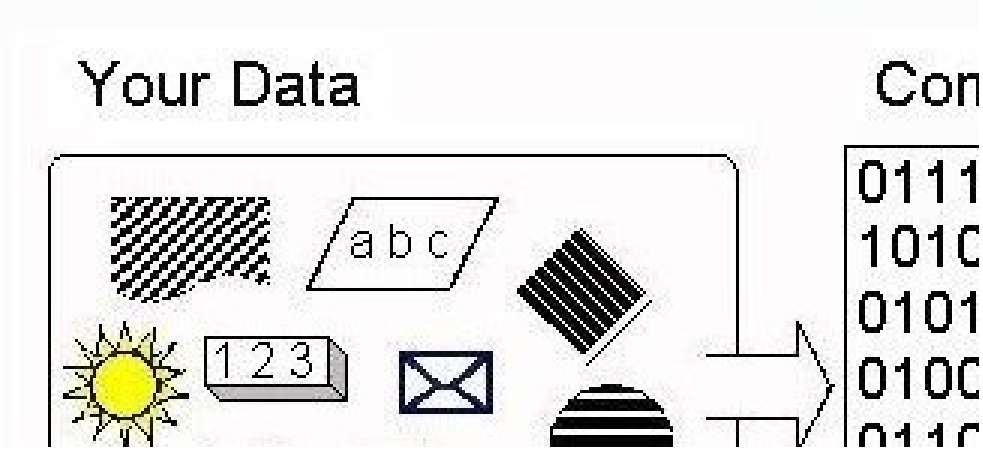
ENCODING OF CATEGORICAL VARIABLES

What is Categorical Encoding?

Typically, any structured dataset includes multiple columns ,a combination of numerical as well as categorical variables. A machine can only understand the numbers. It cannot understand the text.



That's primarily the reason we need to convert categorical columns to numerical columns so that a machine learning algorithm understands it. This process is called **categorical encoding**.

Different Approaches to Categorical Encoding

So, how should we handle categorical variables? As it turns out, there are multiple ways of handling Categorical variables,two most widely used techniques:

1. **Label Encoding**
2. **One-Hot Encoding**

Label Encoding

Label Encoding is a popular encoding technique for handling categorical variables.
In this technique, each label is assigned a unique integer based on alphabetical ordering.

Let's see how to implement label encoding in Python using the scikit-learn library and also understand the challenges with label encoding.

```
#importing the libraries
import pandas as pd
import numpy as np

#reading the dataset
df =pd.read_csv("Salary.csv")
```

Output:

Country	Age
India	44
US	34
Japan	46

```
<class 'pandas.core.frame.
RangeIndex: 14 entries, 0
Data columns (total 3 colu
Country    14 non-null obj
Age        14 non-null int
```

Understanding the datatypes of features:

print df.info

As you can see here, the first column, Country, is the categorical feature as it is represented by the **object data type** and the rest of them are numerical features as they are represented by *int64*.

```
from sklearn import preprocessing

# label_encoder object knows how to understand
word labels.
label_encoder = preprocessing.LabelEncoder()

# Encode labels in column 'Country'.

df['Country']=
label_encoder.fit_transform(df[‘Country'])
print(df.head())
```

Country	Age
0	44
2	34
1	46

As you can see here, label encoding uses alphabetical ordering. Hence, India has been encoded with 0, the US with 2, and Japan with 1.

Challenges with Label Encoding

In the above scenario, the Country names do not have an order or rank. But, when label encoding is performed, the country names are ranked based on the alphabets. Due to this, there is a very high probability that the model captures the relationship between countries such as India < Japan < the US.

This is something that we do not want! So how can we overcome this obstacle? Here comes the concept of **One-Hot Encoding**.