## The dummy variable trap for Linear Regression

Suppose we have categorical variable `Sex` with categories (discrete values) $\{\text{Female}, \text{Male}\}$

One Hot Encoding represents each value as a vector $\mathbf{Is}$ of length 2

- Replace feature `Sex`
- with the two binary valued *indicator* variables
    - $\mathbf{Is}_\text{Female}, \mathbf{Is}_\text{Male}$

So for example $i$

- where $\mathrm{Sex}^{\mathbf{(i)}} = \mathrm{Female}$
- $\mathbf{Is}^{\mathbf{(i)}}_{\mathrm{Female}} = 1,$ $\mathbf{Is}^{\mathbf{(i)}}_{\mathrm{Male}} = 0$

And for example $i'$

If we were to use the OHE in a Linear Regression, the design matrix might look something like

$$\mathbf{X}' = \begin{array}{c} \left(\begin{array}{cccc} \textbf{const} & \textbf{Is}_{\text{Female}} & \textbf{Is}_{\text{Male}} & \dots \\ 1 & 1 & 0 & \dots \\ \vdots & & & \\ 1 & 0 & 1 & \dots \\ \vdots & & & \end{array}\right) \end{array} \begin{array}{c} \\ \text{Female} \\ \vdots \\ \text{Male} \\ \vdots \end{array}$$

Note that, for every example $i$

$$\mathbf{Is}^{\mathbf{(i)}}_{\text{Female}} + \mathbf{Is}^{\mathbf{(i)}}_{\text{Male}} = 1$$

When a categorical variable with categories from the set $C$ is encoded with OHE

$$\sum_{c \in C} \mathbf{Is}^{\mathbf{(i)}}_{c} = 1$$

Also note that $\mathbf{const^{(i)}} = 1$ for every example $i$ so

$$\sum_{c \in C} \mathbf{Is}_c^{\mathbf{(i)}} = 1 = \mathbf{const^{(i)}}$$

That is, for each example

- The linear combination of some features (i.e., the set of indicator variables for a categorical variable)
- Is exactly equal to some other feature (i.e, the constant)

Such a situation is called *Perfect Multi-Colinearity*.

Multi-colinearity (either Perfect or Imperfect) poses mathematical difficulties for Linear Regression and must be eliminated

The problem manifests itself in Linear Regression as

- Some variables with huge positive parameter values (e.g., $\Theta_{\mathbf{Is}_{\text{Female}}}$ )
- And other variables with huge (offsetting) negative parameter values (e.g., $\Theta_{\mathbf{Is}_{\text{Male}}}$ ).

## Multi-colinearity

- Arises when categorical variables are One Hot Encoded, i.e., added dummy/indicator variables
- Is **not necessarily** a difficulty for models other than Linear Regression
- Is called the *Dummy Variable Trap for Linear Regression*

It may appear that the simplest way to avoid multi-colinearity is to drop the constant feature $\mathbf{const}$

This will work if there is a *single* categorical variable but consider what happens if there is a second categorical varialbe

- `Pclass` with values in $\{\text{First}, \text{Second}, \text{Third}\}$
  $(\mathbf{Is}_{\text{Female}} + \mathbf{Is}_{\text{Male}}) = (\mathbf{Is}_{\text{First}} + \mathbf{Is}_{\text{Second}} + \mathbf{Is}_{\text{Third}}) = 1$

Thus the indicator variables encoding `Sex` are multi-colinear with the binary variables encoding `Pclass`.

The better solution is

- To retain the constant $\mathbf{const}$ variable
- Encode a categorical variable having category values in $C$ using $(||C|| - 1)$ binary indicator variables
  - elminate the indicator variable for a single category in each class

The Linear Regression equations changes from

$$\begin{aligned}\mathbf{y} &= \Theta_0 * \mathbf{const} + \Theta_{\mathbf{Is}_{\text{Female}}} * \mathbf{Is}_{\text{Female}} + \Theta_{\text{Male}} * \mathbf{Is}_{\text{Male}} \\ \text{to} & \\ \mathbf{y} &= \Theta'_0 * \mathbf{const} + \Theta'_{\mathbf{Is}_{\text{Female}}} * \mathbf{Is}_{\text{Female}}\end{aligned}$$

This will eliminate multi-colinearity, but where did the indicator (binary variable) for the missing category go ?

Answer: the constant !

$$\begin{aligned} \Theta'_0 &= \Theta_0 + \Theta_{\text{Male}} \\ \Theta'_{\text{Female}} &= \Theta_{\text{Female}} - \Theta_{\text{Male}} \end{aligned}$$

To convince yourself of this, consider the original equation for examples with either of the two values for Sex

$$\begin{aligned} \text{Sex}^{(\mathbf{i})} = \text{Female}: &\quad \mathbf{y}^{(\mathbf{i})} = \Theta_0 + \Theta_{\mathbf{Is}_{\text{Female}}} \\ \text{Sex}^{(\mathbf{i})} = \text{Male}: &\quad \mathbf{y}^{(\mathbf{i})} = \Theta_0 + \Theta_{\mathbf{Is}_{\text{Male}}} \end{aligned}$$

and the corresponding changed equations

$$\begin{aligned} \text{Sex}^{(\mathbf{i})} = \text{Female}: \quad \mathbf{y}^{(\mathbf{i})} &= \Theta'_0 + \Theta'_{\mathbf{Is}_{\text{Female}}} \\ &= (\Theta_0 + \Theta_{\mathbf{Is}_{\text{Male}}}) + (\Theta_{\text{Female}} - \Theta_{\mathbf{Is}_{\text{Male}}}) \\ &= \Theta_0 + \Theta_{\mathbf{Is}_{\text{Female}}} \\ \text{Sex}^{(\mathbf{i})} = \text{Male}: \quad \mathbf{y}^{(\mathbf{i})} &= \Theta'_0 \\ &= \Theta_0 + \Theta_{\mathbf{Is}_{\text{Male}}} \end{aligned}$$

Identical !

What has effectively happened

- The new intercept
    - Captures the "base" contribution (e.g. $\Theta_{\mathbf{Is}_{\text{Male}}}$) of the missing category
- The remaining binary variables
    - Capture the incremental contribution (over the base contribution $\Theta_{\mathbf{Is}_{\text{Male}}}$) of the example being different than the base category (e.g., having Sex = $\text{Male}$)

That is:

- $\Theta_{\mathbf{Is}_{\text{Female}}}$: absolute contribution to $\hat{\mathbf{y}}$ for being Female rather than Male
- $\Theta'_{\mathbf{Is}_{\text{Female}}}$: *incremental* contribution to $\hat{\mathbf{y}}$ for being Female
    - Over and above the contribution for being Male

## Titanic example: why no problem when we added OHE for `Pclass` ?

When we encoded `Pclass` using OHE, we didn't drop one category for the feature.

How come this didn't manifest itself as large, offsetting positive/negative parameter values ?

- Answer: `LogisticRegression` in `sklearn` defaults to using regularization in the loss function.

Regularization skirts the issue by enforcing a constraint that restricts large values for parameters.

By turning the parameter value of one indicator in a class to $0$, we effectively eliminate 1 indicator and avoid perfect collinearity.

## Titanic example: why no problem when feature **Sex** was not OHE ?

Recall our first pass at solving the Titanic classification problem

- Categorical variable Sex was encoded as the single binary variable with values 0/1
- Rather than 2 binary indicator variables $\mathbf{Is}_{\text{Female}}, \mathbf{Is}_{\text{Male}}$

Essentially: our representation was equivalent to having dropped one value for the variable Sex

**Bottom line**

Some models (Linear Regression and Logistic Regression by extension) may encounter problems with OHE of features.

By luck or design, we avoided any potential Dummy Variable Trap issues in the Titanic example.

But it's much better to be smart than lucky: do it the right way !

`sklearn` makes this easy for you

- `OneHotEncoder` transformer has optional argument `drop` that can drop one category per feature

In [4]:
```
print("Done")
```
```
Done
```