TheStatsGeek

R squared in logistic regression

In previous posts I've looked at R squared in linear regression, and argued that I think it is more appropriate to think of it is a measure of explained variation, rather than goodness of fit.

Of course not all outcomes/dependent variables can be reasonably modelled using linear regression. Perhaps the second most common type of regression model is logistic regression, which is appropriate for binary outcome data. How is R squared calculated for a logistic regression model? Well it turns out that it is not entirely obvious what its definition should be. Over the years, different researchers have proposed different measures for logistic regression, with the objective usually that the measure inherits the properties of the familiar R squared from linear regression. In this post I'm going to focus on one of them, which is McFadden's R squared, and it is the default 'pseudo R2' value reported by the Stata package. There are certain drawbacks to this measure - if you want to read more about these and some of the other measures, take a look at this 1996 Statistics in Medicine paper by Mittlbock and Schemper.

McFadden's pseudo-R squared
Logistic regression models are fitted using the method of maximum likelihood - i.e. the parameter estimates are those values which maximize the likelihood of the data which have been observed. McFadden's R squared measure is defined as

$$R^2_{\text{McFadden}} = 1 - \frac{log(L_c)}{log(L_{\text{null}})}$$

where L_c denotes the (maximized) likelihood value from the current fitted model, and L_{null} denotes the corresponding value but for the null model - the model with only an intercept and no covariates.

To try and understand whether this definition makes sense, suppose first that the covariates in our current model in fact give no predictive information about the outcome. In this case, although the likelihood value for the current model will be (it is always) larger than the likelihood of the null model, it will not be much greater. Therefore the ratio of the two log-likelihoods will be close to 1, and R^2_{McFadden} will be close to zero, as we would hope.

Next, suppose our current model explains virtually all of the variation in the outcome, which we'll denote Y. How would this happen? Remembering that the logistic regression model's purpose is to give a prediction for $P(Y=1)$ for each subject, we would need $P(Y=1) \approx 1$ for those subjects who did have $Y=1$, and $P(Y=1) \approx 0$ for those subjects who had $Y=10$. If this is the case, the probability

of seeing $Y=1$ when $P(Y=1) \approx 1$ is almost 1, and similarly the probability of seeing $Y=0$ when $P(Y=1) \approx 0$ is almost 1. This means that the likelihood value for each observation is close to 1. The log of 1 is 0, and so the log-likelihood value $log(L_c)$ will be close to 0. Then R^2_{McFadden} will be close to 1.

Of course in most empirical research typically one could not hope to find predictors which are strong enough to give predicted probabilities so close to 0 or 1, and so one shouldn't be surprised if one obtains a value of R^2_{McFadden} which is not very large.

Deterministic or inherently random?

The definition of R^2_{McFadden} also raises (I think) an interesting philosophical point. From one perspective, we might think of nature (or whatever it is we're investigating and trying to predict) as deterministic. In this case, our stochastic probability models are models which include randomness which is caused by our imperfect knowledge of predictors or our inability to correctly model their effects on the outcome. From this perspective, the definition of R^2_{McFadden} seems quite appropriate - the gold standard value of 1 corresponds to a situation where we can predict whether a given subject will have Y=0 or Y=1 with almost 100% certainty.

An alternative perspective says that there is, at some level, intrinsic randomness in nature - parts of quantum mechanics theory state (I am told!) that at some level there is intrinsic randomness. Because of this, it will never be possible to predict with almost 100% certainty whether a new subject will have Y=0 or Y=1. In this case, a value of $R^2_{\text{McFadden}} = 1$ will never be attainable. Of course the intrinsic randomness might have a relatively small impact in terms of variability in our outcome.

McFadden's R squared in R

In R, the glm (generalized linear model) command is the standard command for fitting logistic regression. As far as I am aware, the fitted glm object doesn't directly give you any of the pseudo R squared values, but McFadden's measure can be readily calculated after fitting the model, by using the stored deviance values:

```
mod <- glm(y~x, family="binomial")
1-mod$deviance/mod$null.deviance
```

To get a sense of how strong a predictor one needs to get a certain value of McFadden's R squared, we'll simulate data with a single binary predictor, X, with P(X=1)=0.5. Then we'll specify values for P(Y=1|X=0) and P(Y=1|X=1). Bigger differences between these two values corresponds to X having a stronger effect on Y. We'll first try P(Y=1|X=0)=0.3 and P(Y=1|X=1)=0.7:

```
set.seed(63126)
n <- 10000
```

```
x <- 1*(runif(n)<0.5)
pr <- (x==1)*0.7+(x==0)*0.3
y <- 1*(runif(n) < pr)
mod <- glm(y~x, family="binomial")
1-mod$deviance/mod$null.deviance
[1] 0.1320256
```

So, even with X affecting the probability of Y=1 reasonably strongly, McFadden's R2 is only 0.13. To increase it, we must make P(Y=1|X=0) and P(Y=1|X=1) more different:

```
set.seed(63126)
n <- 10000
x <- 1*(runif(n)<0.5)
pr <- (x==1)*0.9+(x==0)*0.1
y <- 1*(runif(n) < pr)
mod <- glm(y~x, family="binomial")
1-mod$deviance/mod$null.deviance
[1] 0.5539419
```

Even with X changing P(Y=1) from 0.1 to 0.9, McFadden's R squared is only 0.55. Lastly we'll try values of 0.01 and 0.99 - what I would call a very strong effect!

```
set.seed(63126)
n <- 10000
x <- 1*(runif(n)<0.5)
pr <- (x==1)*0.99+(x==0)*0.01
y <- 1*(runif(n) < pr)
mod <- glm(y~x, family="binomial")
1-mod$deviance/mod$null.deviance
[1] 0.9293177
```

Now we have a value much closer to 1. Although just a series of simple simulations, the conclusion I draw is that one should really not be surprised if, from a fitted logistic regression McFadden's R2 is not particularly large - we need extremely strong predictors in order for it to get close to 1. I personally don't interpret this as a problem - it is merely illustrating that in practice it is difficult to predict a binary event with near certainty.

Further reading

For more on approaches for assessing logistic (and other) regression models in terms of explained variation, I'd recommend looking at Harrell's Regression Modelling Strategies book, and/or Steyerberg's

Clinical Prediction Models book.

You may also be interested in:

- Area under the ROC curve - assessing discrimination in logistic…
- R squared and goodness of fit in linear regression
- R squared and adjusted R squared
- The Hosmer-Lemeshow goodness of fit test for logistic regression

This entry was posted in Uncategorized on February 8, 2014 [http://thestatsgeek.com/2014/02/08/r-squared-in-logistic-regression/] .