What is Probit Regression and how is it used in SAS data analysis?

Authored by

stats writer

June 29, 2024

RECOMMENDED CITATION

stats writer (2024). *What is Probit Regression and how is it used in SAS data analysis?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=157896

Probit Regression is a statistical method used to model the relationship between a binary response variable and one or more independent variables. It is commonly used in SAS data analysis to predict the probability of an event occurring, based on the values of the independent variables. This method is particularly useful in situations where the response variable is not normally distributed and cannot be modeled using traditional linear regression techniques. Probit Regression in SAS involves fitting a probit model to the data and estimating the regression coefficients using maximum likelihood estimation. This allows for the interpretation of the effects of the independent variables on the probability of the event occurring. It is often used in fields such as economics, marketing, and social sciences to analyze and predict the likelihood of certain outcomes.

Probit Regression | SAS Data Analysis Examples

Probit regression, also called a probit model, is used to model dichotomous or binary outcome variables. In the probit model, the inverse standard normal distribution of the probability is modeled as a linear combination of the predictors.

Please Note: The purpose of this page is to show how to use various data analysis commands. It does not cover all aspects of the research process which researchers are expected to do. In particular, it does not cover data cleaning and checking, verification of assumptions, model diagnostics and potential follow-up analyses.

Examples

Example 1: Suppose that we are interested in the factors that influence
whether a political candidate wins an election. The outcome (response) variable
is binary (0/1); win or lose. The predictor variables of interest are the
amount of money spent on the campaign, the amount of time spent campaigning
negatively and whether the candidate is an incumbent.

Example 2: A researcher is interested in how variables, such as GRE (Graduate Record Exam scores), GPA (grade point average) and prestige of the undergraduate institution, effect
admission into graduate school. The outcome variable, admit/don't admit, is binary.

Description of the Data

For our data analysis below, we are going to expand on Example 2 about getting
into graduate school. We have generated hypothetical data, which can be
obtained from our website by clicking on

binary.sas7bdat.
You can store this anywhere you like, but our the example syntax assumes it has been
stored in c:data.
This data set has a binary response (outcome, dependent) variable called admit. There are three predictor variables: gre, gpa and rank. We will treat the variables
gre and gpa as continuous.
The variable rank takes on the values 1 through 4. Institutions with a rank of 1 have the highest
prestige, while those with a rank of 4 have the lowest.
We start out by looking at the data.

```
proc means data="c:databinary";
var gre gpa;
run;
```

The MEANS Procedure

Variable N Mean Std Dev Minimum Maximum

GRE 400 587.7000000 115.5165364 220.0000000 800.0000000
GPA 400 3.3899000 0.3805668 2.2600000 4.0000000

```
--------------------------------------------------------------------------------

proc freq data="c:databinary";
tables rank admit admit*rank;
run;

The FREQ Procedure

Cumulative Cumulative
RANK Frequency Percent Frequency Percent
-----------------------------------------------------------
1 61 15.25 61 15.25
2 151 37.75 212 53.00
3 121 30.25 333 83.25
4 67 16.75 400 100.00

Cumulative Cumulative
ADMIT Frequency Percent Frequency Percent
-----------------------------------------------------------
0 273 68.25 273 68.25
1 127 31.75 400 100.00

Table of ADMIT by RANK

ADMIT RANK
```

Frequency|

Percent |

Row Pct |

Col Pct | 1| 2| 3| 4| Total

---------+--------+--------+--------+--------+-

0 | 28 | 97 | 93 | 55 | 273

| 7.00 | 24.25 | 23.25 | 13.75 | 68.25

| 10.26 | 35.53 | 34.07 | 20.15 |

| 45.90 | 64.24 | 76.86 | 82.09 |

---------+--------+--------+--------+--------+-

1 | 33 | 54 | 28 | 12 | 127

| 8.25 | 13.50 | 7.00 | 3.00 | 31.75

| 25.98 | 42.52 | 22.05 | 9.45 |

| 54.10 | 35.76 | 23.14 | 17.91 |

---------+--------+--------+--------+--------+-

Total 61 151 121 67 400

15.25 37.75 30.25 16.75 100.00

Analysis methods you might consider

Below is a list of some analysis methods you may have encountered.

Some of the methods listed are quite reasonable while others have either

fallen out of favor or have limitations.

Using the Probit Model

There are multiple ways to run a probit model in SAS, this page uses proc logistic with link=probit on the model statement. Alternative methods not shown on this page include using proc probit, or proc genmod. The advantage of running the model using proc logistic is that it is easier to specify the ordering of the categories than it is in proc probit. One possible advantage of using proc probit is that it will produce graphs that may help you interpret and explain the model.

Below we run the probit regression model using proc logistic. To model 1s rather than 0s, we use the descending option. We do this because by default, proc logistic models 0s rather than 1s, in this case that would mean predicting the probability of not getting into graduate school (admit=0) versus

getting in (admit=1).
Mathematically, the models are equivalent, but conceptually, it probably makes more sense to model the probability of getting into graduate school versus not getting in. The class statement tells SAS that rank is a categorical variable. The parm=ref option after the slash requests dummy coding, rather than the default effects coding, for the levels of rank.
For more information on dummy versus effects coding in proc logistic, see our FAQ page: In PROC LOGISTIC why aren't the coefficients consistent with the odds ratios?.
The model statement specifies that we are modeling the outcome admit as a function of the predictor variables gre, gpa, and rank. The link=probit option fits a probit model rather than the default logit model.

```
proc logistic data=data.binary descending;
class rank / param=ref ;
```

model admit = gre gpa rank /link=probit;
run;

The output from proc logistic is broken into several sections each of which is discussed below.

The LOGISTIC Procedure

Model Information

Data Set DATA.BINARY Written by SAS
Response Variable ADMIT
Number of Response Levels 2
Model binary probit
Optimization Technique Fisher's scoring

Number of Observations Read 400
Number of Observations Used 400

Response Profile

Ordered Total
Value ADMIT Frequency

1 1 127
2 0 273

Probability modeled is ADMIT=1.

Class Level Information

Class Value Design Variables

RANK 1 1 0 0
2 0 1 0
3 0 0 1
4 0 0 0

Model Convergence Status

Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics

Intercept
Intercept and
Criterion Only Covariates

AIC 501.977 470.413
SC 505.968 494.362
-2 Log L 499.977 458.413

Testing Global Null Hypothesis: BETA=0

Test Chi-Square DF Pr > ChiSq

Likelihood Ratio 41.5633 5 ChiSq

GRE 1 4.4767 0.0344
GPA 1 5.8685 0.0154
RANK 3 21.3611 <.0001

Parameter DF Estimate Error Chi-Square Pr > ChiSq

Intercept 1 -3.3225 0.6633 25.0872 <.0001
GRE 1 0.00138 0.000650 4.4767 0.0344
GPA 1 0.4777 0.1972 5.8685 0.0154
RANK 1 1 0.9359 0.2453 14.5606 0.0001
RANK 2 1 0.5205 0.2109 6.0904 0.0136
RANK 3 1 0.1237 0.2240 0.3053 0.5806

Association of Predicted Probabilities and Observed Responses

Percent Concordant 69.1 Somers' D 0.385
Percent Discordant 30.6 Gamma 0.387
Percent Tied 0.4 Tau-a 0.167
Pairs 34671 c 0.693

The table above gives information about the relationship between the predicted probabilities from our model, and the actual outcomes in our data.

The output shown above gives a test for the overall effect of rank as well as coefficients that describe the difference between the reference group (rank=4) and each of the other three groups. We can also test for differences between the other levels of rank. For example, we might want to test for a difference in coefficients for rank=2 and rank=3. We can test this type of hypothesis by adding a contrast statement to the code for proc logistic. The syntax shown below is the same as that shown above, except that it uses the contrast statement.
Following the word contrast, is the label that will appear in the output, enclosed in single quotes (i.e. 'rank 2 vs. rank 3'). This is followed by the name of the variable we wish to test hypotheses about (i.e. rank),

and a vector (i.e., 0 1 -1) that describes the desired comparison. In this
case the value computed is the difference between the coefficients for rank=2 and rank=3.
After the slash (i.e. /) we use the estimate = parm option to
request that the estimate be the difference in coefficients. For
more information on the contrast statement, see our FAQ page
How can I create contrasts with proc logistic?.

```
proc logistic data=data.binary descending;
class rank / param=ref ;
model admit = gre gpa rank /link=probit;
contrast 'rank 2 vs. 3' rank 0 1 -1 / estimate=parm;
run;
```

Contrast Rows Estimation and Testing Results

Standard Wald
Contrast Type Row Estimate Error Alpha Confidence Limits Chi-Square Pr > ChiSq

rank 2 vs. 3 PARM 1 0.3967 0.1681 0.05 0.0673 0.7261

5.5725 0.0182

Because the models are the same, most of the output produced by the above proc logistic command is the same as before. The only difference is the additional output produced by the contrast statement (shown above). Under the heading Contrast Test Results we see the label for the contrast (rank 2 vs 3) along with its degrees of freedom, Wald chi-square statistic, and p-value. Based on the p-value in this table we know that the coefficient for rank=2 is significantly different from the coefficient for rank=3.

The second table, shows more detailed information, including the actual estimate of the difference (under Estimate), it's standard error, confidence limits, test statistic, and p-value. We can see that the estimated difference was 0.3967, indicating that having attended an undergraduate institution with a rank of 2, versus an institution with a rank of 3, increases the z-

score by 0.4.

You can also use predicted probabilities to help you understand the model.
The contrast statement can be used to estimate predicted probabilities by
specifying estimate=prob. In the syntax below we use multiple contrast
statements to estimate the predicted probability of admission as gre
changes from 200 to 800 (in increments of 100). When estimating the predicted
probabilities we hold gpa constant at
3.39 (its mean), and rank at 2. The word intercept followed by a 1 indicates that the intercept for
the model is to be included in estimate.

```
proc logistic data=data.binary descending;
class rank / param=ref ;
model admit = gre gpa rank /link=probit;
contrast 'gre=200' intercept 1 gre 200 gpa 3.3899 rank 0 1 0 / estimate=prob;
contrast 'gre=300' intercept 1 gre 300 gpa 3.3899 rank 0 1 0 / estimate=prob;
```

```
contrast 'gre=400' intercept 1 gre 400 gpa 3.3899 rank 0 1 0 / estimate=prob;
contrast 'gre=500' intercept 1 gre 500 gpa 3.3899 rank 0 1 0 / estimate=prob;
contrast 'gre=600' intercept 1 gre 600 gpa 3.3899 rank 0 1 0 / estimate=prob;
contrast 'gre=700' intercept 1 gre 700 gpa 3.3899 rank 0 1 0 / estimate=prob;
contrast 'gre=800' intercept 1 gre 800 gpa 3.3899 rank 0 1 0 / estimate=prob;
run;
```

Contrast Test Results

Wald
Contrast DF Chi-Square Pr > ChiSq

gre=800 1 0.2452 0.6205

Contrast Test Results

Wald
Contrast DF Chi-Square Pr > ChiSq

gre=800 1 0.2452 0.6205

Contrast Estimation and Testing Results by Row

Standard Wald
Contrast Type Row Estimate Error Alpha Confidence Limits Chi-Square Pr > ChiSq

gre=200 PROB 1 0.1821 0.0746 0.05 0.0720 0.3615 10.3375 0.0013
gre=300 PROB 1 0.2206 0.0662 0.05 0.1136 0.3698 11.8864 0.0006
gre=400 PROB 1 0.2635 0.0552 0.05 0.1676 0.3816 14.0168 0.0002
gre=500 PROB 1 0.3103 0.0442 0.05 0.2296 0.4014 15.6564 <.0001
gre=600 PROB 1 0.3604 0.0396 0.05 0.2861 0.4404 11.4014 0.0007
gre=700 PROB 1 0.4130 0.0480 0.05 0.3222 0.5087 3.1785 0.0746
gre=800 PROB 1 0.4672 0.0661 0.05 0.3415 0.5963 0.2452 0.6205

As with the previous example, we have omitted most of the proc logistic output, because it is the same as before. The predicted probabilities are included in the

column labeled Estimate in the second table in the output. Looking at the estimates, we can see that the predicted probability of being admitted is only 0.18 if one's GRE score is 200, but increases to 0.47 if one's GRE score is 800, holding gpa at its mean (3.39), and rank at 2.

Things to consider

References

Hosmer, D. & Lemeshow, S. (2000). Applied Logistic Regression (Second Edition).

New York: John Wiley & Sons, Inc.

Long, J. Scott (1997). Regression Models for Categorical and Limited Dependent Variables.

Thousand Oaks, CA: Sage Publications.

See Also