

# Revisiting Dummy Variables in Statistics

## Table of contents

<b>1</b>	<b>Explain with an example model and sample data (say 10 rows) dummy variable with more than 2 categories. Also interpret the model coefficients</b>	<b>2</b>
<b>2</b>	<b>Answer</b>	<b>2</b>
2.0.1	Scenario . . . . .	2
2.0.2	Step 1. Create Dummy Variables . . . . .	3
2.0.3	Step 2. Fit a Model . . . . .	3
2.0.4	Step 3. Interpret the Coefficients . . . . .	4
2.0.5	Step 4. Example Predictions . . . . .	4
2.0.6	Key Concept . . . . .	4
<b>3</b>	<b>Follow-up: So if marital status has these values</b>	<b>4</b>
<b>4</b>	<b>Answer</b>	<b>4</b>
<b>5</b>	<b>With stata, show minimal codes</b>	<b>5</b>
<b>6</b>	<b>Answer</b>	<b>5</b>
6.0.1	Example Goal . . . . .	5
6.0.2	Step 1. Create a small example dataset . . . . .	6
6.0.3	Step 2. Encode the categorical variable . . . . .	6
6.0.4	Step 3. Run the regression with dummy variables . . . . .	6
6.0.5	Step 4. Interpretation . . . . .	7
<b>7</b>	<b>For this example, how can we make t tests to check differences of income by marital status</b>	<b>7</b>
<b>8</b>	<b>Answer</b>	<b>7</b>
8.1	Situation Recap . . . . .	8
8.2	1 Pairwise t-tests (simple group comparisons) . . . . .	8
8.3	2 Using regression and post-estimation tests (recommended) . . . . .	8

8.4	3 Using <code>margins</code> for all pairwise comparisons . . . . .	9
8.4.1	Summary Table . . . . .	9
<b>9</b>	<b>If I have two variables: gender (m, f), education (Primary, secondary, higher), can I perform t tests for crosstab, like male secondary vs female secondary, and all other combinations? Is it worth it?</b>	<b>10</b>
<b>10</b>	<b>Answer</b>	<b>10</b>
10.1	What you're describing . . . . .	10
10.2	Yes, you <i>can</i> do t-tests for pairs of these groups... . . . .	11
10.3	Better approach: Two-way ANOVA (Factorial ANOVA) . . . . .	11
10.4	If you still want pairwise comparisons . . . . .	12
10.5	Summary . . . . .	12

**1 Explain with an example model and sample data (say 10 rows) dummy variable with more than 2 categories. Also interpret the model coefficients**

## 2 Answer

Excellent — let's take a clear, practical example.

---

### 2.0.1 Scenario

We want to predict **monthly spending (in \$)** based on a person's **education level**.

Education level has **3 categories**:

1. High School
2. Bachelor
3. Master

Since education is **categorical (non-numeric)**, we must use **dummy variables**.

### 2.0.2 Step 1. Create Dummy Variables

We'll choose **High School** as the **reference category** (baseline).

ID	Education	Spending (\$)	Bachelor (X )	Master (X )
1	High School	200	0	0
2	Bachelor	250	1	0
3	Master	300	0	1
4	High School	220	0	0
5	Bachelor	270	1	0
6	Master	310	0	1
7	High School	210	0	0
8	Bachelor	260	1	0
9	Master	320	0	1
10	High School	190	0	0

### 2.0.3 Step 2. Fit a Model

We fit a linear regression model:

$$\text{Spending} = \beta_0 + \beta_1(\text{Bachelor}) + \beta_2(\text{Master}) + \varepsilon$$

Assume we get this result:

Coefficient	Estimate
(Intercept)	205
(Bachelor)	+55
(Master)	+100

### 2.0.4 Step 3. Interpret the Coefficients

- **Intercept** ( $\beta_0 = 205$ )  $\rightarrow$  Average spending for **High School** (the baseline group).
  - **Bachelor** ( $\beta_1 = 55$ )  $\rightarrow$  People with a Bachelor's degree spend **\$55 more** on average than High School graduates.
  - **Master** ( $\beta_2 = 100$ )  $\rightarrow$  People with a Master's degree spend **\$100 more** on average than High School graduates.
- 

### 2.0.5 Step 4. Example Predictions

Education	Prediction
High School	205
Bachelor	$205 + 55 = 260$
Master	$205 + 100 = 305$

---

### 2.0.6 Key Concept

If a variable has  $k$  categories, we create  $(k-1)$  dummy variables. The missing one becomes the **reference group**, and the coefficients for others measure differences relative to it.

## 3 Follow-up: So if marital status has these values

Single, married, divorced, separated, widowed, we make 4 dummies, right?

## 4 Answer

Exactly right!

If **Marital Status** has 5 categories —

Single, Married, Divorced, Separated, Widowed

then you create **4 dummy variables**, not 5. One category becomes the **reference (baseline)** group.

For example, if you choose **Single** as the baseline, the dummies could be:

Marital Status	Married	Divorced	Separated	Widowed
Single	0	0	0	0
Married	1	0	0	0
Divorced	0	1	0	0
Separated	0	0	1	0
Widowed	0	0	0	1

Then, in a regression model:

$$Y = \beta_0 + \beta_1(\text{Married}) + \beta_2(\text{Divorced}) + \beta_3(\text{Separated}) + \beta_4(\text{Widowed}) + \epsilon$$

#### Interpretation:

- = mean (or intercept) for *Single* individuals (the baseline).
- = difference between *Married* and *Single*.
- = difference between *Divorced* and *Single*, etc.

This setup avoids the **dummy variable trap** (perfect multicollinearity) — if you included all 5 dummies, one would be a linear combination of the others.

## 5 With stata, show minimal codes

## 6 Answer

Perfect — let's show a **minimal Stata example** illustrating dummy variables for a categorical predictor with 5 groups (“marital status”) and how Stata handles them automatically.

### 6.0.1 Example Goal

We'll regress **income** on **marital status**, which has 5 categories: **single**, **married**, **divorced**, **separated**, **widowed**.

### 6.0.2 Step 1. Create a small example dataset

```
clear
input str10 marital_status income
"single"      25
"single"      28
"married"     35
"married"     38
"divorced"    30
"divorced"    32
"separated"   26
"widowed"     20
"widowed"     22
"single"      27
end
```

---

### 6.0.3 Step 2. Encode the categorical variable

Convert the string variable into a labeled numeric variable that Stata can use with factor notation:

```
encode marital_status, gen(marital)
```

You can check the coding:

```
list marital_status marital
```

---

### 6.0.4 Step 3. Run the regression with dummy variables

```
regress income i.marital
```

- The prefix `i.` tells Stata to create **indicator (dummy) variables** automatically.
- Stata will **omit one category** as the **base (reference)** group — by default, the one with the smallest numeric code (e.g., “divorced” if coded as 1).

If you want a specific base category (say, “single”), you can set it explicitly:

```
regress income ib(single).marital
```

---

### 6.0.5 Step 4. Interpretation

Output will look roughly like:

Variable	Coefficient	Interpretation
__cons	26.7	Mean income for <b>single</b> individuals (baseline)
1.marital (married)	+10.5	Married earn 10.5 more on average than singles
2.marital (divorced)	+4.0	Divorced earn 4 more than singles
3.marital (separated)	-1.5	Separated earn 1.5 less than singles
4.marital (widowed)	-5.5	Widowed earn 5.5 less than singles

---

**In summary:**

- You only write `i.variable` — Stata makes and manages the dummy variables for you.
- It automatically chooses a baseline and avoids the dummy variable trap.
- You can override the base with `ib(category).variable`.

## 7 For this example, how can we make t tests to check differences of income by marital status

## 8 Answer

Excellent question — this is the natural follow-up to a regression with categorical predictors in **Stata**. Let’s continue with the same dataset and show **how to test income differences among marital groups** in different ways.

## 8.1 Situation Recap

We have:

```
regress income i.marital
```

where `marital` has 5 categories: `single`, `married`, `divorced`, `separated`, `widowed`.

---

### 8.2 1 Pairwise t-tests (simple group comparisons)

To directly compare **mean incomes** between two marital groups (e.g., “married” vs. “single”), you can use:

```
ttest income, by(marital)
```

However, `ttest` works **only if marital has two groups**. Since we have 5, you’d need to do pairwise comparisons like this:

```
ttest income if marital_status=="single" | marital_status=="married", by(marital_status)
```

This tests:

$$H : (\text{single}) = (\text{married})$$

You can repeat that for any pair you want (single–divorced, married–widowed, etc.).

---

### 8.3 2 Using regression and post-estimation tests (recommended)

Since we already ran a regression with dummy variables:

```
regress income i.marital
```

we can **test equality of any two groups’ means** using the `test` command. Suppose “single” is the baseline group (automatically omitted). Then:



```
test 1.marital = 0
```

tests whether the **mean income of married** differs from that of **single**.

To compare *two non-baseline groups*, say **married** vs **divorced**:

```
test 1.marital = 2.marital
```

That performs a **t-test** for

H : (married) = (divorced) which is equivalent to testing whether their mean incomes are equal.

---

### 8.4 3 Using margins for all pairwise comparisons

A very clean approach is:

```
regress income i.marital  
margins marital  
margins marital, pwcompare(effects) ///
```

Explanation:

- `margins marital` computes the adjusted mean income by marital status.
- `pwcompare(effects)` gives **pairwise differences** (like multiple t-tests) with p-values and confidence intervals.

You can also visualize it:

```
marginsplot
```

That shows group means with confidence intervals — great for teaching or reports.

---

#### 8.4.1 Summary Table

Approach	Command	Use Case
Simple two-group t-test	<code>ttest income if ...</code>	Quick pairwise manual test
Post-regression test	<code>test</code> <code>1.marital=2.marital</code>	Compare any two groups (most flexible)
Margins	<code>margins marital,</code> <code>pwcompare</code>	All pairwise comparisons, neatly summarized

**9 If I have two variables: gender (m, f), education (Primary, secondary, higher), can I perform t tests for crosstab, like male secondary vs female secondary, and all other combinations? Is it worth it?**

## 10 Answer

Fantastic question — and it shows you’re thinking like a statistician, not just a coder. Let’s unpack this carefully.

### 10.1 What you’re describing

You have:

- `gender` → 2 levels (M, F)
- `education` → 3 levels (Primary, Secondary, Higher) and say, a continuous variable `income`.

You want to compare combinations like:

- male-secondary vs female-secondary
- male-primary vs male-higher
- female-primary vs female-secondary
- etc.

So, in total,  $2 \times 3 = 6$  groups:

Gender	Education
Male	Primary
Male	Secondary
Male	Higher
Female	Primary
Female	Secondary
Female	Higher

---

## 10.2 Yes, you *can* do t-tests for pairs of these groups...

But: you'd be doing **many** pairwise comparisons —  $\binom{6}{2} = 15$  t-tests in total! That brings two problems:

1. **Inflated Type I error rate** — when you run many t-tests, you increase the chance of finding a “significant” difference by random chance.
2. **Inefficiency** — each test ignores the overall structure of the data.

So while it's *possible*, it's not the **best statistical approach**.

---

## 10.3 Better approach: Two-way ANOVA (Factorial ANOVA)

Instead of many t-tests, you can fit one model that covers everything:

```
anova income gender##education
```

This gives:

1. **Main effect of gender** → Do males and females differ in income overall?
2. **Main effect of education** → Do income levels differ by education overall?
3. **Interaction effect (gender#education)** → Does the effect of education differ between males and females?

This one command replaces all 15 t-tests, controlling error rates properly.

## 10.4 If you still want pairwise comparisons

After the ANOVA, you can get specific comparisons using:

```
margins gender#education
margins gender#education, pwcompare(effects)
```

This will:

- Report all pairwise mean differences between the 6 groups.
- Adjust p-values (Bonferroni, Sidak, etc.) if you request them, e.g.:

```
margins gender#education, pwcompare(effects adjust(bonferroni))
```

That's the **right** way to compare “male secondary vs female secondary” while accounting for multiple testing.

---

## 10.5 Summary

Option	What it does	Should you use it?
Manual t-tests	Pairwise comparisons between any two groups	Not ideal — too many tests, unadjusted error
Two-way ANOVA ( <code>anova income gender##education</code> )	Tests for main and interaction effects	Best global approach
<code>margins ... , pwcompare()</code>	Adjusted post-hoc comparisons	Great follow-up to ANOVA