

Categorical Predictor Variables

Utrecht University Winter School: Regression in R



**Utrecht
University**

Kyle M. Lang

Department of Methodology & Statistics
Utrecht University

2022-02-02

Outline

Dummy Codes

Effects Codes

- Unweighted Effects Codes

- Weighted Effects Codes

Significance Testing



Categorical Predictors

Most of the predictors we've considered thus far have been *quantitative*.

- Continuous variables that can take any real value in their range
- Interval or Ratio scaling
- If we use ordinal items as predictors, we assume interval scaling.

We often want to include grouping factors as predictors.

- These variables are *qualitative*.
 - Their values are simply labels.
 - There is no ordering of the categories.
 - Nominal scaling



How to Model Categorical Predictors

We need to be careful when we include categorical predictors into a regression model.

- The variables need to be coded before entering the model.

Consider the following indicator of major:

$$X_{maj} = \{1 = \textit{Law}, 2 = \textit{Economics}, 3 = \textit{Data Science}\}$$

- What would happen if we naïvely used this variable to predict program satisfaction?



How to Model Categorical Predictors

```
dataDir <- "../.../data/"
mDat    <- readRDS(paste0(dataDir, "major_data.rds"))

mDat[seq(25, 150, 25), ]

      sat majF majN
25  1.9  law    1
50  1.4  law    1
75  4.3 econ    2
100 4.1 econ    2
125 5.7  ds     3
150 5.1  ds     3

out <- lm(sat ~ majN, data = mDat)
```

How to Model Categorical Predictors

```
partSummary(out, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.33200	0.12060	-2.753	0.00664
majN	2.04500	0.05582	36.632	< 2e-16

Residual standard error: 0.5582 on 148 degrees of freedom

Multiple R-squared: 0.9007, Adjusted R-squared: 0.9

F-statistic: 1342 on 1 and 148 DF, p-value: < 2.2e-16

Dummy Coding

The most common way to code categorical predictors is *dummy coding*.

- A G -level factor (i.e., one that represents G groups) will be transformed into a set of $G - 1$ dummy codes.
- Each code is a variable on the dataset that equals 1 for observations corresponding to the code's group and equals 0, otherwise.
- The group without a code is called the *reference group*.



Example Dummy Code

Let's look at the simple example of coding biological sex:

	sex	male
1	female	0
2	male	1
3	male	1
4	female	0
5	male	1
6	female	0
7	female	0
8	male	1
9	female	0
10	female	0



Example Dummy Codes

Now, a slightly more complex example:

	drink	juice	tea
1	juice	1	0
2	coffee	0	0
3	tea	0	1
4	tea	0	1
5	tea	0	1
6	tea	0	1
7	juice	1	0
8	tea	0	1
9	coffee	0	0
10	juice	1	0



Example

First, an example with a single, binary dummy-coded variable:

```
## Read in some data:  
cDat <- readRDS(paste0(dataDir, "cars_data.rds"))  
  
## Fit and summarize the model:  
out1 <- lm(price ~ mtOpt, data = cDat)
```

Example

```
partSummary(out1, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	23.841	1.623	14.691	<2e-16
mtOpt	-6.603	2.004	-3.295	0.0014

Residual standard error: 9.18 on 91 degrees of freedom

Multiple R-squared: 0.1066, Adjusted R-squared: 0.09679

F-statistic: 10.86 on 1 and 91 DF, p-value: 0.001403

Example

Fit a more complex model:

```
out2 <- lm(price ~ front + rear, data = cDat)
partSummary(out2, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	17.63000	2.76119	6.385	7.33e-09
front	-0.09418	2.96008	-0.032	0.97469
rear	11.32000	3.51984	3.216	0.00181

Residual standard error: 8.732 on 90 degrees of freedom

Multiple R-squared: 0.2006, Adjusted R-squared: 0.1829

F-statistic: 11.29 on 2 and 90 DF, p-value: 4.202e-05

Example

Include two sets of dummy codes:

```
out3 <- lm(price ~ mtOpt + front + rear, data = cDat)
partSummary(out3, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	21.7187	2.9222	7.432	6.25e-11
mtOpt	-5.8410	1.8223	-3.205	0.00187
front	-0.2598	2.8189	-0.092	0.92677
rear	10.5169	3.3608	3.129	0.00237

Residual standard error: 8.314 on 89 degrees of freedom

Multiple R-squared: 0.2834, Adjusted R-squared: 0.2592

F-statistic: 11.73 on 3 and 89 DF, p-value: 1.51e-06

Effects Coding

Another useful form of categorical variable coding is *effects coding*.

- Effects codes can be *weighted* or *unweighted*.



Effects Coding

Another useful form of categorical variable coding is *effects coding*.

- Effects codes can be *weighted* or *unweighted*.

We'll first discuss *unweighted* effects codes.

- Unweighted effects codes are identical to dummy codes except that “reference group” rows get values of -1 on all codes.
- The intercept is interpreted as the unweighted mean of the group-specific means of Y .
- The slope associated with each code represents the difference between the coded group's mean of Y and the mean of the group-specific means of Y .



Example Unweighted Effects Codes

	sex	male.ec
1	female	-1
2	male	1
3	male	1
4	female	-1
5	male	1
6	female	-1
7	female	-1
8	male	1
9	female	-1
10	female	-1

	drink	juice.ec	tea.ec
1	juice	1	0
2	coffee	-1	-1
3	tea	0	1
4	tea	0	1
5	tea	0	1
6	tea	0	1
7	juice	1	0
8	tea	0	1
9	coffee	-1	-1
10	juice	1	0

Example

```
## Model with single effects code:
```

```
out6 <- lm(price ~ mtOpt.ec, data = cDat)  
partSummary(out6, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	20.539	1.002	20.501	<2e-16
mtOpt.ec	-3.301	1.002	-3.295	0.0014

Residual standard error: 9.18 on 91 degrees of freedom

Multiple R-squared: 0.1066, Adjusted R-squared: 0.09679

F-statistic: 10.86 on 1 and 91 DF, p-value: 0.001403

Example

```
## Model with two effects codes (for a variable with G = 3):  
out7 <- lm(price ~ front.ec + rear.ec, data = cDat)  
partSummary(out7, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	21.372	1.226	17.433	< 2e-16
front.ec	-3.836	1.372	-2.796	0.00632
rear.ec	7.578	1.758	4.310	4.16e-05

Residual standard error: 8.732 on 90 degrees of freedom

Multiple R-squared: 0.2006, Adjusted R-squared: 0.1829

F-statistic: 11.29 on 2 and 90 DF, p-value: 4.202e-05

Weighted Effects Coding

Weighted effects codes differ from the unweighted version only in how they code the “reference group” rows.

- In weighted effects codes the “reference group” rows get negative fractional values on all codes.
 - Let $g = 1, 2, \dots, G$ index groups.
 - Take the first group as the “reference group.”
 - Then, the g th code's reference group rows will take values of $-N_g/N_1$.
- The intercept is interpreted as the weighted mean of the group-specific outcome means.
 - The arithmetic mean of Y .
- Each slope represents the difference from that group's mean outcome and the overall mean of Y .



Example Weighted Effects Codes

	sex	male.wec
1	female	$-N_{male}/N_{female}$
2	male	1
3	male	1
4	female	$-N_{male}/N_{female}$
5	male	1
6	female	$-N_{male}/N_{female}$
7	female	$-N_{male}/N_{female}$
8	male	1
9	female	$-N_{male}/N_{female}$
10	female	$-N_{male}/N_{female}$

	drink	juice.wec	tea.wec
1	juice	1	0
2	coffee	$-N_{juice}/N_{coffee}$	$-N_{tea}/N_{coffee}$
3	tea	0	1
4	tea	0	1
5	tea	0	1
6	tea	0	1
7	juice	1	0
8	tea	0	1
9	coffee	$-N_{juice}/N_{coffee}$	$-N_{tea}/N_{coffee}$
10	juice	1	0

Example

```
## Model with single effects code:  
out8 <- lm(price ~ mtOpt.wec, data = cDat)  
partSummary(out8, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	19.5097	0.9519	20.495	<2e-16
mtOpt.wec	-2.2720	0.6895	-3.295	0.0014

Residual standard error: 9.18 on 91 degrees of freedom

Multiple R-squared: 0.1066, Adjusted R-squared: 0.09679

F-statistic: 10.86 on 1 and 91 DF, p-value: 0.001403

Example

```
## Model with two effects codes (for a variable with G = 3):  
out9 <- lm(price ~ front.wec + rear.wec, data = cDat)  
partSummary(out9, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	19.5097	0.9054	21.547	< 2e-16
front.wec	-1.9739	0.5640	-3.500	0.000727
rear.wec	9.4403	1.9863	4.753	7.57e-06

Residual standard error: 8.732 on 90 degrees of freedom

Multiple R-squared: 0.2006, Adjusted R-squared: 0.1829

F-statistic: 11.29 on 2 and 90 DF, p-value: 4.202e-05

Significance Testing for Categorical Variables

For variables with only two levels, we can test the overall factor's significance by evaluating the significance of its single code.

```
partSummary(out1, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	23.841	1.623	14.691	<2e-16
mtOpt	-6.603	2.004	-3.295	0.0014

Residual standard error: 9.18 on 91 degrees of freedom

Multiple R-squared: 0.1066, Adjusted R-squared: 0.09679

F-statistic: 10.86 on 1 and 91 DF, p-value: 0.001403

Significance Testing for Categorical Variables

For variables with more than two levels, we need to simultaneously evaluate the significance of all of the variable's codes.

```
partSummary(out3, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	21.7187	2.9222	7.432	6.25e-11
mtOpt	-5.8410	1.8223	-3.205	0.00187
front	-0.2598	2.8189	-0.092	0.92677
rear	10.5169	3.3608	3.129	0.00237

Residual standard error: 8.314 on 89 degrees of freedom

Multiple R-squared: 0.2834, Adjusted R-squared: 0.2592

F-statistic: 11.73 on 3 and 89 DF, p-value: 1.51e-06

Significance Testing for Categorical Variables

```
summary(out3)$r.squared - summary(out1)$r.squared
```

```
[1] 0.1767569
```

```
anova(out1, out3)
```

Analysis of Variance Table

Model 1: price ~ mtOpt

Model 2: price ~ mtOpt + front + rear

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	91	7668.9				
2	89	6151.6	2	1517.3	10.976	5.488e-05 ***

Signif. codes:

0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Significance Testing for Categorical Variables

For models with a single nominal factor is the only predictor, we use the omnibus F-test.

```
partSummary(out2, -c(1, 2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	17.63000	2.76119	6.385	7.33e-09
front	-0.09418	2.96008	-0.032	0.97469
rear	11.32000	3.51984	3.216	0.00181

Residual standard error: 8.732 on 90 degrees of freedom

Multiple R-squared: 0.2006, Adjusted R-squared: 0.1829

F-statistic: 11.29 on 2 and 90 DF, p-value: 4.202e-05

Compare Codings

Let's dig into some numerical properties of the three coding schemes.

```
## Fit models using all three codings:
dcOut  <- lm(price ~ front + rear,          data = cDat)
ecOut  <- lm(price ~ front.ec + rear.ec,    data = cDat)
wecOut <- lm(price ~ front.wec + rear.wec,  data = cDat)

## Compute group-specific means of 'price':
grpMeans <- tapply(cDat$price, cDat$dr, mean)
```

Compare Codings

Compare the parameter estimates to their theoretical equivalents:

```
coef(dcOut)[1] - grpMeans["4WD"]
```

```
(Intercept)  
-1.421085e-14
```

```
coef(ecOut)[1] - mean(grpMeans)
```

```
(Intercept)  
-3.552714e-15
```

```
coef(wecOut)[1] - mean(cDat$price)
```

```
(Intercept)  
-1.065814e-14
```

Compare Codings

Compare the R^2 values from each coding scheme:

```
summary(dcOut)$r.squared
```

```
[1] 0.2006386
```

```
summary(ecOut)$r.squared
```

```
[1] 0.2006386
```

```
summary(wecOut)$r.squared
```

```
[1] 0.2006386
```

Compare Codings

Compare the F-statistics:

```
summary(dcOut)$fstatistic
```

value	numdf	dendf
11.29494	2.00000	90.00000

```
summary(ecOut)$fstatistic
```

value	numdf	dendf
11.29494	2.00000	90.00000

```
summary(wecOut)$fstatistic
```

value	numdf	dendf
11.29494	2.00000	90.00000

Compare Codings

Compare the residual standard errors:

```
summary(dcOut)$sigma
```

```
[1] 8.731638
```

```
summary(ecOut)$sigma
```

```
[1] 8.731638
```

```
summary(wecOut)$sigma
```

```
[1] 8.731638
```

Choosing a Coding Scheme

Any valid coding scheme will represent the information in the categorical variable equally well.

- All valid coding schemes produce equivalent models.

We choose a particular coding scheme based on the interpretations that we want.

- Dummy coding is useful with a meaningful reference group.
 - Control group in an experiment
 - An “industry standard” or benchmark implementation of some feature
- Dummy coding is also preferred if we don’t care about interpretation.
 - Dummy codes are the simplest to construct.



Choosing a Coding Scheme

- Weighted effects codes are good when you believe your sample is representative of the population.
 - Larger groups should be weighted more heavily in the model.
 - Parameter estimates will correctly generalize to the population.
- Unweighted effects codes are good when the group sizes in your sample do not generalize to the population.
 - Convenience samples, for example, are usually not representative.
 - When your sample is not representative, larger groups should not be weighted more heavily.
 - Unweighted effects codes are “agnostic” to differing group sizes.
 - We need to be careful with very small groups.
- Weighted effects codes with known weights are another option.

