Feature enginerring

**Label Encoding:**

Categorical variable having Oordinal categorical values

This approach requires the category column to be of ‘category’ datatype. By default, a non-numerical column is of ‘object’ type

from sklearn.preprocessing import LabelEncoder

# creating instance of labelencoder  
labelencoder = LabelEncoder()

Dataset['NewColumn’] = labelencoder.fit\_transform(Dataset ['Feature’])

Label will be assigned to categories in hierary/ order.

# to visualise al the columns in the dataframe

pd.pandas.set\_option('display.max\_columns', None)

**One-Hot Encoder:** Though label encoding is straight but it has the disadvantage that the numeric values can be misinterpreted by algorithms as having some sort of hierarchy/order in them. This ordering issue is addressed in another common alternative approach called ‘One-Hot Encoding’. In this strategy, each category value is converted into a new column and assigned a 1 or 0 (notation for true/false) value to the column.

from sklearn.preprocessing import OneHotEncoder

# creating instance of one-hot-encoder  
enc = OneHotEncoder(handle\_unknown='ignore')

# passing bridge-types-cat column (label encoded values of bridge\_types)

enc\_df = pd.DataFrame(enc.fit\_transform(bridge\_df[['Bridge\_Types\_Cat']]).toarray())