Dummy Variable Trap in Regression Models

DUMMY VARIABLE TRAP IN REGRESSION MODELS

Using categorical data in Multiple Regression Models is a powerful method to include non-numeric data types into a regression model. Categorical data refers to data values which represent categories - data values with a fixed and unordered number of values, for instance gender (male/female) or season (summer/winder/spring/fall). In a regression model, these values can be represented by dummy variables - variables containing values such as 1 or 0 representing the presence or absence of the categorical value.

By including dummy variable in a regression model however, one should be careful of the Dummy Variable Trap. The Dummy Variable trap is a scenario in which the independent variables are multicollinear - a scenario in which two or more variables are highly correlated; in simple terms one variable can be predicted from the others.

To demonstrate the Dummy Variable Trap, take the case of gender (male/female) as an example. Including a dummy variable for each is redundant (of male is 0, female is 1, and vice-versa), however doing so will result in the following linear model:

```
y ~ b + {0|1} male + {0|1} female
```

Represented in matrix form:

```
       | y1 |
       | y2 |
   Y = | y3 |
       |... |
       | yn |


       | 1   m1   F1 |
       | 1   m2   f2 |
   X = | 1   m3   f3 |
       |... ...  ... |
       | 1   mn   fn |
```

In the above model, the sum of all category dummy variable for each row is equal to the intercept value of that row - in other words there is perfect multi-collinearity (one value can be predicted from the other values). Intuitively, there is a duplicate category: if we dropped the male category it is inherently defined in the female category (zero female value indicate male, and vice-versa).

The solution to the dummy variable trap is to drop one of the categorical variables (or alternatively, drop the intercept constant) - if there are m number of categories, use m-1 in the model, the value left out can be thought of as the reference value and the fit values of the remaining categories represent the change from this reference.

As an example, lets take data containing 3 categories - C1, C2, and C3:

C1	C2	C3	y
1	0	0	12.4
1	0	0	11.9
0	1	0	8.3
0	1	0	3.1

0	0	1	5.4
0	0	1	6.2

Using R, we can fit this model in several ways, but for demonstration I'll use the ordinary least squares linear algebra equation:

$$\mathbf{B = (X^TX)^{-1}X^TY}$$

```
> Y
   1    2    3    4    5    6
12.4 11.9  8.3  8.1  5.4  6.2
> X
  C1 C2 C3 b
1  1  0  0 1
2  1  0  0 1
3  0  1  0 1
4  0  1  0 1
5  0  0  1 1
6  0  0  1 1
> solve(t(X)%*% X)
Error in solve.default(t(X) %*% X) :
  Lapack routine dgesv: system is exactly singular: U[4,4] = 0
```

Whoops, the matrix cannot be inverted because it is singular. To fix the issue, we can remove the intercept, or alternatively remove one of the dummy variable columns

```
> X = X[,-1]
> X
  C2 C3 b
1  0  0 1
2  0  0 1
3  1  0 1
4  1  0 1
5  0  1 1
6  0  1 1
> solve(t(X) %*% X) %*% t(X) %*% Y
    [,1]
C2 -3.95
C3 -6.35
b  12.15
```

The calculated values are now referenced to the dropped dummy variable (in this case C1). In other words, if the category is C2 it is -3.95 less than the reference (in this example the reference value is 12.15).

In some cases it may be necessary (or educational) to program dummy variables directly into a model. However in most cases a statistical package such as R can do the math for you - in R categories can be represented by factors, letting R deal with the details:

```
> a
     y C
1 12.4 1
2 11.9 1
3  8.3 2
4  8.1 2
5  5.4 3
6  6.2 3
#Column C is a factor column
> class(a[,2])
[1] "factor"
> lm(y ~ ., a)

Call:
lm(formula = y ~ ., data = a)

Coefficients:
(Intercept)           C2           C3
      12.15        -3.95        -6.35
```

The same answer produced with factors as using dummy variables directly (above).

<-- Line Numbers in Java Swing Enable/Disable JComboBox Items in Java Swing -->

Add Comment

There are no comments on this article.

Back to Articles

PRIVACY POLICY | COPYRIGHT | LINKS | SITEMAP