

# CATEGORICAL VARIABLES AND INTERACTIONS

## Contents

1	Learning objectives	1
2	Dealing with categorical data	1
3	Interactions	2
3.1	Residual plots	4
4	Non-linear models	5

## 1 Learning objectives

- Understand how to add categorical predictors to a linear regression model with multiple predictors using dummy variables.
- Understand how to add interactions to linear regression models
- Be able to interpret regression coefficients when interactions are present
- Understand how to make residual plots and especially why we plot the predicted value of the response variable on the x axis.
- Understand how to build nonlinear models by adding “features”, and why for continuous predictors sines and cosines make good features.

## 2 Dealing with categorical data

- One situation in which models with multiple predictors frequently arises is when trying to predict a Y variable based on categorical predictors, such as race. In this case, we need to transform the categories into numerical values. For example, if there are two categories (e.g. YES and NO) we map our variable to 0 or 1. If we have 3 categories (e.g. White, Black, Other), we might first think to map them to 0, 1 and 2. This has a problem though: A change from 1 to 2 should not necessarily correspond to a change from 0 to 1. In other words, **there is no clear ordering of the x values**. Sometimes we refer to such predictors and qualitative rather than quantitative, since they express a quality of our data points instead of a numerical quantity.
- To address this issue, we create dummy variables. In particular, In order to take a categorical variable and transform it into a set of indicator variables in python, we use the python function `get_dummies`. The usage of this is illustrated in the following example.

**Example 1** (Racial disparities in earnings). *Here we will fit the earnings data to a model with race as a predictor. In particular, we want to know: What is the association between race and earnings among adults in the US? We will start with a model using only race as a predictor. One way to approach this would be to simply use a binary predictor and consider only 2 race categories (e.g. White and non-White). This is limiting though. Instead, we can create a variable for each race category we are interested in. In the dataset there are 4 race categories (not sure why these 4, but that's what we'll work with)*

{Black, White, Hispanic, Other}

*In principle, we could create a binary variable for each one (these are what we call dummy variable), to obtain a model like*

$$Y = \beta_0 + \beta_{\text{black}}X_{\text{black}} + \beta_{\text{hispanic}}X_{\text{hispanic}} + \beta_{\text{other}}X_{\text{other}} + \beta_{\text{white}}X_{\text{white}} + \epsilon$$

*This is problematic though, since at least one of the predictors above MUST be 1. This means that the first 3 of the predictors are perfectly correlated with the other one. By the default, python will drop the first predictor (in alphabetical order), leaving us with the model*

$$Y = \beta_0 + \beta_{\text{hispanic}}X_{\text{hispanic}} + \beta_{\text{other}}X_{\text{other}} + \beta_{\text{white}}X_{\text{white}} + \epsilon.$$

*Question: Fit the data to the model above. What is the expected disparity in earnings between someone who is white and someone who is hispanic.*

*Solution: See colab notebook. To answer the question posed above, we begin with the interpretations of the regression coefficients. In terms of conditional expectation, these are*

$$\begin{aligned}\beta_{\text{white}} &= E[Y|X_{\text{white}} = 1, X_{\text{hispanic}} = X_{\text{other}} = 0] - E[Y|X_{\text{white}} = 0, X_{\text{hispanic}} = X_{\text{other}} = 0] \\ &= E[Y|\text{someone is white}] - E[Y|\text{someone is black}] \approx 4.9 \\ \beta_{\text{hispanic}} &= E[Y|X_{\text{hispanic}} = 1, X_{\text{white}} = X_{\text{other}} = 0] - E[Y|X_{\text{hispanic}} = 0, X_{\text{white}} = X_{\text{other}} = 0] \\ &= E[Y|\text{someone is hispanic}] - E[Y|\text{someone is black}] \approx -0.7\end{aligned}$$

*Our goal however is to compute*

$$\begin{aligned}&E[Y|\text{someone is white}] - E[Y|\text{someone is hispanic}] \\ &= E[Y|X_{\text{white}} = 1, X_{\text{hispanic}} = 0, X_{\text{other}} = 0] - E[Y|X_{\text{white}} = 0, X_{\text{hispanic}} = 1, X_{\text{other}} = 0] \\ &= \beta_0 + \beta_{\text{white}} - \beta_0 - \beta_{\text{hispanic}} \\ &= \beta_{\text{white}} - \beta_{\text{hispanic}}\end{aligned}$$

### 3 Interactions

- The important assumption of the multiple predictor regression models we have seen so far is that the “effect” of one predictor does not depend on the value of the other. Here are some examples where this could be violated:
  - The difference in test scores between kids whose mothers did and did not go to high school depends on their mother’s score on the iq test.
  - The association between earnings and height depends on gender, e.g. being taller tends to give men a larger advantage than women
  - The effect of a drug for treating covid depends on whether someone has had covid before.

We call the dependencies described above interactions. It turns out it is possible to include interactions within the regression modeling framework we have already introduced, as is illustrated by the following example.

- Let’s work in the case of two predictors. If you’d like, you can refer to Example 3 to make this math more concrete and always replace  $X_1$  with  $X_{\text{hs}}$  and  $X_2$  with  $X_{\text{iq}}$ . In each of the cases above, what is going on is that the slope of  $\mathbb{E}[Y|X_1, X_2]$  vs.  $X_1$  for fixed  $X_2$  is not a constant,  $\beta_1$ , but rather something that depends on  $X_2$ . The simplest way to account for this is to let effect of  $X_2$  on the slope be linear in  $X_2$ , so instead of  $\beta_1$  being the slope of  $\mathbb{E}[Y|X_1, X_2]$  vs.  $X_1$  we would have  $\beta_1 + \beta_{1,2}X_2$  be this slope, but this leads to a regression model with a nonlinear term:

$$Y = \beta_0 + (\beta_1 + \beta_{1,2}X_2)X_1 + \beta_2X_2 = \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_{1,2}X_1X_2 + \epsilon$$

We call  $X_1X_2$  an interaction term, but Yuck! This isn’t linear anymore and the regression coefficients are also don’t have the same interpretation. How do we deal with this?

- The idea is that we can create a new predictor from our already existing predictors which accounts for the interactions. To this end, consider the two predictor regression model written now as

$$Y = \beta'_0 + \beta'_1X_1 + \beta'_2X_2 + \epsilon.$$

I’m again using the ‘ to distinguish this model from an expanded model with a new predictor. The new predictor which accounts for the interaction will be  $X_3 = X_1X_2$ ! Note that  $X_3$  is going to be correlated

with both  $X_1$  and  $X_2$ , but crucially it will not be perfectly correlated unless  $X_1$  and  $X_2$  are with each other. The new model is

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{1,2} X_3 + \epsilon$$

Or written as a conditional distribution

$$Y|X \sim \text{Normal}(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{1,2} X_3, \sigma^2).$$

- Now how do we interpret the regression coefficients in the new model? If we write out the usual formula for the regression coefficient with no interactions, we get

$$\begin{aligned} \mathbb{E}[Y|X_1 = x + 1, X_2] - \mathbb{E}[Y|X_1 = x, X_2] \\ &= (\beta_1(x + 1) + \beta_2 X_2 + \beta_3(x + 1)X_2) - (\beta_1 x + \beta_2 X_2 + \beta_{1,2} x X_2) \\ &= \beta_1 + \beta_{1,2} X_2 \end{aligned}$$

Here I've replaced  $X_3$  with  $X_1 X_2$ . In words,  $\beta_{1,2}$  is the additional difference in the regression slope between  $Y$  and  $X_1$  with all other predictors fixed when  $X_2$  is changed by one unit. If  $\beta_{1,2}$  is positive (resp. negative) then increasing  $X_2$  has a tendency to increase (resp. decrease) the difference between the conditional averages of  $Y$ , thus, the additional predictor  $X_3 = X_1 X_2$  allows us to capture an interaction wherein the association between  $X_1$  and  $Y$  depends on  $X_2$ . We can do the same calculation with  $X_1$  fixed and  $X_2$  changed to obtain

$$\mathbb{E}[Y|X_1 = x + 1, X_2] - \mathbb{E}[Y|X_1 = x, X_2] = \beta_2 + \beta_3 X_1$$

Now as always, let's make sure we understand how this translates into code and also I think the visualizations in this example are helpful.

**Example 2** (Visualizing interactions). Consider the regression model with interactions term  $X_3 = X_1 X_2$  defined above, assuming that

$$X_1 \sim \text{Normal}(0, 1)$$

$$X_2 \sim \text{Bernoulli}(1/2)$$

Assume  $\beta_1 = 1.2, \beta_2 = 0$  and  $\beta_3 = -1$ .

Question: What are the slopes of  $E[Y|X_1, X_2 = 0]$  vs.  $X_1$  and  $E[Y|X_1, X_2 = 1]$  vs.  $X_1$ ? Plot these in colab.

**Example 3** (Interaction in test score model). Here we once again consider the test score data with high school education and IQ as predictors. In particular, we will fit the model

$$Y = \beta_0 + \beta_{\text{hs}} X_{\text{hs}} + \beta_{\text{iq}} X_{\text{iq}} + \beta_{\text{iq,hs}} X_{\text{iq}} X_{\text{hs}}$$

Question:

- What are the values and interpretations of the regression coefficients in the new model?
- What do the results tell us about the "effect" of IQ on test scores and how it is related to high school education?

Solutions: The output of statsmodels is

```
> =====
>
>               coef      std err          t      P>|t|      [0.025      0.975]
> -----
> const         -11.4820     13.758      -0.835     0.404     -38.523     15.559
> mom_hs          51.2682     15.338       3.343     0.001      21.122     81.414
> mom_iq           0.9689      0.148       6.531     0.000       0.677      1.260
> hs_iq          -0.4843      0.162      -2.985     0.003     -0.803     -0.165
```

- The values and interpretations are as follows

- $\beta_0 \approx -11$  does not have a clear interpretation since it doesn't make sense to have 0 IQ.
  - $\beta_{iq} \approx 1$  is the expected signed difference in scores between students whose IQ differs by 1 point and whose mothers did not go to high school.
  - $\beta_{hs} \approx 51$  is the expected difference between scores of students whose mothers did and did not go to high school when IQ is zero – **much like  $\beta_0$ , this doesn't really make sense, so we don't have a clear interpretation of  $\beta_{hs}$**
  - $\beta_{hs,iq} \approx -0.5$  is the expected increase between the group of students whose mothers attended high school and those that did not in the difference between test scores of students whose mothers iq differs by one point. In other words, it is the increase in regression slope of scores vs. iq between the two high school groups. It is easier to interpret  $\beta_{iq} + \beta_{hs,iq}$  which is the expected signed difference in scores between students whose IQ differs by 1 point and whose mothers did attend high school (this is the same as the interpretation of  $\beta_{iq}$  for student's whose mothers attended high school).
  - It appears that the association between IQ and test scores is weaker among students whose mother's attended high school. This makes intuitive sense. If someone attended high school the subtle differences in cognitive ability that IQ is supposed to measure are not as relevant.
- In the example above we saw that adding an interaction term can make the interpretation of the regression coefficients a bit clunky, even those regression coefficients that are not involved in an interaction. In particular, we ran into the problem that the interpretation of  $\beta_{iq}$  was lost. To remedy this, we can define the centered predictor

$$\Delta_{iq} = X_{iq} - \bar{X}_{iq}$$

and now we fit the model

$$Y = \beta_0 + \beta_{hs}X_{hs} + \beta_{iq}\Delta_{iq} + \beta_{hs-iq}X_{hs}\Delta_{iq} + \epsilon$$

Now  $\beta_1$  has the interpretation as the average difference in scores among students whose mothers have the average iq. You can also standardize the predictor to get around the awkward interpretation of the regression coefficients.

### 3.1 Residual plots

- We address the question of how we might identify when it is appropriate to add an interaction term to a model. In this case of two predictors we can easily visualize the data by making plots of the  $Y$  vs.  $X_1$  slopes for different  $X_2$  values (especially when one predictor is binary). With many predictors, it becomes less clear what plot might reveal some hidden interaction terms. We could of course go on adding every possible interaction term, but with many predictors this becomes in practical and leads to overfitting (as we will soon discuss).
- The basic idea of residual plots is that by plotting the difference between the observed  $y$  values and the prediction of the  $E[Y|X]$ , or

$$\epsilon_j = Y_j - \sum_k^K \hat{\beta}_k X_{k,j}$$

where  $X_{k,j}$  is the value of predictor  $i$  at the  $j$ th data point. Our goal is to plot  $\epsilon_j$  in a such a way that disagreement between our model and the data is revealed by this plot. In the instance of a single-predictor, we can simply plot  $\epsilon_j$  as a function of the predictor  $X$ . If we notice that the residuals do not appear to follow a normal distribution, or that the variance and mean change, then we should be skeptical. The question is: **When we have multiple predictors, what do we plot on the horizontal axis?** The answer is to plot  $\epsilon_j$  as a function of the predictors value of  $E[Y|X]$ ; that is  $\epsilon_j$  vs.  $\hat{y}_j = \sum_i^K \hat{\beta}_i X_{i,j}$ . The following examples is supposed to help us understand why.

**Example 4** (Residual plots). Consider the regression model with two predictors and suppose the true parameter values are  $\beta_0 = 0, \beta_1 = 0.2, \beta_2 = 20$  and  $\sigma = 1$ .

Question: Compare two ways of plotting the residuals

- (a) Plot  $\epsilon_i$  as a function of  $\sum_k^K \hat{\beta}_k X_{k,i}$ .  
 (b) Plot  $\epsilon_i$  as a function of  $Y_i$ . Why does there appear to be a bias towards high values of  $\epsilon_i$  for large  $Y_i$  that is not present in the first plot?

*Solution: See colab notebook*

- Let's take a closer look at the mathematics underlying the residual plot. Note that

$$(1) \quad \epsilon_j \approx Y_j - \underbrace{E[Y|(X_1, \dots, X_K) = (X_{1,j}, \dots, X_{K,j})]}_{\approx \sum_k^K \hat{\beta}_k X_{k,j}}$$

where  $K$  is the number of predictors. Thus, the distribution of  $\epsilon_j$  is approximately

$$\epsilon_j \sim \text{Normal}(0, \sigma^2).$$

This tells us how the points should be distributed in the vertical direction. It **does not** say anything about the distribution of points in the horizontal direction, which is determined by the distribution of the predictors. Therefore, we expect a plot which is symmetric around the line  $\epsilon_j$  for all values of  $\hat{y}_j$  (our predicted values of  $Y$ ), but any distribution in the horizontal direction is okay.

Now compare this to what would happen if we plotted  $Y_j$  on the horizontal axis, not  $\hat{y}_j$ . In this case, based on Equation 1  $\epsilon_j$  and  $Y_j$  are correlated. This is why we see a bias of the residuals for small/large  $Y_j$  in Example 4.

## 4 Non-linear models

- Here, we discuss how to build more complex models and directly access their predictive power on out-of-sample data. In the context of interactions, we already saw how a model can be extended by defining a new predictor  $X_3 = X_1 X_2$ . The more general idea that we can define a new predictor which is a function of the other predictors allows us to develop very complex and flexible models which nonetheless can be analyzed within linear regression framework. Here, we will formalize this, beginning with the case of a single predictor.

In general, the linear regression framework allows us to fit models of the form consider the model

$$(2) \quad Y|X \sim \text{Normal}(f(X), \sigma^2)$$

provided we can express  $f(X)$  as a linear combinations of nonlinear functions of  $X$ . What I mean by this is that we can find function  $\phi_1(x), \dots, \phi_K(x)$  such that

$$f(X) = \sum_{i=1}^K \beta_i \phi_i(X)$$

The functions  $\phi_i(X)$  are often referred to as basis functions, or features in machine learning lingo. We can think of each  $\phi_i(X)$  as a new predictor.

**Example 5** (Simulating a nonlinear model). *Consider the conditional Gaussian model given in Equation 2 with*

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2$$

*This is simply a linear model once we define the new predictors*

$$\begin{aligned} X_1 &= \phi_1(X) = X \\ X_2 &= \phi_2(X) = X^2. \end{aligned}$$

*Question:*

- (a) Generate data from this model  
 (b) Fit the data to the model using `statsmodels`.

*Solution: See colab notebook*

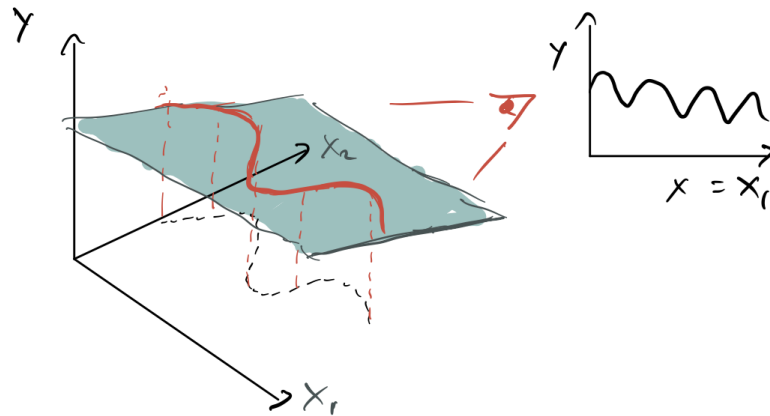


Figure 1. An illustration of how a nonlinear dependence on our predictor can be incorporated into the linear modeling framework by adding a feature.

- Which functions  $\phi$  should we use? The answer of course depends on the problem at hand. For example, we might know something about the physics of the data we are modeling. In some cases, we may select the  $\phi$  so that the parameters  $a_i$  have clear interpretations (as is the case in linear regression model). The following illustrates such an example.

**Example 6** (Mauna Kea Data). *Solution: See colab notebook*

- We now understand that within the linear regression framework we can fit very complicated nonlinear data. However, we need to be careful when adding new features to the model. The following example will illustrate how it is possible to “overshoot” and make our too complex. In the next set of notes we will dive much deeper into this, bridging the gap between machine learning and statistics.

**Example 7** (Models with very large numbers of parameters). *Consider the model*

$$Y|X \sim \text{Normal}(f(X), \sigma^2)$$

$$f(X) = 1 - X^2 + 0.9X^3$$

*This is a linear regression model with the features  $X_1 = X^2$  and  $X_2 = X^3$ . We will pretend we have some data from this model, but not only do we lack knowledge of the coefficients, we in-fact lack knowledge of what the function  $f(X)$  is. This, we are going to fit the data to a polynomial regression model*

$$f(X) = \beta_0 + \beta_1 X + \beta_2 X^2 + \dots + \beta_K X^K$$

*The question is which value of  $K$  to select, since we don't know a-priori that  $K = 3$ .*

Question: *How does the predicted function  $f(x)$ , which we will denote  $\hat{f}(x)$  compare to the true function? How does this depend on the number of features we include in our model?*