Here is an easy way to make sure which level has been coded as "Yes". First, fit the all-cases-the-same model, `outcome` ~ 1. The fitted model value P from this model will be the proportion of cases for which the outcome was "Yes."

```
> mod2 = glm( outcome ~ 1, data=whickham, family="binomial")
```

```
> fitted(mod2)
```

```
    1     2     3     4     5
0.281 0.281 0.281 0.281 0.281
... for 1314 cases altogether ...
```

So, 28% of the cases were "Yes." But which of the two levels is "Yes?" You can check by looking at the proportion of each level in the response variable:

```
> tally( ~ outcome, data=whickham, format="proportion" )
```

```
Alive  Dead Total
0.719 0.281 1.000
```

Evidently, "Yes" means `Dead`.

If you want to dictate which of the two levels is going to be encoded as 1, you can use a comparison operation to do so:

```
> mod3 = glm( outcome=="Alive" ~ 1, data=whickham, family="binomial")
```

```
    1     2     3     4     5
0.719 0.719 0.719 0.719 0.719
... for 1314 cases altogether ...
```

In this model, "Yes" means `Alive`.

16.4.4 Analysis of Deviance

The same basic logic used in analysis of variance applies to logistic regression, although the quantity being broken down into parts is not the sum of squares of the residuals but, rather, the **deviance**.

The `anova` software will take apart a logistic model, term by term, using the order specified in the model.

```
> anova(mod, test="Chisq")
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL			1313	1560.32	
age	1	613.81	1312	946.51	0.0000
smoker	1	1.49	1311	945.02	0.2228

Notice the second argument, `test="Chisq"`, which instructs `anova` to calculate a p-value for each term. This involves a slightly different test than the F test used in linear-model Anova.