ACADEMIC YEAR 2017/2018
Università degli Studi di Milano
GRADUATE SCHOOL IN SOCIAL AND POLITICAL SCIENCES
APPLIED MULTIVARIATE ANALYSIS
Luigi Curini
luigi.curini@unimi.it

Do not quote without author's permission

Regression with a binary dependent variable

We have regularly used binary (dummy) variables as regressors and they caused no particular problems. But when the DV is binary, things are more difficult: what does it mean to fit a line to a DV that can take on only two values, zero and one?

One possible answer to this question is to interpret the regression function as a predicted probability. But the predicted probability interpretation also suggests that alternative nonlinear regression models can do a better job modeling these probabilities. These models are called "probit" and "logit" regression. The method used to estimate the coefficients of the probit and logit regressions is the method of maximum likelihood estimation.

Binary DP and the Linear Probability Model

The simplest model to use with binary DV is using an OLS, producing what we call "a linear probability model".

Example: we want to explain the quality of a high school in California `hiqual`. Such school has only two values (good=1 or bad=0). As IV we employ the average education `avg_ed` (ranging from 1 to 5) of the parents of the students in the high schools.

```
twoway (scatter hiqual avg_ed) (lfit hiqual avg_ed)
```

The scatterplot looks different than the other scatterplots we used because we have a binary DV. Still, it seems to show a positive relationship between the DV and the IV. Lots of low quality schools when education is low, and viceversa.

Let's estimate the regression then:

```
regress hiqual avg_ed
```

The regression gives us a coefficient of 0.43. Therefore, when education is equals 3, then the predicted value of hiqual is 0.43.

```
lincom _b[_cons]+_b[avg_ed]*3
margins, at(avg_ed=3)
```

But what does it mean for the predicted value of the binary DV to be 0.43? The key to answering this question, and more generally to understanding regression with a binary DV, is to interpret the regression as modeling the probability that the DV equals 1. Thus, the predicted value of 0.43 is interpreted as meaning that when education is 3, then the probability of having a high school is estimated to be 43%. Said differently, if there were high schools with avg_ed=3, then 43% of them would be school of high quality.

This interpretation follows from two facts. First, the regression function tells us the expected value of Y given the regressors: $E(Y|X_1, \dots, X_k)$. Second, if Y is a binary variable, then its expected value (or mean) is the probability that Y=1, that is E(Y)=Pr(Y=1). Thus for a binary variable, $E(Y|X_1, \dots, X_k) = \Pr(1|X_1, \dots, X_k)$. In short, for a binary DV the predicted (expected) value from the regression is the probability that Y=1, given (conditional on) X.

Addendum:

The *expected-value* of a random variable Y, denoted E(Y) is the long-run average value of the DV over many repeated trials or occurrences. The expected value of Y is also called the expectation of Y or the mean of Y.

The expected value of a *Bernoulli random variable* (i.e., a binary variable with just two occurrence: 1 with probability p and 0 with probability 1-p) is:

E(Y)=1*p+0*(1-p)=p

Thus, the expected value of a Bernoulli random variable is p, the probability that it takes on the value “1”.

The linear probability model is the name for the OLS when the DV is binary rather than continuous. Because the dependent variable Y is binary, the regression function corresponds to the probability that the DV equals 1, given X. The coefficient β_1 on a regressor X is the change in the probability that Y=1 associated with a unit change in X.

Problems:

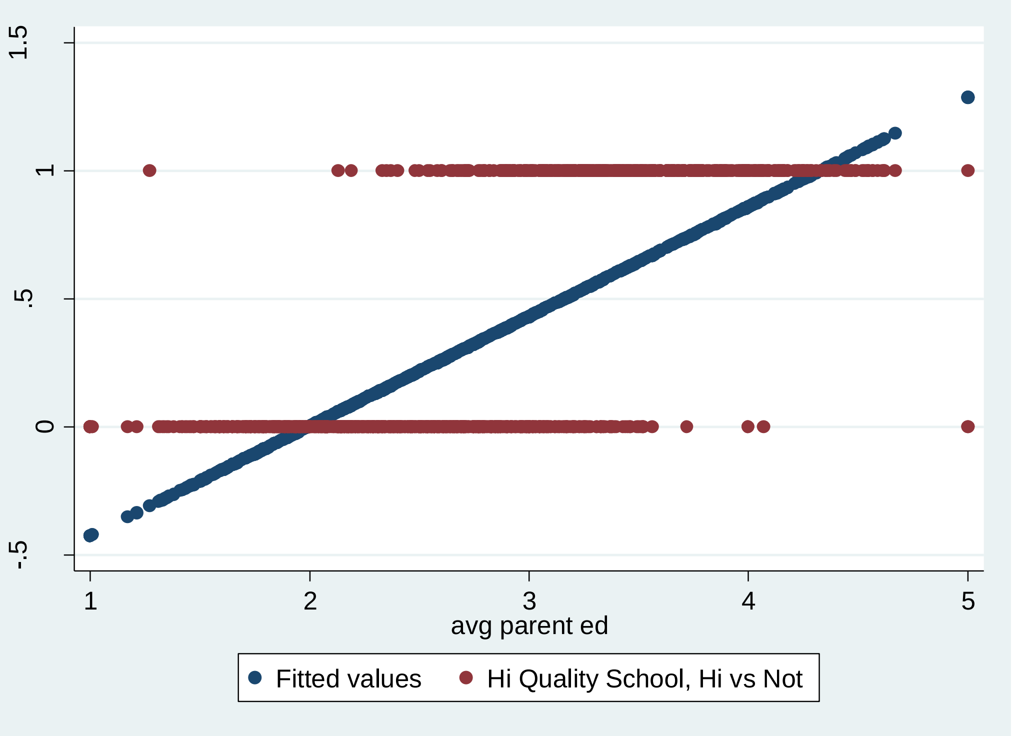
1) the R^2 has no meaning here. When the DV is continuous, it is possible to imagine a situation in which R^2 equals 1 (all the data lie exactly on the regression line). This is impossible when the DV is binary (look at the scatter before!)

2) Take a look at here:

```
predict yhat
```

```
twoway scatter yhat hiqual avg_ed, ylabel(0 1)
```

In the graph we have plotted the predicted values (called "fitted values" in the legend, the blue line) along with the observed data values (the red dots). Upon inspecting the graph, you will notice that some things that do not make sense. First, there are predicted values that are less than zero and others that are greater than +1. Such values are not possible with our outcome variable. The estimated line representing the predicted probabilities drops below 0 for very low values of education and exceeds 1 for high values. This is non-sense! A probability cannot be less than 0 or greater than 1! This nonsensical feature is an inevitable consequence of the linear regression (i.e., try to fit a line with a dichotomous DV). Also, the line does a poor job of "fitting" or "describing" the data points.



To address this problem, we need to do something else. In other words, the linear probability model is easier to use and to interpret, but it cannot capture the nonlinear nature of the true population regression function!

Probit and Logit regression

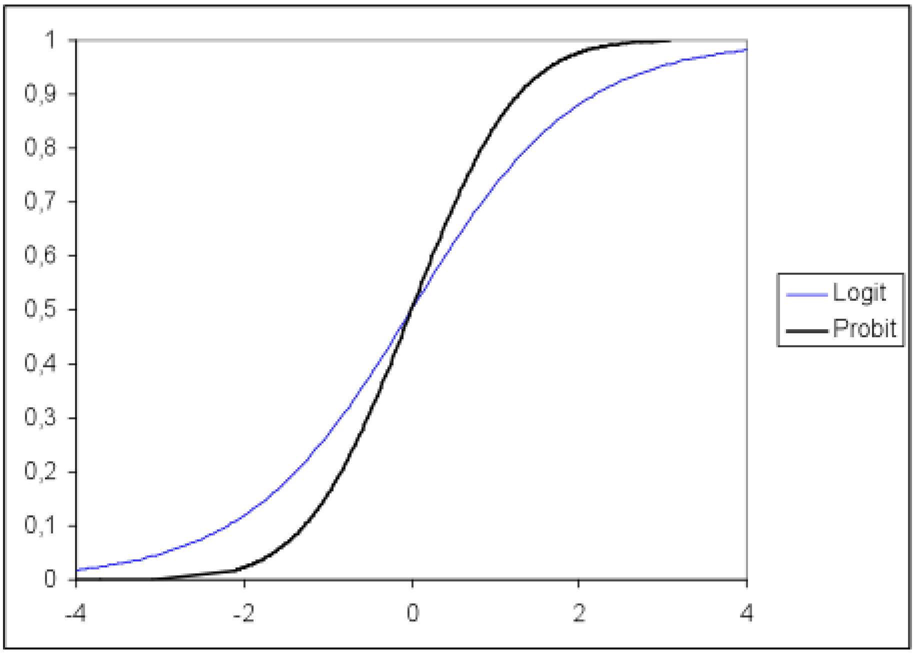
Probit and Logit regression are nonlinear regression models specifically designed for binary DV. Because a regression with a binary DV models the probability that Y=1, it makes sense to adopt a nonlinear formulation that forces the predicted values to be between 0 and 1.

Because CDF (Cumulative probability distribution functions) produce probabilities between 0 and 1 they are used in logit and probit regressions. Probit regression uses the standard normal CDF. Logit regression, uses the Logistic CDF.

In both situations, the arguments of the CDF depends on the regressors.

Addendum:

The *cumulative probability distribution* is the probability that a random variable is less than or equal to a particular value. Overall, a CDF ranges between 0 and 1



Probit regression

The probit regression model with a single regressor X is

$\Pr(Y = 1|X) = \varphi(\beta_0 + \beta_1 X_i)$ (1)

where φ (phi) is the cumulative standard normal distribution function.

```
probit hiqual avg_ed
```

For example, suppose Y is the binary quality school variable, X is the average education of parents, and β_0=-6.42 and β_1=2.03. What is the probability of quality of school if average education=3? According to (1), this probability is $\varphi(\beta_0 + \beta_1 avg\_edu)$= $\varphi(-6.42 + 2.03 * 3)$= $\varphi(-0.33)$. According to the cumulative normal distribution table (see the .pdf file: "Cumulative Standard Normal Distribution Function") this is equal to 37%. That is, when `avg_ed` is 3, the predicted probability that the school will be of a good quality is 37%, computed using the probit model with the coefficients β_0=-6.42 and β_1=2.03 (remember that previously it was 43%!).

In the probit model, the term $\beta_0 + \beta_1 X_i$ plays the role of "z" in the cumulative standard normal distribution table. Thus, the calculation just saw can equivalently be done by first computing the "z value", z= $(\beta_0 + \beta_1 avg\_edu)$= $(-6.42 + 2.03 * 3)$= (-0.33), then looking up the probability in the tail of the normal distribution to the left of z=-0.33, which is 37%.

# Here margins command gives you as an answer $\Pr(Y = 1|X = 3)$

```
margins, at(avg_ed=3)
marginsplot
```

# Here margins command gives the linear prediction (not the probability!): we get our usual value: -0.33!

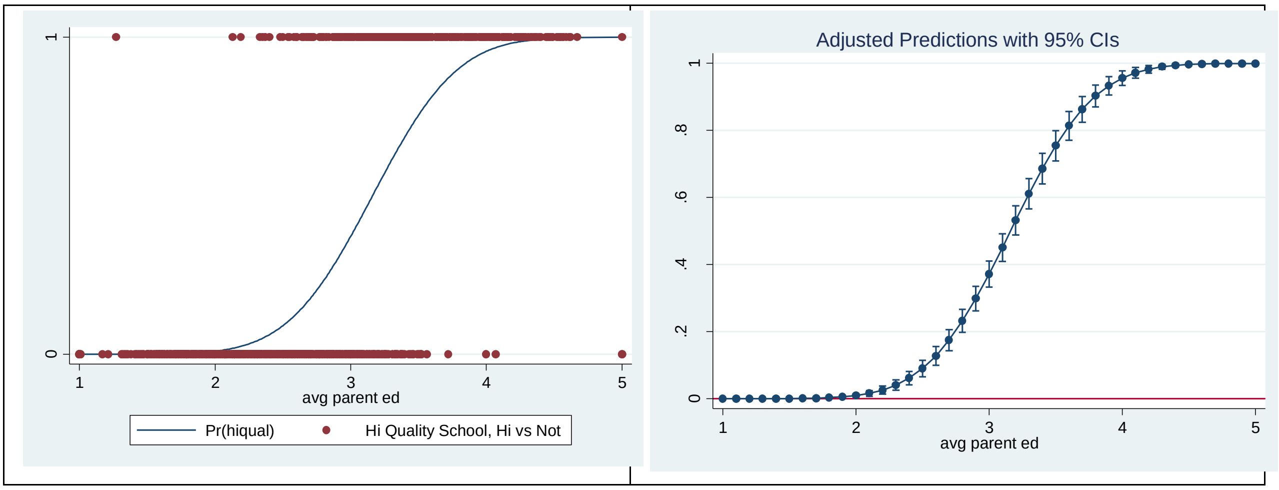
```
margins, at(avg_ed=3) predict(xb)
```

If β_1 in (1) is positive, then an increase in X increases the probability that Y=1, and viceversa. Beyond this, however, it is **not easy to interpret the probit coefficients directly**. Instead, the coefficients are best interpreted indirectly by **computing probabilities** and/or **changes in probabilities**. When there is just one regressor, the easiest way to interpret a probit regression is to plot the probabilities. The estimated probit regression function has a stretched "S" shape: it is nearly zero and flat for small values of `avg_ed` ; it turns and increases for intermediate level; and it flattens out again and is nearly one for large values.

```
probit hiqual avg_ed
predict yhat1
twoway scatter yhat1 hiqual avg_ed, ylabel(0 1)
```

Alternatively:

```
probit hiqual avg_ed
margins, at(avg_ed=(1 (0.1) 5))
marginsplot
```



For example, for `avg_ed` = 2, the estimated probability of having a school of high quality is less than 1%. When `avg_ed` = 3, the estimated probability of having a school of high quality is 37%. When `avg_ed` = 4, the estimated probability of having a school of high quality is 96%. Note that unlike the linear probability model, the probit conditional probabilities are always between 0 and 1. Also, the probit regression curve does a much better job of "fitting" or "describing" the data points.

And the effect of a change in X? The expected change is estimated in three steps: first, compute the predicted value at the original value of X using the estimated regression function; next, compute the predicted value at the changed value of X; then compute the difference between the two predicted values. Why always doing such procedure? Because the probit regression function is non-linear! Therefore the effect of a change in X depends on the starting value of X (i.e., moving `avg_ed` from 2 to 3 increases Y by 36%; moving `avg_ed` from 3 to 4 increases Y by 59%!).

```
probit hiqual avg_ed
margins,  at(avg_ed=(2 3)) contrast(atcontrast(r._at) wald)
margins,  at(avg_ed=(3 4)) contrast(atcontrast(r._at) wald)
```

More in details: the nonlinear models studies in the previous lectures are nonlinear functions of the IV but are linear functions of the unknown coefficients (parameters). Consequently the unknown coefficients of those nonlinear regression functions can be estimated by OLS. In contrast, the probit regression functions are a nonlinear function of the coefficients, given that the probit coefficients appear inside the CDF.

Because the population regression function is a nonlinear function of the coefficients, those coefficients are more complicated to estimate than linear regression functions. The coefficients of the probit model can be estimated via maximum likelihood. The maximum likelihood estimator is consistent and normally distributed in large samples, so that confidence intervals for the coefficients can be constructed in the usual way.

Logit regression

The logit regression model is similar to the probit regression model, except that the cumulative normal distribution function φ in (1) is replaced by the cumulative standard logistic distribution function, which we denote by F. The logistic cumulative distribution function has a specific functional form, defined in terms of the exponential function, which is given in equation (2) below (assuming multiple regressors)

$$\Pr\left(Y = 1\middle|X_{1,} \dots, X_k\right) = F(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k) = \frac{1}{1+exp^{-(\beta_0+\beta_1 X_1+\beta_2 X_2+\cdots+\beta_k X_k)}} \text{ (2)}$$

As with probit, the logit coefficients are best interpreted by computing predicted probabilities and differences in predicted probabilities.

```
logit hiqual avg_ed
```

What happens when education equals 3 now?

```
di 1/(1+exp(-( -12.30054 +3*3.909635  )))
```

Alternatively:

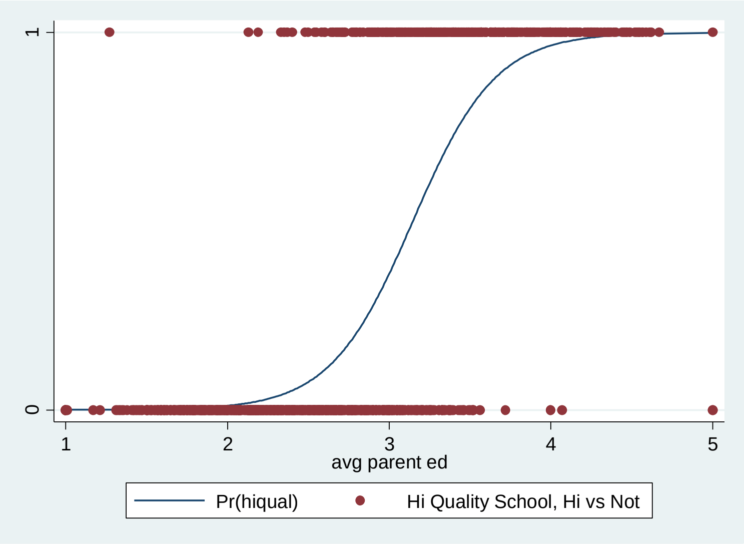
```
di exp( -12.30054 +3*3.909635  )/(1+exp( -12.30054 +3*3.909635 
))
```

Alternatively:

```
margins, at(avg_ed=3)
```

That is 36.1% compared to 37% with probit. You have to put your estimated coefficients back in the logit CDF to get your probability!

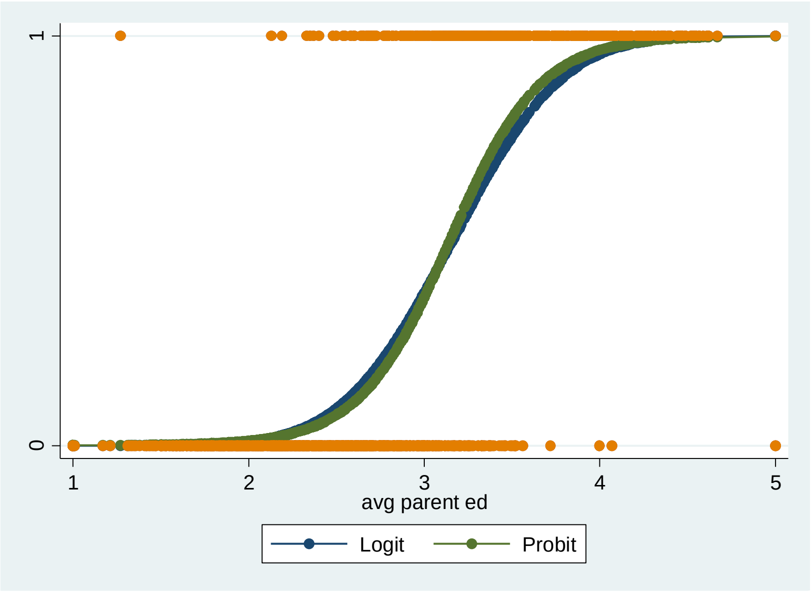
```
logit hiqual avg_ed
predict yhat2
twoway scatter yhat1 hiqual avg_ed, ylabel(0 1)
```



As before, we have calculated the predicted probabilities and have graphed them against the observed values. With the logistic regression, we get predicted probabilities that make sense: no predicted probabilities is less than zero or greater than one.

Let's compare the probit with the logit prediction:

```
corr yhat1 yhat2
twoway (scatter yhat1 hiqual avg_ed, ylabel(0 1)) ///
(scatter yhat2 hiqual avg_ed, ylabel(0 1)), legend(order(1 
"Logit" 3 "Probit"))
```



As we can see the logit and the probit regression functions are pretty similar (correlation = .999!!!). So which should you use in practice? There is no one right answer. Just pick up the one you like most. Historically the main motivation for logit regression was that the logistic cumulative distribution function could be computed faster than the normal CDF. But now, this distinction is no longer important.

Addendum:

A <u>different</u> (but <u>related</u> interpretation) of the logit coefficients:

Probability is defined as the quantitative expression of the chance that an event will occur. More formally, it is the number of times the event "occurs" divided by the number of times the event "could occur". For a simple example, let's consider tossing a coin. On average, you get heads once out of every two tosses. Hence, the probability of getting heads is 1/2 or .5.

Next let's consider the **odds**. In statistics, probability and odds are not the same. The **odds** of an event happening is defined as the probability that the event occurs divided by the probability that the event does not occur (i.e., p/(1-p)). To continue with our coin-tossing example, the probability of getting heads is .5 and the probability of not getting heads (i.e., getting tails) is also .5. Hence, the odds are .5/.5 = 1. Note that the probability of an event happening and its compliment, the probability of the event not happening, must sum to 1. Now let's pretend that we alter the coin so that the probability of getting heads is .6. The probability of not getting heads is then .4. The odds of getting heads is .6/.4 = 1.5. If we had altered the coin so that the probability of getting heads was .8, then the odds of getting heads would have been .8/.2 = 4. As you can see, when the *odds equal one*, the probability of the event happening is equal to the probability of the event not happening. When the odds are *greater than one*, the probability of the event happening is higher than the probability of the event not happening, and when the odds are *less than one*, the probability of the event happening is less than the probability of the event not happening.

Also note that odds can be converted back into a probability: probability = odds / (1+odds).

In summary:

- probability: the number of times the event occurs divided by the number of times the event could occur (possible values range from 0 to 1)
- odds: the probability that an event will occur divided by the probability that the event will not occur: probability(success) / probability(failure)

Note that Stata has two commands for logistic regression, **logit** and **logistic**. The main difference between the two is that the former displays the coefficients and the latter displays the odds. You can also obtain the odds ratios by using the **logit** command with the **or** option. Which command you use is a matter of personal preference.

From logit coefficients to odds: **Log odds** are the natural logarithm of the odds, that is **log(p/(1-p))**. The coefficients in the output of the logit regression are given in units of log odds. Therefore, the coefficients indicate the amount of change expected in the log odds when there is a one unit change in the predictor variable.

But then $\exp^{\log(p/(1-p))}$= **p/(1-p)**

```
logit  hiqual avg_ed
di exp( 3.909635  ): you have the odds ratio of increasing by 1
unit avg_ed!
logit  hiqual avg_ed, or
# as an alternative:
logistic  hiqual avg_ed
```

An odds ratio of 49.8 means that a school at a given level of average education is 49.8 times more likely to be a school with good quality than a school at the next lower level of average education.

Another example: explaining the turnout in the US Presidential election 2004:

```
logit vote_2004 educ
predict yhat
```

As already discussed, the logit assumes a nonlinear relationship between years of education and the probability of voting. That is, it is assumed that for people who lie near the extremes of the IV (respondents with either low or high level of education) a 1-year increase in education will have a weaker effect on the probability of voting than will a 1-year increase for respondents in the middle range of the IV. Let’s try to understand it better by using the predicted command.

```
tab educ, sum(yhat) nost
```

Notice that the increments are higher in the middle range of education.

Addendum: Things to consider when estimating a probit (or logit) model

As we have stated, probit (and logit) regression uses a maximum likelihood to get the estimates of the coefficients. Many of desirable properties of maximum likelihood are found as the *sample size increases*. The behavior of maximum likelihood with small sample sizes is not well understood. In particular, in small samples one may get high standard errors. In the extreme, if there are too few cases in relation to the number of variables, it may be impossible to converge on a solution. According to Long (1997, pages 53-54), 100 is a minimum sample size, and you want *at least* 10 observations per predictor. This does not mean that if you have only one predictor you need only 10 observations. Peduzzi et al. (1996) recommend that the smaller of the classes of the dependent variable have at least 10 events per parameter in the model. Pedhazur (1997) recommends sample size be at least 30 times the number of parameters being estimated. More observations are needed when the dependent variable is very lopsided; in other words, when there are very few 1's and lots of 0's, or vice versa. Moreover, when you have problems with multicollinearity, you will need a larger sample size.

Similarly, it is always better to check for empty or small cells by doing a crosstab between categorical predictors and the outcome variable. If a cell has very few cases (a small cell), the model may become unstable or it might not run at all.

It is however sometimes possible to estimate models for binary outcomes in datasets with only a small number of cases and/or when some of the cells formed by the outcome and categorical predictor variable have no observations using exact logistic regression (the **exlogistic** command).

It is also important to keep in mind that when the outcome is rare, even if the overall dataset is large, it can be difficult to estimate a probit or logit model. In that case it would be more appropriate to use for example a “Rare event logistic regression” (the **relogit** command thanks to Gary King! http://gking.harvard.edu/files/abs/0s-abs.shtml).

Measures of fit

As a measure of fit after a logit, you can rely on different statistics.

First, you can compare the initial log likelihood and the final one to have an idea of how well a model performs. For example:

```
logit vote_2004 educ
```

If education greatly improves the model’s predictive power, then the two logs would be much difference, and the final log would be much closer to 0 than the initial one (where 0 is the maximum values - because likelihoods can vary between 0 and 1, the logs of likelihoods can vary between large negative numbers and 0 (the log of 1)).

We have two measures to assess the strength of the relationship between the IV and the likelihood of the DV: the likelihood ratio (LR chi2) and pseudo R2. The first one is obtained in the following way: -2(initial log likelihood – final log likelihood).

di -2*(-553.07398- - 509.37393). Then you have a test on this (where H0 is that your model does not improve anything against a know-nothing model). In particular you have to compare the result with the ChiSquared distribution to understand if such difference is significant or not. Otherwise you can rely to our usual lrtest:

```
logit vote_2004 educ
est store full_model
logit vote_2004  if e(sample)
lrtest full_model .
```

Alternatively, you can have a pseudo R2 to seek to communicate the strength of association between DV and IV (because of the statistical foundations of logistic regression, the notion of explained variance/variation has no direct analog in logistic regression). It is a "pseudo" R-square because it is unlike the R-square found in OLS regression, where R-square measures the proportion of variance explained by the model. The pseudo R-square is not *measured* in terms of variance, since in logistic regression the variance is fixed as the variance of the standard logistic distribution (i.e., $\pi^2/3$; the variance of the standard normal distribution employed in a probit is once again fixed and equals to 1).

Stata reports the measure suggested by McFadden:

(initial log likelihood – final log likelihood) / (initial log likelihood)

```
di (-553.07398 - -509.37393)/(-553.07398)
```

However note there have been many variations of this particular pseudo R-square. We should also note that different pseudo R-squares can give very different assessments of a model's fit, and that there is no one version of pseduo R-square that is preferred by most data analysts over other versions.

Another commonly used test of model fit is to estimate the “fraction correctly predicted”. Such measure uses the following rule: if Y_i=1 and the predicted probability exceeds 50% or if Y_i=0 and the predicted probability is less than 50%, then Y_i is said to be correctly predicted. Otherwise Y_i is said to be incorrectly predicted. The “fraction correctly predicted” is the fraction of the n observations in your sample that are correctly predicted by your model. An advantage of this measure of fit is that it is easy to understand. A disadvantage is that it does not reflect the quality of the prediction: if Y_i =1, the observation is treated as correctly predicted whether the predicted probability is 51% or 99%.

```
estat classification
```

di (821+19)/1065
di 821/837
di 19/228

The overall rate of correct classification is estimated to be 78.87, with 98.09% of the voting group correctly classified (sensitivity: 821 out of 1065) and only 8.3% of the not-voting group correctly classified (specificity: 19 out of 228). Classification is sensitive to the relative sizes of each component group, and always favors classification into the larger group. This phenomenon is evident here. By default, estat classification uses a cutoff of 0.5, although you can vary this with the cutoff() option.

Another commonly used test of model fit is the Hosmer and Lemeshow's goodness-of-fit test. The idea behind the Hosmer and Lemeshow's goodness-of-fit test is that the predicted frequency and observed frequency should match closely, and that the more closely they match, the better the fit. The Hosmer-Lemeshow goodness-of-fit statistic is computed as the Pearson chi-squared from the contingency table of observed frequencies and expected frequencies.

The command estat gof more in details presents the Pearson chi-squared goodness-of-fit test for the fitted model. The Pearson chi-squared goodness-of-fit test is a test of the observed against expected number of responses using cells defined by the covariate patterns.

```
estat gof, table
```

Why a table with 15 groups?

```
table educ if vote_2004!=.
```

Similar to a test of association of a two-way table, a good fit as measured by Hosmer and Lemeshow's test will yield a large p-value (i.e., the difference between observed and predicted frequency is not significantly different). With a p-value of .19 (take a look also at the table of the Chi-Squared if you are unsure!), we can say that Hosmer and Lemeshow's goodness-of-fit test indicates that our model fits the data well.

When there are continuous predictors in the model, there will be many cells defined by the predictor variables, making a very large contingency table, which would yield significant result more than often. So a common practice is to regrouping the data by ordering on the predicted probabilities and then forming, say, 4 nearly equal-sized groups.

```
estat gof, group(4) table
```

Logistic regression with multiple IVs

Let’s see what happens if we add age to our model.

```
logit vote_2004 educ
logit vote_2004 educ age
```

First thing note that now the education coefficient increased compared to the previous situation (without age). Why this? First note that the correlation between age and education is negative. Thus in our first analysis (without age) we were also comparing older respondents (who, on average, have fewer years of schooling) with younger respondents (who, on average, have more years of schooling). Since younger people are less likely to vote than are older people, the uncontrolled effect of age worked to weaken the relationship between education and vote. In a situation like this, age is said to be a suppressor variable, because it suppresses or attenuates education’s true effect on turnout.

Overall, how does the model with education and age performs compared to the model with just age? The pseudo R2 increases. And the likelihood ratio increases. Still, is our second model significantly better than the more parsimonious only education model? Remember than in the OLS we had the R2 adjusted. Here? Use our old friend, that is, the likelihood ratio test!

To use this command, you first run the model that you want to use as the basis for comparison (the full model). Next, you save the estimates with a name using the **est store** command. Next, you run the model that you want to compare to your full model, and then issue the **lrtest** command with the name of the full model. In our example, we will name our full model **full_model**. The output of this is a likelihood ratio test which tests the null hypothesis that the coefficients of the variable(s) left out of the reduced model is/are simultaneously equal to 0. In other words, the null hypothesis for this test is that removing the variable(s) has no effect; it does not lead to a poorer-fitting model.

```
logit vote_2004 educ age
est store full_model
logit vote_2004 educ if e(sample)
lrtest full_model .
```

The chi-square statistic equals 29.26, which is statistically significant. This means that the variable that was removed to produce the reduced model resulted in a model that has a significantly poorer fit, and therefore the variable should be included in the model.

Now let's take a moment to make a few comments on the code used above. For the second logit (for the reduced model), we have added **if e(sample)**, which tells Stata to only use the cases that were included in the first model. If there were missing data on one of the variables that was dropped from the full model to make the reduced model, there would be more cases used in the reduced model. That exactly the same cases are used in both models is important because the **lrtest** assumes that the same cases are used in each model. The dot (.) at the end of the **lrtest** command is not necessary to include, but we have included it to be explicit about what is being tested. Stata "names" a model . if you have not specifically named it.

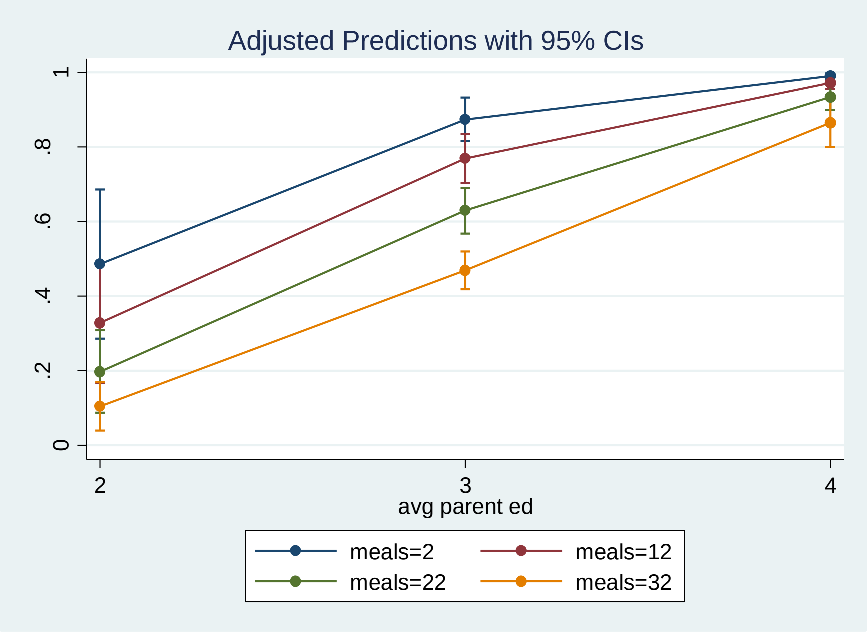
For another example, imagine that you have a model with lots of predictors in it. You could run many variations of the model, dropping one variable at a time or groups of variables at a time. Each time that you run a model, you would use the **est store** command and give each model its own name. We will try a mini-example below.

```
### another example
* full model
logit vote_2004 educ age income_hh
est store a
* with income_hh removed from the model
logit vote_2004 educ age if e(sample)
est store b
lrtest a b, stats
* with income_hh and age removed from the full model
logit vote_2004 educ if e(sample)
est store c
lrtest a c
lrtest b c
```

Now, given that you have more than one IV, any effect of a unit change of one IV on the predicted probability of your DV, will depend on the value of the other IV (besides being a non-linear relationship between each IV and the DV as we already noted!). In other words, also the relationship between a given IV and DV changes accordingly to the value of the control IV in a non-linear way.

Let's go back to our previous example with Quality of Schools. And let's run a probit regression with multiple regressors.

```
probit hiqual avg_ed enroll meals
margins, at(avg_ed=(2 3 4) (mean)_all)
margins, at(avg_ed=(2 3 4) meals=2 (mean)_all)
margins, at(avg_ed=(2 3 4) meals=12 (mean)_all)
margins, at(avg_ed=(2 3 4) meals=(2 12 22 32) (mean)_all)
marginsplot
```



As a result a probit and/or a logit (that is, non-linear models) implicitly forces the effect of all the IV to depends on each other. That is, even in an additive model, the marginal effect of X depends on the value of the other IV as well (so we need to consider also the other IV not involved in the interaction; moreover, the beta changed according to these values).

However, note that this dependence occurs whether the analyst's hypothesis is conditional or not – it is just part of deciding to use a non-linear model as logit: it is always THERE! If one wants to test a conditional hypothesis in a meaningful way, then the analyst has to include an explicit **interaction term** (or a **quadratic one**!) in the model.

Let's see an example with a **quadratic term**:

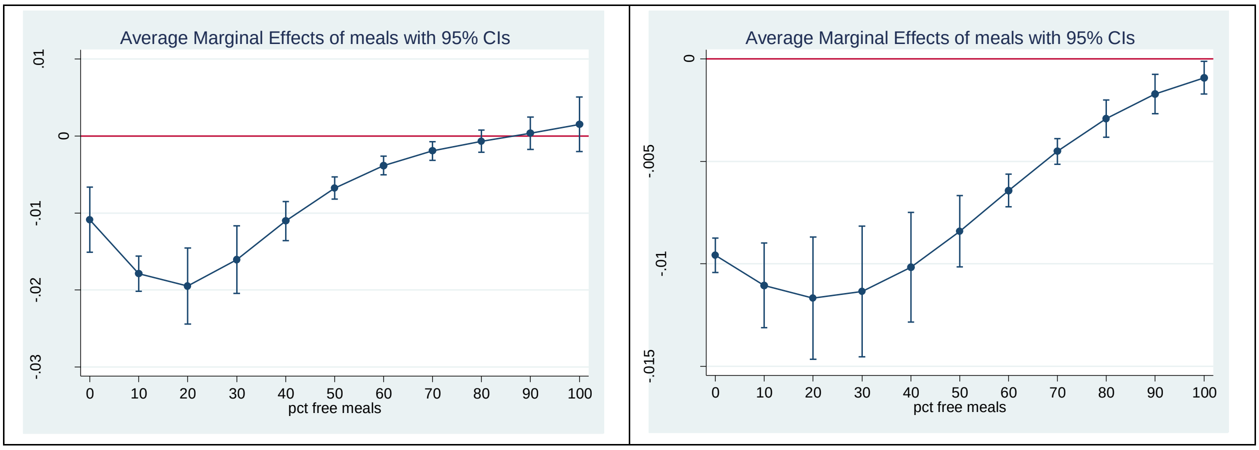
```
probit hiqual avg_ed enroll c.meals##c.meals
```

$\Pr(Y = 1|X)$ at different values of meals from 0 to 100

```
margins,   at(meals=(0 (10) 100))
marginsplot
```

# marginal impact of increasing meals by 1 unit

```
probit hiqual avg_ed enroll c.meals##c.meals
margins, dydx(meals) at(meals=(0 (10) 100))
marginsplot, yline(0)
```



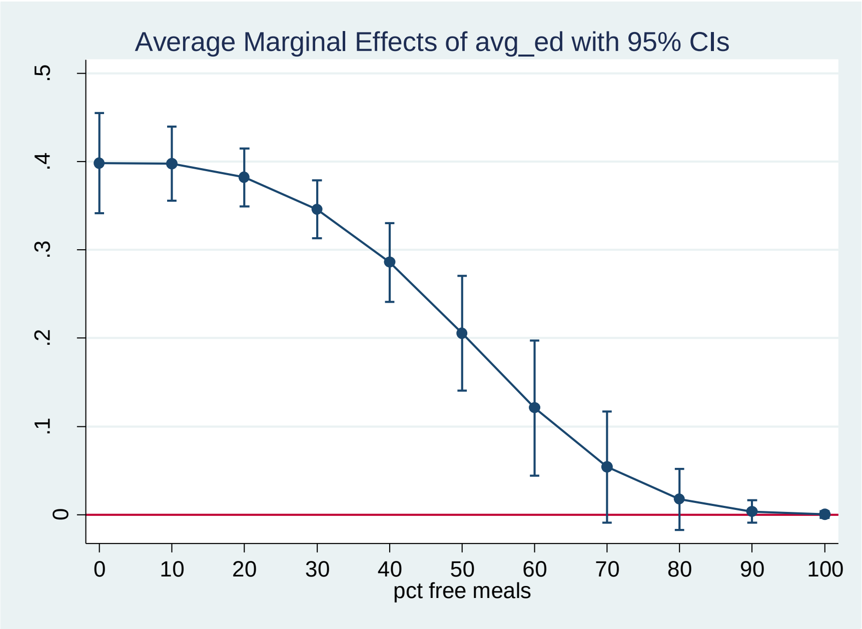
On the left panel the marginal impact of meals when you have a quadratic term in the model, on the right panel the marginal impact of meals when you DO NOT have a quadratic term in the model. A substantial difference!

Let's see an example with an **interaction term**:

```
probit hiqual c.avg_ed##c.meals enroll
```

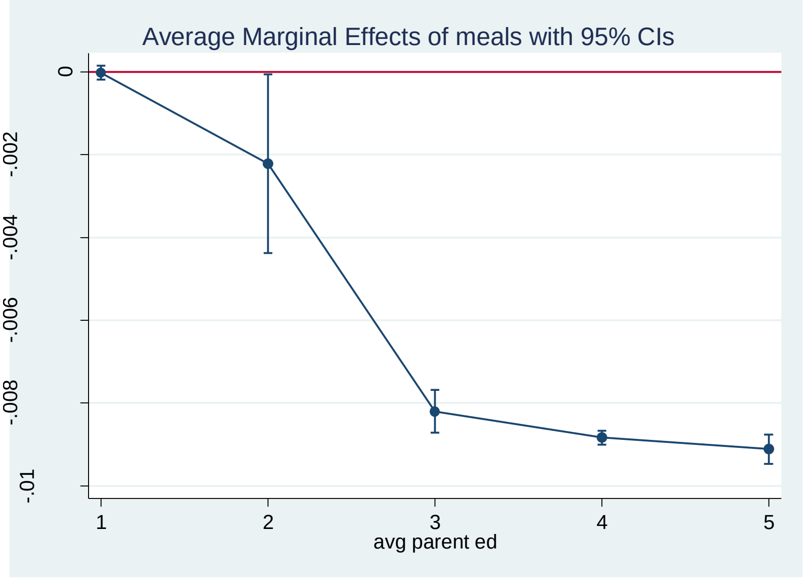
# marginal impact of increasing avg_ed by 1 unit

```
margins, dydx(avg_ed) at(meals=(0 (10) 100))
marginsplot, yline(0)
```



# marginal impact of increasing meals by 1 unit

```
margins, dydx(meals) at(avg_ed=(1 (1) 5))
marginsplot, yline(0)
```



Addendum:

Be careful with margins when using a non-linear model!

Compare the results of these two margins:

```
probit hiqual avg_ed enroll meals
margins, at(avg_ed=(2 3 4) (mean)_all)
margins, at(avg_ed=(2 3 4) )
```

Why are the two set of results different? In the second case technically the margins command for `avg_ed` is estimated by considering the values of the two other IVs as they are found in the dataset!!! For example, for the first observation: enroll = 638 meals = 78, for the second observation: enroll = 308 meals = 49, and so on.

If we had an OLS this wouldn't matter at all! Why? No matter the value of enroll or meals, the impact of `avg_ed` is always constant!

```
reg hiqual avg_ed enroll meals
margins, at(avg_ed=(2 3 4) (mean)_all)
margins, at(avg_ed=(2 3 4))
```

But this could matter for non-linear model (as we will see: both if we use a quadratic, interaction in an OLS, or <u>always</u> with a logit/probit!)

Addendum:

Additional and useful command for logit models:

```
# findit fitstat
logit vote_2004 educ age
fitstat
```

The **fitstat** command gives a listing of various pseudo-R-squares. As you can see from the output, some statistics indicate that the model fit is relatively good, while others indicate that it is not so good. The values are so different because they are measuring different things. We will not discuss the items in this output; rather, our point is to let you know that there is little agreement regarding an R-square statistic in logistic regression, and that different approaches lead to very different conclusions. If you use an R-square statistic at all, use it with great care.

You can use fitstat also to compare among different nested models:

```
# Comparing nested models with fitstat
logit vote_2004 educ age income_hh
fitstat, saving(m1)
logit vote_2004 educ age if e(sample)
fitstat, using(m1)
```

Other limited dependent variable models

The binary DV is an example of a DV with a limited range, that is of a limited DV. Other models of this type:

1) censored and truncated regression models: when you have a ceiling or a floor on the distribution of you data by construction: for example, IQ test – you cannot score more than 100 even if you ideally could have more than 100 as you IQ test! i.e., not all people with a IQ of 100 are the same! But you cannot observe them! Observations are simply unavailable when the DV is above or below a certain cutoff

2) count data: when the DV is a counting number

3) ordered responses: when the DV is an ordinary variable (i.e., which is the degree you have - ordinary logit model)

4) discrete choice data: when the DV is a categorical variable (i.e., the mode of transport you select to go to University - multinomial logit regression models)