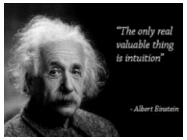
Data Transformations

Transformation #1:
Categorical to Dummy Variables



Categorical Variables: Intuition

- Categorical variables are **not quantitative**, so the assumption of linearity does not hold. But a very simple **transformation** using **dummy variables** solves this problem.
- For example, if you have a categorical variable called **"State"**, it will contain a text value of 2 upper case letters with one of 50 values corresponding to one of the 50 U.S. states.
- If you want to predict housing prices, the state where the house is located will probably make a difference. So, how do we **model** the **effect of state** location?
- The answer is to **convert** the categorical variable "State" into 50 **dummy variables**, each named after each state, with a value of 1 if the house is in that state and 0 otherwise.
- For example, the variable **MD** would have a 1 for all houses in the state of Maryland and 0 for any of the other states.
- We would then have **50** dummy (quantitative) variables to model

The Dummy Variable Trap

- This is a well-known problem when you convert a categorical variable into various **“mutually exclusive”** dummy variables.
- For example, if you have a categorical variable called “LocationType” and it has one of **three possible values** (Urban, Suburban and Rural) we can create 3 dummy variables called Urban, Suburban and Rural, respectively.
- If LocType = “Urban” → **Urban = 1**; 0 otherwise
  If LocType = “Suburban” → **Suburban = 1**; 0 otherwise
  If LocType = “Rural” → **Rural = 1**; 0 otherwise
- However, these three dummy variables are **mutually exclusive**, so if Urban = Suburban = 0, then Rural must be 1.
- That is, the value in any of these variables is fully dependent on the other 2
- Including all 3 variables in a regression model will not only violate the **assumption of independence**, but will also create **infinite multicollinearity** and **infinite standard errors**

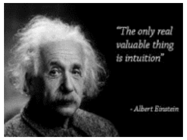
Modeling Categorical Variables

- If the **dependent variable** is categorical, you need to employ a **classification method** (later in the semester) – e.g., logistic, probit, linear discriminant analysis, decision trees, etc.
- If an **independent variable** is categorical, you can convert it to **N dummy variables**, where N is the number of categories in the data for that variable.
- Because of the dummy variable trap, you can only **model N-1** dummy variables.
- The variable **left out** is called the **"baseline"** or **"reference"** variable (e.g., MD)
- The regression **intercept** represents the effect of the reference variable, when all other variables are 0 (e.g., house value for houses in MD – i.e., all **included** dummy variables **= 0**).
- A **coefficient** for any other variable (e.g., VA) represents the **effect change** when switching the reference variable for that variable (e.g., the difference in house value in VA, compared to MD, all else equal)

Categorical Variables in R

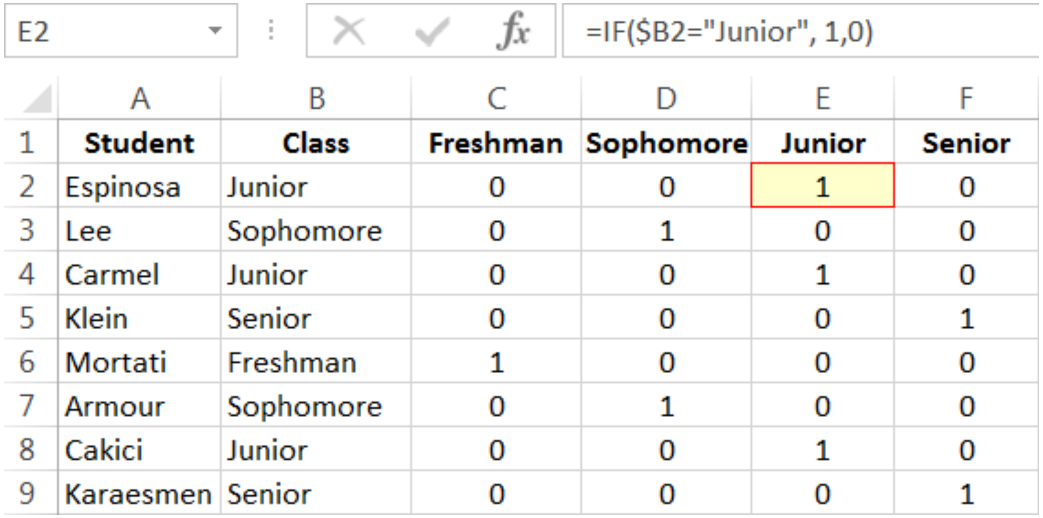
Like most modern statistical software:

- If you **include** the **categorical variable** in the model, R will automatically convert it into the necessary **N-1 dummy** variables
- By default, R will **leave out** the dummy variable corresponding to the first category **alphabetically**.
- If this is not what you want, you can use the **relevel()** function in R to specify which **reference** dummy variable to **leave out**.
- If you **hand-code** the dummy variables, you need to remember to **leave out** the **reference** variable of your choice
- However, **R** will notice if you don't and will **leave one** dummy variable **out** on your behalf, usually the first one specified, to avoid the **dummy variable trap**.



Categorical to Dummy: Intuition

- As we discussed earlier, a categorical variable with N distinct categories can be transformed into N dummy variables
- For example, this model:
  StudentRetention = f(GPA, Major, Class)
  Has 1 continuous and 2 categorical variables
  Class can be: Freshman, Sophomore, Junior, Senior, Graduate
- We can create a dummy variable Freshman = 1 if the student is a freshman, 0 otherwise, and so on
- Because of the dummy variable trap, we can only model 3 of the 4 variables (i.e., N-1 variables)

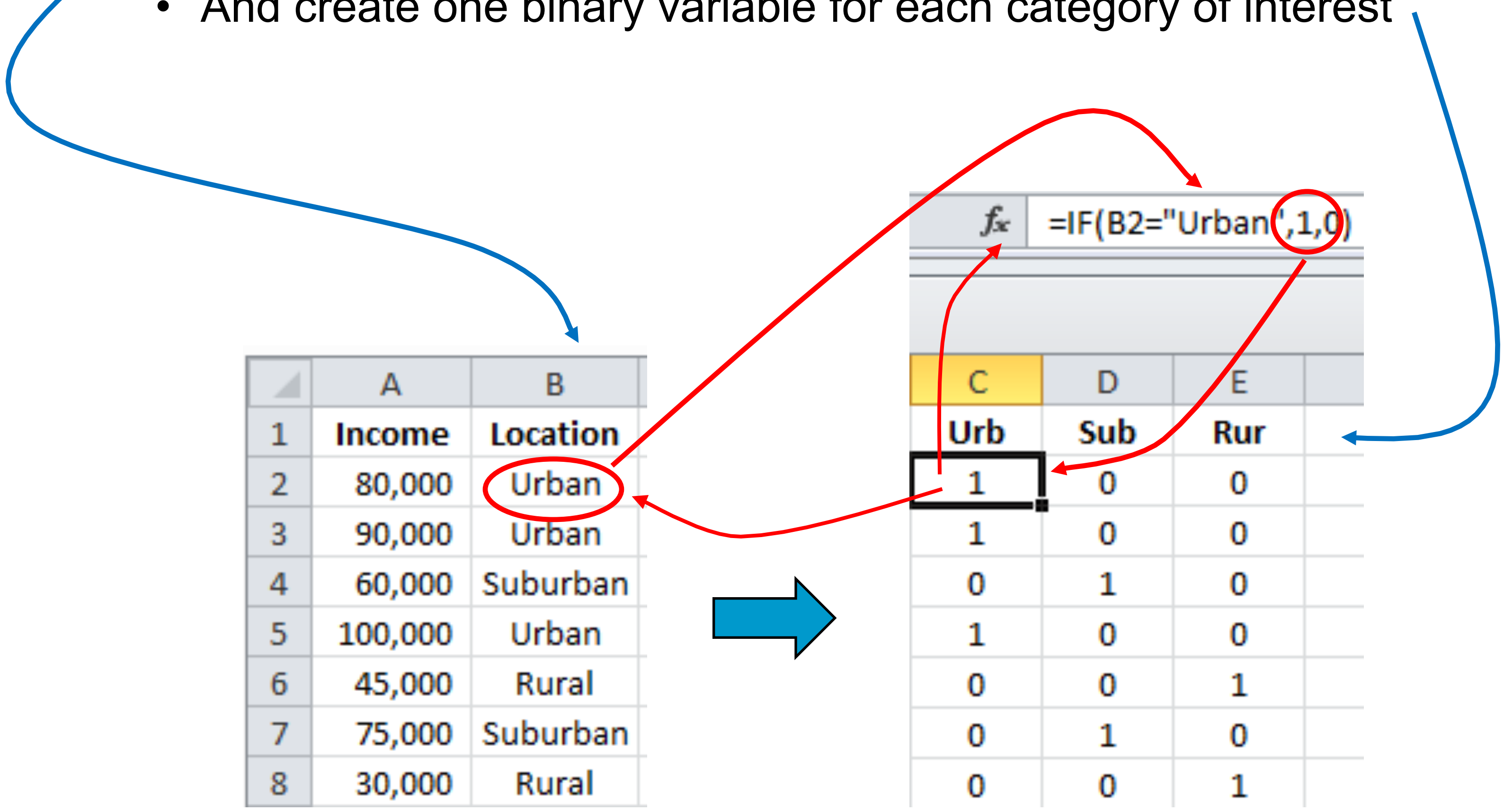


E2 =IF($B2="Junior", 1,0)

	A	B	C	D	E	F
1	**Student**	**Class**	**Freshman**	**Sophomore**	**Junior**	**Senior**
2	Espinosa	Junior	0	0	1	0
3	Lee	Sophomore	0	1	0	0
4	Carmel	Junior	0	0	1	0
5	Klein	Senior	0	0	0	1
6	Mortati	Freshman	1	0	0	0
7	Armour	Sophomore	0	1	0	0
8	Cakici	Junior	0	0	1	0
9	Karaesmen	Senior	0	0	0	1

Converting Categorical to Binary

- Look at the categorical values you are interested in analyzing
- And create one binary variable for each category of interest



	A	B
1	Income	Location
2	80,000	Urban
3	90,000	Urban
4	60,000	Suburban
5	100,000	Urban
6	45,000	Rural
7	75,000	Suburban
8	30,000	Rural

fx =IF(B2="Urban",1,0)

C	D	E
Urb	Sub	Rur
1	0	0
1	0	0
0	1	0
1	0	0
0	0	1
0	1	0
0	0	1

R Tips

`levels{dataFrame$variableName}` → A useful function to display the unique values of a categorical variable. If there are n values, the categorical data transformation will automatically create n-1 dummy variables, leaving out the first category alphabetically

`attach(dataFrame)` → Load a data set into memory

`contrasts(ShelveLoc)` → Show how the respective categorical dummy variables were coded

```
dataFrame$variableName =
     relevel(dataFrame$variableName, ref=“otherValue")
```

→ Use the `relevel()` function to select a different category value as the reference or baseline dummy variable to leave out

Note: in R, when modeling categorical (**factor**) variables (e.g., `LocType`) with values like `“Rural”`, `“Suburban”` and `“Urban”`, R will create 3 dummy variables, one for each of these values and append the value to the dummy variable name → e.g., `LocTypeRural` (1 if Rural, 0 otherwise), `LocTypeSuburban` and `LocTypeUrban`

Kogod School
of
Business