - \frac{n_{Ab}+n_{aB}}{1-p^2} + \frac{n_{ab}}{p^2}\right\} + 16p^2\left\{\frac{n_{AB}}{(2+p^2)^2} + \frac{n_{Ab}+n_{aB}}{(1-p^2)^2} + \frac{n_{ab}}{p^4}\right\},\\
-E\frac{\partial^2 \ell}{\partial p^{\,2}} &= 4np^2\left\{\frac{1}{2+p^2} + \frac{2}{1-p^2} + \frac{1}{p^2}\right\}.
\end{aligned}$$

Page 236 of Weir (1996) parametrize this using $\theta = p^2$.

2.3.1 The `AB.Ab.aB.ab2` Blood Group

This is very strongly related to the above! The log-likelihood function is

$$\ell(p) \;=\; n_{AB}\,\log((p^2-2p+3)/4) + n_{Ab}\,\log((-p^2+2p)/4) + n_{aB}\,\log((-p^2+2p)/4) + n_{ab}\,\log((p^2-2p+1)/4).$$

This is equivalent to Table 15.4 of Elandt-Johnson (1971) (which is reproduced in Table 5) who use this model to estimate linkage from intercrosses in coupling. That book defines λ as the *recombination fraction*, and has a range from 0 to $\frac{1}{2}$.

This is related to double-intercross—see Page 44 of Lange (2002),

Table 4: *Probability table for the combinations of A and B bloodgroups.*

F2	phenotypes	F2	genotypes from F3 progeny
AB	$(2+p^2)/4$	AABB	$p^2/4$
		AABb	$p(1-p)/2$
		AaBB	$p(1-p)/2$
		AaBb	$(1-2p(1-p))/2$
Ab	$(1-p^2)/4$	AAbb	$(1-p)^2/4$
		Aabb	$p(1-p)/2$
aB	$(1-p^2)/4$	aaBB	$(1-p)^2/4$
		aaBb	$p(1-p)/2$
ab	$p^2/4$	aabb	$p^2/4$

Table 5: *Table 15.4 of Elandt-Johnson (1971).*

Class (Phenotype in F_2)	$A-B-$	$A-bb$	$aaB-$	$aabb$
Class frequency	r_1	r_2	r_3	r_4
Probability	$\frac{1}{4}[2+(1-\lambda)^2]$	$\frac{1}{4}[1-(1-\lambda)^2]$	$\frac{1}{4}[1-(1-\lambda)^2]$	$\frac{1}{4}(1-\lambda)^2$

2.4 The `AA.Aa.aa` Blood Group

Table 6 gives the formulae for this blood group. For this one parameter model, one has

$$\ell(p_A) \;=\; (2n_{AA}+n_{Aa})\,\log(p_A) + (n_{Aa}+2n_{aa})\,\log(1-p_A)$$

and

$$\begin{aligned}
\frac{\partial \ell}{\partial p_A} &= \frac{2n_{AA}+n_{Aa}}{p_A} - \frac{n_{Aa}+2n_{aa}}{1-p_A},\\
-\frac{\partial^2 \ell}{\partial p_A^2} &= \frac{2n_{AA}+n_{Aa}}{p_A^2} - \frac{n_{Aa}+2n_{aa}}{(1-p_A)^2}.
\end{aligned}$$

This has been implemented (using Fisher scoring) in the VGAM family function `AA.Aa.aa()`. See pp.56–58 of Weir (1996) and Elandt-Johnson (1971) for more details.

Table 6: *Probability table for the AA-Aa-aa blood group.*

Blood group/genotype	AA	Aa	aa
Probability	p_A^2	$2p_A(1-p_A)$	$(1-p_A)^2$

2.4.1 Without Hardy-Weinberg Equilibrium

The above assumes Hardy-Weinberg equilibrium. Without this assumption, one has Table 7. An identity link is applied to the parameter f, and a zero value of this parameter corresponds to the Hardy-Weinberg assumption. The family function is `AAaa.nohw()`.

Table 7: *Probability table for the AA-Aa-aa blood group without the assumption of Hardy-Weinberg equilibrium.*

Blood group/genotype	AA	Aa	aa
Probability	P_A	P_{Aa}	P_{aa}
Probability	$p_A^2 + p_A(1-p_A)f$	$2p_A(1-p_A)(1-f)$	$(1-p_A)^2 + p_A(1-p_A)f$

2.5 Three Alleles

Three alleles give rise to the 6 possible genotypes in Table 8.

$$L(p_1,p_2) \propto (p_1^2)^{n_1}(2p_1p_2)^{n_2}(p_2^2)^{n_3}[2p_1(1-p_1-p_2)]^{n_4}[2p_2(1-p_1-p_2)]^{n_5}[(1-p_1-p_2)^2]^{n_6}.$$

This is described in pp.61–63 of Weir (1996), and is implemented in the family function `A1A2A3()`.

Table 8: *Frequencies for three alleles.*

A_1A_1	A_1A_2	A_2A_2	A_1A_3	A_2A_3	A_3A_3
n_1	n_2	n_3	n_4	n_5	n_6

2.6 The `G1G2G3` Family Function

This model is described in Section 3.3 of Lange (2002). Table 9 gives data cited by Yasuda (1968) from 1948 people from north east Brazil. Additionally, the probabilities are also given in terms of the three independent parameters p_1, p_2 and f (the probability $p_3 = 1 - p_1 - p_2$.) The parameter f is the inbreeding coefficient, which is the probability that the two genes of a random person at a given locus are copies of the same ancestral gene.

Table 9: *Brazilian genotypes at the Haptoglobin locus.*

Genotype	Genotype frequency	Observed number
G_1/G_1	$fp_1 + (1-f)p_1^2$	108
G_1/G_2	$2(1-f)p_1p_2$	196
G_1/G_3	$2(1-f)p_1p_3$	429
G_2/G_2	$fp_2 + (1-f)p_2^2$	143
G_2/G_3	$2(1-f)p_2p_3$	513
G_3/G_3	$fp_3 + (1-f)p_3^2$	559

2.7 The Dirichlet Distribution

We now discuss the Dirichlet distribution, a natural extension of the beta distribution, because it has applications in genetics. Suppose $\boldsymbol{Y} = (Y_1, \ldots, Y_M)^T$. We say $\boldsymbol{Y}$ has a Dirichlet distribution if $(Y_1, \ldots, Y_{M-1})^T$ has density

$$\frac{\Gamma(\alpha_+)}{\prod_{j=1}^M \Gamma(\alpha_j)} \prod_{j=1}^M y_j^{\alpha_j - 1} \tag{1}$$

where $\alpha_+ = \alpha_1 + \cdots + \alpha_M$, $\alpha_j > 0$, and the density is defined on the unit simplex

$$\Delta_M = \left\{ (y_1, \ldots, y_M)^T : y_1 > 0, \ldots, y_M > 0, \sum_{j=1}^M y_j = 1 \right\}.$$

One has $E(Y_j) = \alpha_j/\alpha_+$. The family function is `dirichlet()`, and its input is a M column matrix. The function `rdiric()`, which generates Dirichlet random variates, is based on the fact that $Y_i = X_i / \sum_{j=1}^M X_j$ where the X_j are independent gamma random variates of unit scale. This ensures $\sum_{j=1}^M Y_j = 1$ and $Y_j \geq 0$.

2.8 The Dirichlet-Multinomial Distribution

Its density is given by

$$P(Y_1 = y_1, \ldots, Y_M = y_M) = \binom{2y_*}{y_1\ y_2 \cdots y_M} \frac{\Gamma(\alpha_+)}{\Gamma(2y_* + \alpha_+)} \prod_{j=1}^M \frac{\Gamma(y_j + \alpha_j)}{\Gamma(\alpha_j)} \tag{2}$$

where $2y_* = \sum_{j=1}^{M} y_j$ and $\alpha_j > 0$. The motivation for this distribution is a Bayesian one—see Page 86 of Weir (1996) and Section 3.7 of Lange (2002). The posterior mean is

$$E(Y_j) \;=\; \frac{y_j + \alpha_j}{2y_* + \alpha_+}.$$

Note that y_j must be a non-negative integer, and each row of the matrix response in the call to `vglm()` corresponds to transpose of the M-vector $\boldsymbol{y}_i$.

Table 10 gives data from Edwards et al. (1992). The family function `dirmul.old()` returns these *posterior* means as $\mu_{ij} = (y_{ij} + \alpha_j)/(2y_{i*} + \alpha_+)$ where the i subscript has been added to represent the ith row.

3 Other Topics

3.1 Input

The response in `vglm()`/`vgam()` for most VGAM family functions is of the form

```
vglm(y ~ x, ABO, weights=w)
```

where `weights` may be optional. Here, `y` may be one of following types:

1. a matrix of sample proportions, where each row adds to unity, and `weights` is a vector containing the n_i,

2. a matrix of counts. In this case, `weights` is usually unnecessary.

The second type is converted into the first type. The columns of `y` must match the family function, e.g., for `ABO` the second column corresponds to B etc. Often, the matrix contains 3, 4 or 6 columns.

In all VGAM family functions, parameters which are probabilities are given the choice of the logit, probit, cloglog and identity links. These are $\eta = \log(p/(1-p))$, $\Phi(p)$, $\log(\log(1-p))$ and $\eta = p$ respectively. It is not possible to choose a different link for different η_j, however, it would be quite easy to modify the family functions to implement this.

3.2 Output

Suppose `fit` is a genetic VGAM object. Then the fitted values (in `fitted(fit)`) are a $n \times 4$ matrix of probabilities (whose rows sum to unity) and `fit@prior.weights` (better, `weights(fit)`) contain the $n_i = \sum_{j=1}^{M=1} y_{ij}$.

Table 10: *Allele counts in four subpopulations.*

Allele:	5	6	7	8	9	10	11	12	Total $2y$
White	2	84	59	41	53	131	2	0	372
Black	0	50	137	78	54	51	0	0	370
Chicano	0	80	128	26	55	95	0	0	384
Asian	0	16	40	8	68	14	7	1	154

3.3 Convergence

It is always advisable to try several starting values because this will safeguard against problems of nonconvergence, or convergence to solutions other than the MLE. All family functions allow this, e.g., `ABO(..., init.r=NULL, init.p=NULL)`, in which `init.r` can be assigned the initial value(s) of r etc. By default, the family function tries some reasonable starting value(s).

3.4 Implementation Details

Family functions for the models in Table 1 use the expected Hessian, which is better behaved than the observed Hessian (it is more likely to be positive-definite).

4 Tutorial Examples

In this section we illustrate some of the models.

4.1 The ABO Blood Group

```
> y = cbind(150, 29, 6, 160)
> fit = vglm(y ~ 1, ABO, trace = TRUE)

VGLM    linear loop  1 :  loglikelihood = -344.9612
VGLM    linear loop  2 :  loglikelihood = -344.9601
VGLM    linear loop  3 :  loglikelihood = -344.9601

> fit = vglm(y ~ 1, ABO(link = identity), trace = TRUE, crit = "coef")

VGLM    linear loop  1 :  coefficients = 0.2607695010, 0.0522626446
VGLM    linear loop  2 :  coefficients = 0.2607647692, 0.0522648283
VGLM    linear loop  3 :  coefficients = 0.2607647548, 0.0522648349
VGLM    linear loop  4 :  coefficients = 0.2607647548, 0.0522648349

> Coef(fit)

         p          q
0.26076475 0.05226483

> rbind(y, sum(y) * fitted(fit))

         A        B      AB        O
  150.0000 29.00000 6.00000 160.0000
1 147.0644 25.71644 9.40389 162.8153
```

The latter shows the actual and fitted values are very similar, as expected.

Here is another example, mimicing the results of pp.401–2 of Elandt-Johnson (1971). It involves the northeastern Brazilian population.

```
> y = cbind(A = 725, B = 258, AB = 72, O = 1073)
> fit = vglm(y ~ 1, ABO(link = identity), trace = TRUE, crit = "coef")
```

```
VGLM    linear loop  1 :  coefficients = 0.2091306527, 0.0808010076
VGLM    linear loop  2 :  coefficients = 0.2091306545, 0.0808010082

> Coef(fit)

         p          q
0.20913065 0.08080101

> rbind(y, sum(y) * fitted(fit))

         A        B       AB        O
  725.0000 258.0000 72.00000 1073.000
1 725.0729 258.0780 71.91775 1072.931
```

4.2 The AB.Ab.aB.ab Blood Group

```
> y = cbind(1997, 906, 904, 32)
> fit = vglm(y ~ 1, AB.Ab.aB.ab(link = identity, init.p = 0.9),
+     trace = TRUE)

VGLM    linear loop  1 :  loglikelihood = -4241.948
VGLM    linear loop  2 :  loglikelihood = -4092.555
VGLM    linear loop  3 :  loglikelihood = -4075.751
VGLM    linear loop  4 :  loglikelihood = -4074.890
VGLM    linear loop  5 :  loglikelihood = -4074.879
VGLM    linear loop  6 :  loglikelihood = -4074.879

> Coef(fit)

        p
0.1889867

> rbind(y, sum(y) * fitted(fit))

        AB       Ab       aB       ab
  1997.000 906.0000 904.0000 32.00000
1 1953.778 925.4716 925.4716 34.27841
```

The latter shows the actual and fitted values are very similar, as expected.

4.3 The Dirichlet-Multinomial Distribution

Here we fit a Dirichlet-multinomial distribution to the data of Edwards et al. (1992) given in Table 10.

```
> set.seed(123)
> y = c(2, 84, 59, 41, 53, 131, 2, 0, 0, 50, 137, 78, 54,
+     51, 0, 0, 0, 80, 128, 26, 55, 95, 0, 0, 0, 16, 40,
+     8, 68, 14, 7, 1)
> dim(y) = c(8, 4)
> y = t(y)
> fit = vglm(y ~ 1, dirmul.old)
> round(Coef(fit), dig = 2)
```

```
shape1 shape2 shape3 shape4 shape5 shape6 shape7 shape8
  0.11   4.64   7.33   2.97   5.32   5.26   0.27   0.10
```

```
> round(t(fitted(fit)), dig = 4)
```

```
          1      2      3      4
[1,] 0.0053 0.0003 0.0003 0.0006
[2,] 0.2227 0.1380 0.2064 0.1147
[3,] 0.1667 0.3645 0.3301 0.2630
[4,] 0.1105 0.2045 0.0707 0.0609
[5,] 0.1465 0.1498 0.1471 0.4073
[6,] 0.3424 0.1421 0.2445 0.1070
[7,] 0.0057 0.0007 0.0007 0.0404
[8,] 0.0002 0.0002 0.0002 0.0061
```

Evidently, convergence of the fitted model is slow because of the existence of zero counts. The second row of Table 3.5 of Lange (2002) are the posterior mean estimates.

Question: are all the α_j's the same? We can test this as follows.

```
> pfit = vglm(y ~ 1, dirmul.old(parallel = TRUE))
> round(Coef(pfit), dig = 2)
```

```
(Intercept)
      -1.12
```

```
> 1 - pchisq(2 * (logLik(fit) - logLik(pfit)), df = pfit@df.residual -
+     fit@df.residual)
```

```
[1] 1.003194e-09
```

The common estimate is $\hat{\alpha} = 0.33$, however, a likelihood ratio test shows that there is very strong evidence against the equal α_j assumption.

5 Discussion

As well as estimation of the parameters an important problem is goodness of fit tests. This will be implemented later.

Exercises

1. For the k-allele case $(k > 2)$ one has

$$\begin{aligned} P_{uu} &= p_u^2 + p_u(1-p_u)f, \quad u = 1, 2, \ldots, k, \\ P_{uv} &= 2p_u p_v(1-f), \quad u, v = 1, 2, \ldots, k; u \neq v. \end{aligned}$$

   There are a total of k independent parameters, f and $k-1$ allele frequencies. Write a VGAM family function to implement the $k = 3$ and 4 case each.

2. Maximum likelihood estimation in classical segregation analysis involves maximizing the likelihood
$$\prod_k \binom{s_k}{r_k} p^{r_k}(1-p)^{s_k-r_k}\pi^{a_k}(1-\pi)^{r_k-a_k}/\left[1-(1-p\pi)^{s_k}\right]$$
where k indexes over ascertained families, the kth ascertained family has s_k siblings of whom r_k are affected and a_k are ascertained. Write a VGAM family function for this to estimate p and π. Call it `segregation()`. Apply it to the data in Table 12 to obtain $\hat{p} = 0.2679$ and $\hat{\pi} = 0.3594$. What are the standard errors?

3. Consider a Reed-Frost transmission chain model with a household of size 4, say. Let θ be the probability of avoiding infection by one infective. The probability distribution for the outbreak size is given in Table 11. The Reed-Frost assumption means the probability of avoiding infection when exposed to two infectives is θ^2. Write a VGAM family function (call it `reedfrost4()`) to implement this model. Assume the input is a 4-column matrix of counts with Size $= j$ being the jth column, and the log-likelihood should be maximized. Allow for several link functions for θ. The fitted values can be either the estimate of the expected number of within household infections or the proportion in each group.

Acknowledgements

This document and VGAM family functions were initially started by Thiha Thwin, a project student at the University of Auckland.

References

Chambers, J. M., Hastie, T. J. (Eds.), 1993. Statistical Models in S. Chapman & Hall, New York.

Edwards, A., Hammond, H. A., Jin, L., Caskey, C. T., Chakraborty, R., 1992. Genetic variation at five trimeric and tetrameric tandem repeat loci in four human population groups. Genomics 12, 241–253.

Elandt-Johnson, R. C., 1971. Probability Models and Statistical Methods in Genetics. Wiley, New York.

Lange, K., 2002. Mathematical and Statistical Methods for Genetic Analysis, 2nd Edition. Springer-Verlag, New York.

Liu, B.-H., 1998. Statistical Genomics: Linkage, Mapping, and QTL Analysis. CRC Press, Boca Raton, FL.

Table 11: *Probability distribution for a Reed-Frost transmission chain model with a household size of 4.*

Size	Probability
1	θ^3
2	$3\theta^4(1-\theta)$
3	$3\theta^3(1-\theta)^2(1+2\theta)$
4	Balance of probability

Ott, J., 1999. Analysis of Human Genetic Linkage, 3rd Edition. The Johns Hopkins University Press, Baltimore, MD.

Thompson, E. A., 2000. Statistical Inference from Genetic Data on Pedigrees. Institute of Mathematical Statistics, Beachwood, OH.

Weir, B. S., 1996. Genetic Data Analysis II: Methods for Discrete Population Genetic Data. Sinauer Associates, Inc., Sunderland, MA.

Yasuda, N., 1968. Estimation of the interbreeding coefficient from phenotype frequencies by a method of maximum likelihood scoring. Biometrics 24, 915–934.

Table 12: *Cystic Fibrosis data. Source: Crow (1965).*

Siblings s	Affected r	Ascertained a	Families n
10	3	1	1
9	3	1	1
8	4	1	1
7	3	2	1
7	3	1	1
7	2	1	1
7	1	1	1
6	2	1	1
6	1	1	1
5	3	3	1
5	3	2	1
5	2	1	5
5	1	1	2
4	3	2	1
4	3	1	2
4	2	1	4
4	1	1	6
3	2	2	3
3	2	1	3
3	1	1	10
2	2	2	2
2	2	1	4
2	1	1	18
1	1	1	9