

# When is Deviance Goodness of Fit – and When It's Not!

G. Monette

March 17, 2020

## Contents

Simulation	11
Method 1:	15

Exercise . . . . .	30
<b>Leave-one-out (LOO) cross-validation</b>	<b>30</b>
<b>Aggregated with frequencies as weights</b>	<b>33</b>
<b>Binomial model with logit link</b>	<b>45</b>
Comparisons . . . . .	50
Exercise: . . . . .	67

In discussions a few days ago it became apparent that there might be confusion over the use of residual deviance as a measure of goodness of fit (GOF).

Consider the following logistic regression predicting whether a family in Bangladesh switched the well from which it obtained water as a function of the amount of arsenic in its original well:

```
library(car)
```

```
Loading required package: carData
```

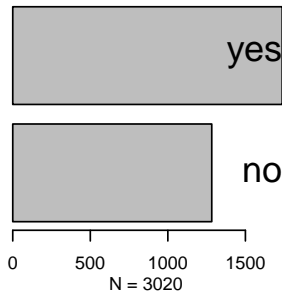
```
library(spida2)
```

```
head(Wells)
```

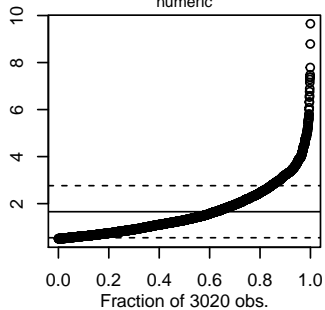
	switch	arsenic	distance	education	association
1	yes	2.36	16.826	0	no
2	yes	0.71	47.322	0	no
3	no	2.07	20.967	10	no
4	yes	1.15	21.486	12	no
5	yes	1.10	40.874	14	yes
6	yes	3.90	69.518	9	yes

```
xqplot(Wells)
```

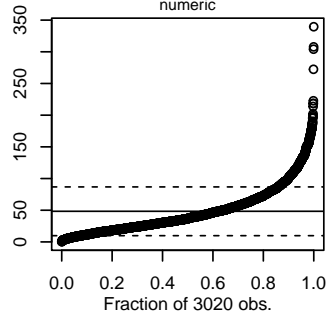
switch  
factor



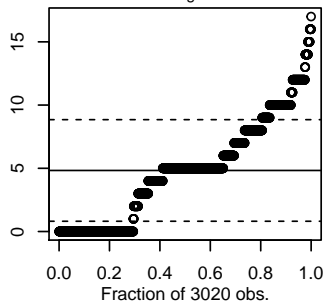
arsenic  
numeric



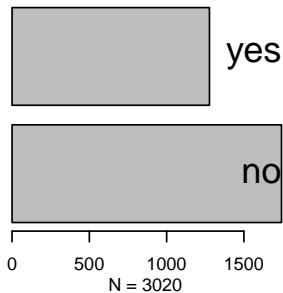
distance  
numeric



education  
integer



association  
factor



```
fit <- glm(switch ~ arsenic, Wells, family = binomial)
summary(fit)
```

Call:

```
glm(formula = switch ~ arsenic, family = binomial, data = Wells)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.331	-1.201	0.822	1.097	1.225

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-0.30553	0.07030	-4.346	1.39e-05	***
arsenic	0.37914	0.03853	9.840	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 4118.1 on 3019 degrees of freedom  
Residual deviance: 4008.7 on 3018 degrees of freedom  
AIC: 4012.7

Number of Fisher Scoring iterations: 4

Anova(fit)

Analysis of Deviance Table (Type II tests)

Response: switch

	LR	Chisq	Df	Pr(>Chisq)
arsenic	109.43	1	< 2.2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
AIC(fit)
```

```
[1] 4012.673
```

```
logLik(fit)
```

```
'log Lik.' -2004.337 (df=2)
```

It's tempting to look at the residual deviance of 4008.6 on 3018 degrees of freedom as an indicator of the goodness of fit of the model but the fact is that a logistic model can give a 'perfect' fit, yet still have a very large residual deviance as we show below.

In general, the residual deviance of a particular model has no meaning in itself. It can only be used in comparisons with other models using exactly the same data. There is one exception: when the data can be modeled with a **saturated model** that fits the data perfectly and has a deviance of zero.

Specifically, the meaningful uses of deviance are to:

- perform a likelihood-ratio-test of a hypothetical null model against a

larger alternative model fitting the same data whose parameter space includes the null model parameter space as a subset. You perform the test by subtracting the deviance of the larger model from that of the smaller model and comparing the result with a  $\chi^2$  distribution with appropriate degrees of freedom.

- use the deviance to compute the AIC or the BIC for the model, which have no interpretation in themselves but can be used for comparison with AICs or BICs of other models that have been fitted on exactly the same data. (Beware of missing data in the predictors used in one model but not the other which can result in models being fitted on different subsets of the data.)

In both of these examples the individual model deviances need not have any interpretation in themselves, only when subtracted from other deviances.

There is one situation in which the residual deviance does have a useful interpretation. That is when the formulation and parametrization of the model allows for a **saturated model** that gives a *perfect fit*. The residual deviance for the saturated model is 0 and the residual deviances of smaller models are actually comparisons of the smaller model with the saturated model.



Thus, the residual deviances don't have a meaning in themselves. They only have a meaning because they happen to provide comparisons with the largest feasible model.

Although logistic regression with a 0/1 response cannot be formulated directly in a form that has a saturated model, it is possible to reshape the data so there is a saturated model **provided all predictors are categorical**.

A logistic regression on categorical predictors can be formulated in four ways:

- the usual logistic regression with a 0/1 response.
- aggregated data with frequencies for all combinations of levels of the response and of the predictors using a weighted logistic regression.
- aggregated data by combinations of levels of the predictors with a matrix of frequencies of 'successes' and 'failures' used in a binomial regression with a *logit* link.
- a special log-linear model saturated on the predictors but with varying dependencies between the response category and the predictors categories.

The last two have full models that are saturated models and the residual deviances of non-full models provided comparisons with the saturated models.

However, the residual deviances for the first two forms cannot be interpreted except in comparison with other models.

The examples below show how to simulate logistic data with categorical predictors and how to transform data from one form to another. They show how differences in residual deviance are consistent between data representations but the absolute values only have clear interpretations in cases with saturated models.

```
library(spida2)
library(car)
library(lattice)
library(latticeExtra)
library(boot)
```

```
Attaching package: 'boot'
```

```
The following object is masked from 'package:lattice':
```

melanoma

The following object is masked from 'package:car':

logit

## Simulation

Simulating logistic data with categorical predictors

```
sim <- function(n = 100, p = 0.5, na = 2, nb = 2) {  
  df <- expand.grid(  
    a = paste0('a', seq_len(na)),  
    b = paste0('b', seq_len(nb)))  
  df <- within(df,  
    {  
      p <- rep(p, length.out = nrow(df))  
      n <- rep(n, length.out = nrow(df))  
    })  
}
```

```
df <- df[rep(1:nrow(df), df$n),]  
df$y <- rbinom(nrow(df), 1, df$p)  
df  
}  
set.seed(1273)  
dd <- sim(p = c(.2, .6, .6, .1))  
head(dd)
```

	a	b	n	p	y
1	a1	b1	100	0.2	0
1.1	a1	b1	100	0.2	0
1.2	a1	b1	100	0.2	0
1.3	a1	b1	100	0.2	1
1.4	a1	b1	100	0.2	0
1.5	a1	b1	100	0.2	1

```
tab(dd, ~ y + a + b)
```

```
, , b = b1
```

	a		
y	a1	a2	Total
0	83	39	122
1	17	61	78
Total	100	100	200

, , b = b2

	a		
y	a1	a2	Total
0	44	86	130
1	56	14	70
Total	100	100	200

, , b = Total

	a		
y	a1	a2	Total

0	127	125	252
1	73	75	148
Total	200	200	400

```
tab(dd, ~ y + a + b, pct = c(2,3))
```

```
, , b = b1
```

	a		
y	a1	a2	All
0	83.0	39.0	61.0
1	17.0	61.0	39.0
Total	100.0	100.0	100.0

```
, , b = b2
```

	a		
y	a1	a2	All
0	44.0	86.0	65.0

1	56.0	14.0	35.0
Total	100.0	100.0	100.0

, , b = All

	a		
y	a1	a2	All
0	63.5	62.5	63.0
1	36.5	37.5	37.0
Total	100.0	100.0	100.0

## Method 1:

Regressing response as 0/1 variable

This is the only way available if there is even a single continuous predictor with distinct values since you can't aggregate over rows.

```
fit0 <- glm(y ~ 1, dd, family = binomial)
fita <- glm(y ~ a + b, dd, family = binomial) # additive model
fiti <- glm(y ~ a * b, dd, family = binomial) # interaction model
summary(fit0)
```

Call:

```
glm(formula = y ~ 1, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-0.9613	-0.9613	-0.9613	1.4101	1.4101

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.5322	0.1036	-5.139	2.76e-07 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1



(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 527.16 on 399 degrees of freedom  
AIC: 529.16

Number of Fisher Scoring iterations: 4

```
summary(fita)
```

Call:

```
glm(formula = y ~ a + b, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.0027	-0.9859	-0.9363	1.3628	1.4587

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.46885	0.17851	-2.626	0.00863 **
aa2	0.04298	0.20732	0.207	0.83578
bb2	-0.17175	0.20736	-0.828	0.40754

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 526.43 on 397 degrees of freedom  
AIC: 532.43

Number of Fisher Scoring iterations: 4

```
summary(fiti)
```

Call:

```
glm(formula = y ~ a * b, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.3723	-0.6105	-0.5492	0.9943	1.9830

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.5856	0.2662	-5.956	2.58e-09	***
aa2	2.0329	0.3360	6.050	1.45e-09	***
bb2	1.8268	0.3339	5.472	4.45e-08	***
aa2:bb2	-4.0894	0.4864	-8.408	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom

Residual deviance: 443.11 on 396 degrees of freedom  
AIC: 451.11

Number of Fisher Scoring iterations: 4

```
anova(fit0, fita, fiti, test = 'LRT')
```

Analysis of Deviance Table

Model 1:  $y \sim 1$

Model 2:  $y \sim a + b$

Model 3:  $y \sim a * b$

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	399	527.16			
2	397	526.43	2	0.730	0.6943
3	396	443.11	1	83.329	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
AIC(fit0, fita, fiti)
```

	df	AIC
fit0	1	529.1645
fita	3	532.4349
fiti	4	451.1055

```
BIC(fit0, fita, fiti)
```

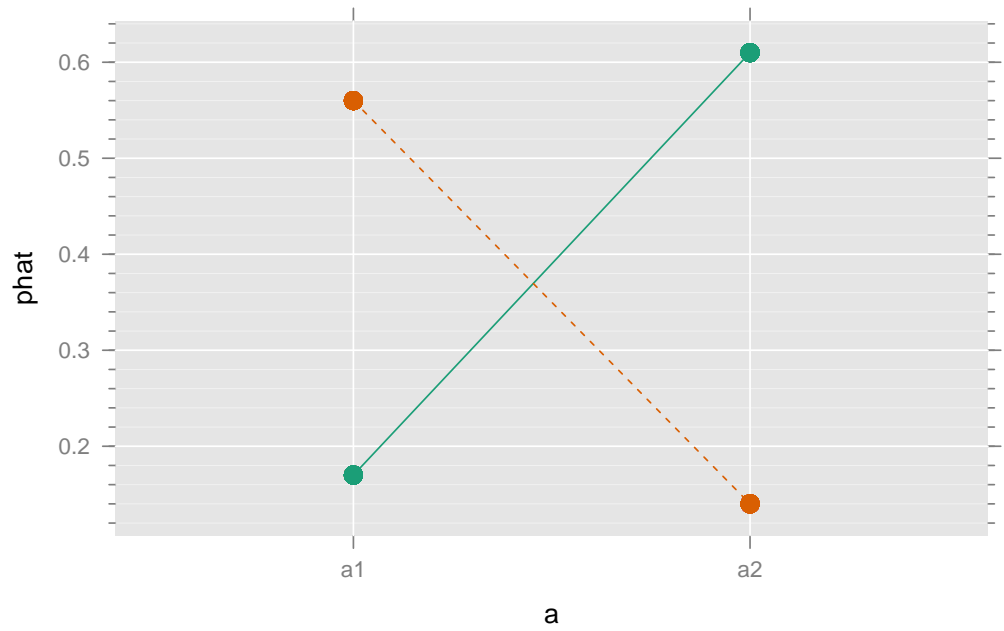
	df	BIC
fit0	1	533.1560
fita	3	544.4093
fiti	4	467.0714

Predicted probabilities and log odds

```
dd$phat <- predict(fiti, type = 'response')  
dd$lo_interaction <- predict(fiti, type = 'link')  
dd$lo_additive <- predict(fita, type = 'link')
```

```
gd(pch = 16)
xyplot(phat ~ a, dd, groups = b, type = 'b',
       auto.key = list(columns = 2, lines = T))
```

b1 ● — b2 ● - - -



Reshape data set to plot both models

```
pred <- tolong(dd, sep = '__', timevar = 'model')
some(pred)
```

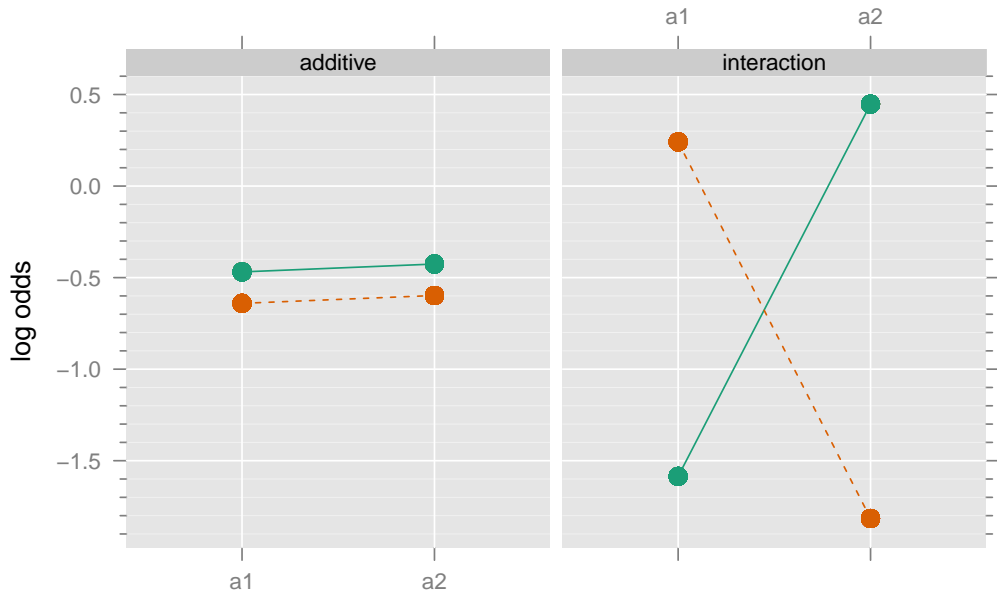
	a	b	n	p	phat	y	model	lo	id
48.additive	a1	b1	100	0.2	0.17	0	additive	-0.4688509	48
68.additive	a1	b1	100	0.2	0.17	0	additive	-0.4688509	68
176.additive	a2	b1	100	0.6	0.61	1	additive	-0.4258751	176
221.additive	a1	b2	100	0.6	0.56	1	additive	-0.6405964	221
247.additive	a1	b2	100	0.6	0.56	1	additive	-0.6405964	247
266.additive	a1	b2	100	0.6	0.56	1	additive	-0.6405964	266
271.additive	a1	b2	100	0.6	0.56	0	additive	-0.6405964	271
188.interaction	a2	b1	100	0.6	0.61	1	interaction	0.4473122	188
237.interaction	a1	b2	100	0.6	0.56	1	interaction	0.2411621	237
328.interaction	a2	b2	100	0.1	0.14	0	interaction	-1.8152900	328

```
xyplot(lo ~ a | model, pred, groups = b, type = 'b',
       ylab = 'log odds',
       auto.key = list(columns = 2, lines = T))
```



b1 ● —

b2 ● - - -



a

**Figure:** Estimated log odds for an additive model and a model with interaction.

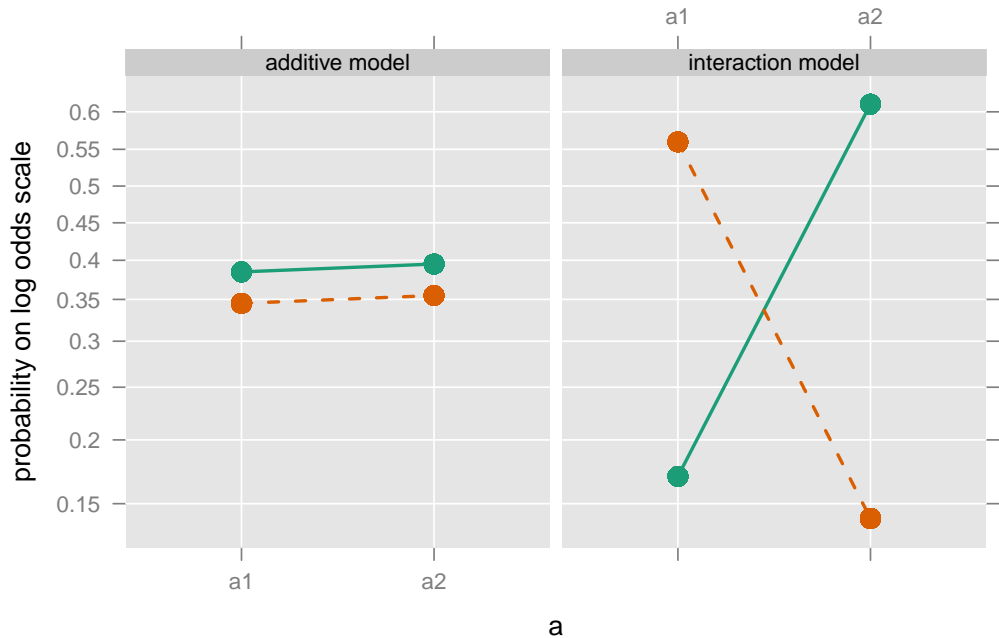
```
logit <- function(p) log(p/(1-p))  
expit <- function(lo) 1/(1 + exp(- lo))
```

Plotting probabilities on log odds scale

```
gd(lwd = 2)  
xyplot(lo ~ a | paste(model, 'model'), pred, groups = b, type = 'b',  
       ylab = 'probability on log odds scale',  
       scale = list(  
         y = list(at = logit(seq(0,1,.05)),  
                  labels = seq(0,1,.05))),  
       auto.key = list(columns = 2, lines = T))
```

b1 ● —

b2 ● - - -

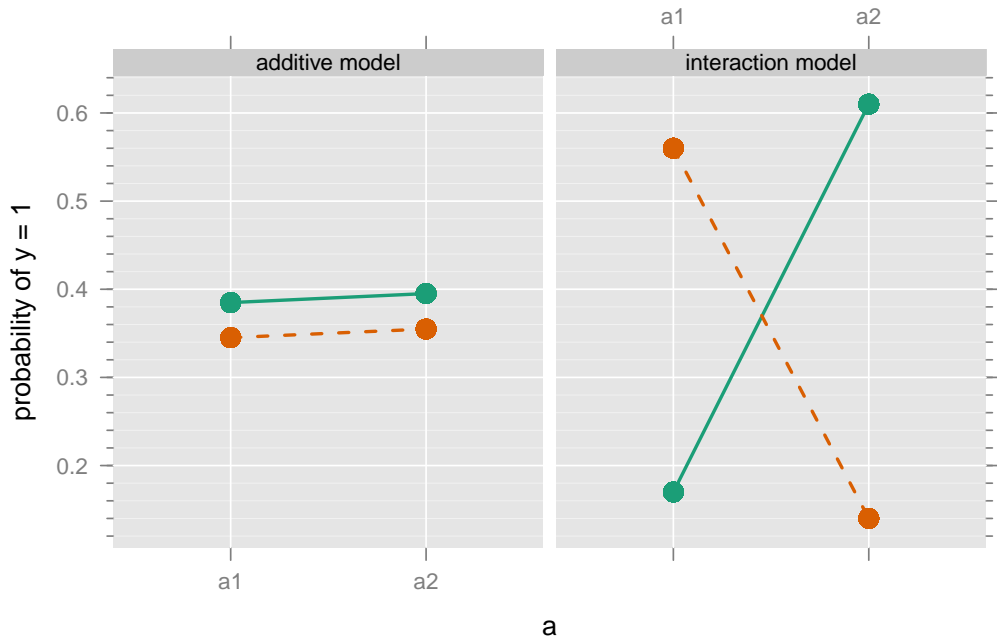


**Figure:** Estimated probability plotted on a log-odds scale for an additive model and a model with interaction.

```
xyplot(expit(lo) ~ a | paste(model, 'model'), pred, groups = b, type  
  ylab = 'probability of y = 1',  
  auto.key = list(columns = 2, lines = T))
```

b1 ● —

b2 ● - - -



**Figure:** Estimated probability for an additive model and a model with interaction.

## Exercise

What difference does it make whether you plot probabilities on a log-odds scale on a probability scale? The difference is very slight in this case.

## Leave-one-out (LOO) cross-validation

```
library(boot)
cv.glm(dd, fiti)[-4]
```

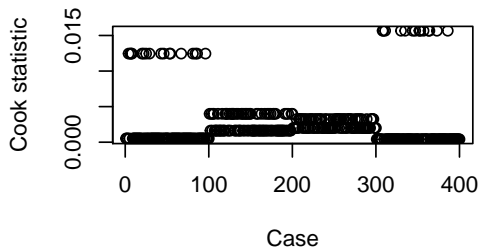
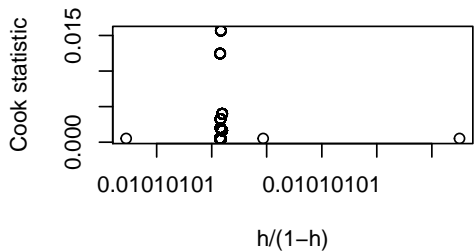
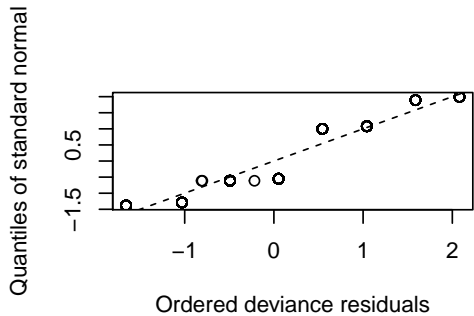
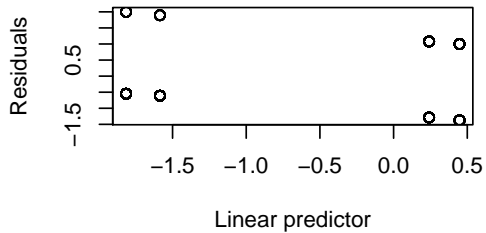
```
$call
cv.glm(data = dd, glmfit = fiti)
```

```
$K
[1] 400
```

```
$delta
```

```
[1] 0.1902357 0.1902309
```

```
glm.diag.plots(fiti)
```





# Aggregated with frequencies as weights

Aggregated data within each combination of levels of y, a and b:

```
ddag <- as.data.frame(tab__(dd, ~ y + a + b))  
ddag
```

	y	a	b	Freq
1	0	a1	b1	83
2	1	a1	b1	17
3	0	a2	b1	39
4	1	a2	b1	61
5	0	a1	b2	44
6	1	a1	b2	56
7	0	a2	b2	86
8	1	a2	b2	14

Fitting models on aggregated data

```
fitag0 <- glm(y ~ 1, ddag, family = binomial, weights = Freq)
fitaga <- glm(y ~ a + b, ddag, family = binomial, weights = Freq)
fitagi <- glm(y ~ a * b, ddag, family = binomial, weights = Freq)
summary(fitag0)
```

Call:

```
glm(formula = y ~ 1, family = binomial, data = ddag, weights =
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-8.9146	-6.9718	-0.3635	6.9988	11.0136

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.5322	0.1036	-5.139	2.76e-07 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 7 degrees of freedom  
Residual deviance: 527.16 on 7 degrees of freedom  
AIC: 529.16

Number of Fisher Scoring iterations: 4

```
summary(fit0)
```

Call:

```
glm(formula = y ~ 1, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-0.9613	-0.9613	-0.9613	1.4101	1.4101

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.5322	0.1036	-5.139	2.76e-07 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 527.16 on 399 degrees of freedom  
AIC: 529.16

Number of Fisher Scoring iterations: 4

```
summary(fitaga)
```

Call:

```
glm(formula = y ~ a + b, family = binomial, data = ddag, weight
```

Deviance Residuals:

1	2	3	4	5	6	7	8
-8.982	5.698	-6.262	10.644	-6.103	10.916	-8.683	5.386

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.46885	0.17851	-2.626	0.00863 **
aa2	0.04298	0.20732	0.207	0.83578
bb2	-0.17175	0.20736	-0.828	0.40754

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 7 degrees of freedom  
Residual deviance: 526.43 on 5 degrees of freedom  
AIC: 532.43

Number of Fisher Scoring iterations: 4

```
summary(fita)
```

Call:

```
glm(formula = y ~ a + b, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.0027	-0.9859	-0.9363	1.3628	1.4587

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.46885	0.17851	-2.626	0.00863 **
aa2	0.04298	0.20732	0.207	0.83578
bb2	-0.17175	0.20736	-0.828	0.40754
---				

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 526.43 on 397 degrees of freedom  
AIC: 532.43

Number of Fisher Scoring iterations: 4

```
summary(fitagi)
```

Call:

```
glm(formula = y ~ a * b, family = binomial, data = ddag, weight
```

Deviance Residuals:

1	2	3	4	5	6	7	8
-5.561	7.762	-8.570	7.766	-8.500	8.059	-5.093	7.420

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.5856	0.2662	-5.956	2.58e-09	***
aa2	2.0329	0.3360	6.050	1.45e-09	***
bb2	1.8268	0.3339	5.472	4.45e-08	***
aa2:bb2	-4.0894	0.4864	-8.408	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 7 degrees of freedom  
Residual deviance: 443.11 on 4 degrees of freedom  
AIC: 451.11

Number of Fisher Scoring iterations: 5

```
summary(fiti)
```



Call:

```
glm(formula = y ~ a * b, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.3723	-0.6105	-0.5492	0.9943	1.9830

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.5856	0.2662	-5.956	2.58e-09	***
aa2	2.0329	0.3360	6.050	1.45e-09	***
bb2	1.8268	0.3339	5.472	4.45e-08	***
aa2:bb2	-4.0894	0.4864	-8.408	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 443.11 on 396 degrees of freedom  
AIC: 451.11

Number of Fisher Scoring iterations: 4

Compare Likelihood ratio tests

```
anova(fitag0, fitaga, fitagi, test = 'LRT')
```

Analysis of Deviance Table

Model 1: y ~ 1

Model 2: y ~ a + b

Model 3: y ~ a \* b

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	7	527.16			
2	5	526.43	2	0.730	0.6943
3	4	443.11	1	83.329	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
anova(fit0, fita, fiti, test = 'LRT')
```

Analysis of Deviance Table

Model 1: y ~ 1

Model 2: y ~ a + b

Model 3: y ~ a \* b

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	399	527.16			
2	397	526.43	2	0.730	0.6943
3	396	443.11	1	83.329	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
AIC(fitag0, fitaga, fitagi)
```

	df	AIC
fitag0	1	529.1645

```
fitaga 3 532.4349
fitagi 4 451.1055
```

```
AIC(fit0, fita, fiti)
```

```
      df      AIC
fit0  1 529.1645
fita  3 532.4349
fiti  4 451.1055
```

Note how BIC use the 'wrong' n:

```
BIC(fitag0, fitaga, fitagi)
```

```
      df      BIC
fitag0 1 529.2440
fitaga 3 532.6732
fitagi 4 451.4233
```

```
BIC(fit0, fita, fiti)
```

	df	BIC
fit0	1	533.1560
fita	3	544.4093
fiti	4	467.0714

## Binomial model with logit link

```
ddg <- as.data.frame(tab__(dd, ~ y + a + b))  
ddg
```

	y	a	b	Freq
1	0	a1	b1	83
2	1	a1	b1	17
3	0	a2	b1	39
4	1	a2	b1	61
5	0	a1	b2	44

```
6 1 a1 b2    56
7 0 a2 b2    86
8 1 a2 b2    14
```

```
ddw <- towide(ddg, timevar = 'y', idvar = c('a','b'))
ddw$ymat <- with(ddw, cbind(Freq_0, Freq_1))

fitb0 <- glm(ymat ~ 1, ddw, family = binomial)
fitba <- glm(ymat ~ a + b, ddw, family = binomial)
fitbi <- glm(ymat ~ a * b, ddw, family = binomial)
summary(fitb0)
```

Call:

```
glm(formula = ymat ~ 1, family = binomial, data = ddw)
```

Deviance Residuals:

```
      1      2      3      4
4.396 -3.851 -4.857  5.130
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	0.5322	0.1036	5.139	2.76e-07 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 84.059 on 3 degrees of freedom  
Residual deviance: 84.059 on 3 degrees of freedom  
AIC: 104.95

Number of Fisher Scoring iterations: 4

```
summary(fitba)
```

Call:

```
glm(formula = ymat ~ a + b, family = binomial, data = ddw)
```

```
Deviance Residuals:
```

1	2	3	4
4.686	-4.384	-4.330	4.838

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	0.46885	0.17851	2.626	0.00863 **
aa2	-0.04298	0.20732	-0.207	0.83578
bb2	0.17175	0.20736	0.828	0.40754

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)
```

```
Null deviance: 84.059  on 3  degrees of freedom  
Residual deviance: 83.329  on 1  degrees of freedom  
AIC: 108.22
```



Number of Fisher Scoring iterations: 4

```
summary(fitbi)
```

Call:

```
glm(formula = ymat ~ a * b, family = binomial, data = ddw)
```

Deviance Residuals:

```
[1] 0 0 0 0
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	1.5856	0.2662	5.956	2.58e-09	***
aa2	-2.0329	0.3360	-6.050	1.45e-09	***
bb2	-1.8268	0.3339	-5.472	4.45e-08	***
aa2:bb2	4.0894	0.4864	8.408	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 8.4059e+01 on 3 degrees of freedom  
Residual deviance: -1.4211e-14 on 0 degrees of freedom  
AIC: 26.893

Number of Fisher Scoring iterations: 3

## Comparisons

Null models

```
summary(fitb0)
```

Call:

```
glm(formula = ymat ~ 1, family = binomial, data = ddw)
```

Deviance Residuals:

1	2	3	4
4.396	-3.851	-4.857	5.130

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	0.5322	0.1036	5.139	2.76e-07 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 84.059 on 3 degrees of freedom  
Residual deviance: 84.059 on 3 degrees of freedom  
AIC: 104.95

Number of Fisher Scoring iterations: 4

```
summary(fitag0)
```

```
Call:
```

```
glm(formula = y ~ 1, family = binomial, data = ddag, weights =
```

```
Deviance Residuals:
```

Min	1Q	Median	3Q	Max
-8.9146	-6.9718	-0.3635	6.9988	11.0136

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.5322	0.1036	-5.139	2.76e-07 ***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)
```

```
Null deviance: 527.16 on 7 degrees of freedom
Residual deviance: 527.16 on 7 degrees of freedom
AIC: 529.16
```

```
Number of Fisher Scoring iterations: 4
```

```
summary(fit0)
```

```
Call:
```

```
glm(formula = y ~ 1, family = binomial, data = dd)
```

```
Deviance Residuals:
```

Min	1Q	Median	3Q	Max
-0.9613	-0.9613	-0.9613	1.4101	1.4101

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.5322	0.1036	-5.139	2.76e-07 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 527.16 on 399 degrees of freedom  
AIC: 529.16

Number of Fisher Scoring iterations: 4

Additive models

```
summary(fitba)
```

Call:

```
glm(formula = ymat ~ a + b, family = binomial, data = ddw)
```

Deviance Residuals:

1	2	3	4
4.686	-4.384	-4.330	4.838

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	0.46885	0.17851	2.626	0.00863 **
aa2	-0.04298	0.20732	-0.207	0.83578
bb2	0.17175	0.20736	0.828	0.40754

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 84.059 on 3 degrees of freedom  
Residual deviance: 83.329 on 1 degrees of freedom  
AIC: 108.22

Number of Fisher Scoring iterations: 4

```
summary(fitaga)
```

```
Call:
```

```
glm(formula = y ~ a + b, family = binomial, data = ddag, weight
```

```
Deviance Residuals:
```

1	2	3	4	5	6	7	8
-8.982	5.698	-6.262	10.644	-6.103	10.916	-8.683	5.386

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.46885	0.17851	-2.626	0.00863 **
aa2	0.04298	0.20732	0.207	0.83578
bb2	-0.17175	0.20736	-0.828	0.40754

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 7 degrees of freedom  
Residual deviance: 526.43 on 5 degrees of freedom  
AIC: 532.43

Number of Fisher Scoring iterations: 4

```
summary(fita)
```

Call:

```
glm(formula = y ~ a + b, family = binomial, data = dd)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.0027	-0.9859	-0.9363	1.3628	1.4587

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.46885	0.17851	-2.626	0.00863 **
aa2	0.04298	0.20732	0.207	0.83578
bb2	-0.17175	0.20736	-0.828	0.40754

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 526.43 on 397 degrees of freedom  
AIC: 532.43

Number of Fisher Scoring iterations: 4

Interaction models

```
summary(fitbi)
```

Call:

```
glm(formula = ymat ~ a * b, family = binomial, data = ddw)
```

Deviance Residuals:

```
[1] 0 0 0 0
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	1.5856	0.2662	5.956	2.58e-09	***
aa2	-2.0329	0.3360	-6.050	1.45e-09	***
bb2	-1.8268	0.3339	-5.472	4.45e-08	***
aa2:bb2	4.0894	0.4864	8.408	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 8.4059e+01 on 3 degrees of freedom  
Residual deviance: -1.4211e-14 on 0 degrees of freedom

AIC: 26.893

Number of Fisher Scoring iterations: 3

```
summary(fitagi)
```

Call:

```
glm(formula = y ~ a * b, family = binomial, data = ddag, weight
```

Deviance Residuals:

1	2	3	4	5	6	7	8
-5.561	7.762	-8.570	7.766	-8.500	8.059	-5.093	7.420

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.5856	0.2662	-5.956	2.58e-09	***
aa2	2.0329	0.3360	6.050	1.45e-09	***
bb2	1.8268	0.3339	5.472	4.45e-08	***

```
aa2:bb2      -4.0894      0.4864  -8.408  < 2e-16 ***
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)
```

```
Null deviance: 527.16  on 7  degrees of freedom  
Residual deviance: 443.11  on 4  degrees of freedom  
AIC: 451.11
```

```
Number of Fisher Scoring iterations: 5
```

```
summary(fiti)
```

```
Call:
```

```
glm(formula = y ~ a * b, family = binomial, data = dd)
```

```
Deviance Residuals:
```

Min	1Q	Median	3Q	Max
-1.3723	-0.6105	-0.5492	0.9943	1.9830

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.5856	0.2662	-5.956	2.58e-09	***
aa2	2.0329	0.3360	6.050	1.45e-09	***
bb2	1.8268	0.3339	5.472	4.45e-08	***
aa2:bb2	-4.0894	0.4864	-8.408	< 2e-16	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 527.16 on 399 degrees of freedom  
Residual deviance: 443.11 on 396 degrees of freedom  
AIC: 451.11

Number of Fisher Scoring iterations: 4

## Compare Likelihood ratio tests

```
anova(fitb0, fitba, fitbi, test = 'LRT')
```

### Analysis of Deviance Table

Model 1: ymat ~ 1

Model 2: ymat ~ a + b

Model 3: ymat ~ a \* b

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	3	84.059			
2	1	83.329	2	0.730	0.6943
3	0	0.000	1	83.329	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
anova(fitag0, fitaga, fitagi, test = 'LRT')
```

### Analysis of Deviance Table

Model 1:  $y \sim 1$

Model 2:  $y \sim a + b$

Model 3:  $y \sim a * b$

	Resid.	Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1		7	527.16			
2		5	526.43	2	0.730	0.6943
3		4	443.11	1	83.329	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
anova(fit0, fita, fiti, test = 'LRT')
```

Analysis of Deviance Table

Model 1:  $y \sim 1$

Model 2:  $y \sim a + b$

Model 3:  $y \sim a * b$

	Resid.	Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1		399	527.16			



2	397	526.43	2	0.730	0.6943
3	396	443.11	1	83.329	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Compare AIC

```
AIC(fitb0, fitba, fitbi)
```

	df	AIC
fitb0	1	104.95204
fitba	3	108.22242
fitbi	4	26.89302

```
AIC(fitag0, fitaga, fitagi)
```

	df	AIC
fitag0	1	529.1645
fitaga	3	532.4349
fitagi	4	451.1055

```
AIC(fit0, fita, fiti)
```

```
      df      AIC
fit0  1 529.1645
fita  3 532.4349
fiti  4 451.1055
```

Note how BIC use the 'wrong' n:

```
BIC(fitb0, fitba, fitbi)
```

```
      df      BIC
fitb0  1 104.33834
fitba  3 106.38130
fitbi  4  24.43819
```

```
BIC(fitag0, fitaga, fitagi)
```

```
      df      BIC
fitag0  1 529.2440
```

```
fitaga 3 532.6732
```

```
fitagi 4 451.4233
```

```
BIC(fit0, fita, fiti)
```

	df	BIC
fit0	1	533.1560
fita	3	544.4093
fiti	4	467.0714

## Exercise:

Perform log-linear model analyses and compare results.