**Dummy Variables**

Instructions:

Please share your answers filled inline in the word document. Submit Python code and R code files wherever applicable.

Please ensure you update all the details:

**Name: ANANDAKRISHNAN K V**

**Batch Id: 19042021**

**Topic: Preliminaries for Data Analysis**

**Problem Statement:**

Data is one of the most important assets. It is often common that data is stored in distinct systems with different formats and forms. Non-numeric form of data makes it tricky while developing mathematical equations for prediction models. We have the preprocessing techniques to make the data convert to numeric form. Explore the various techniques to have reliable uniform standard data, you can go through this link:

<https://360digitmg.com/mindmap-data-science>

1. Prepare the dataset by performing the preprocessing techniques, to have the all the features in numeric format.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Animals | Gender | Homly | Types |
| 1 | Cat | Male | Yes | A |
| 2 | Dog | Male | Yes | B |
| 3 | Mouse | Male | Yes | C |
| 4 | Mouse | Male | Yes | C |
| 5 | Dog | Female | Yes | A |
| 6 | Cat | Female | Yes | B |
| 7 | Lion | Female | Yes | D |
| 8 | Goat | Female | Yes | E |
| 9 | Cat | Female | Yes | A |
| 10 | Dog | Male | Yes | B |

**Ans:-**

**Python code**

#################### Dummy variable ##################

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from scipy import stats

from sklearn.impute import SimpleImputer as si

## import data set animal\_category as Data Frame ####

df = pd.DataFrame(animal\_categorycsv)

### creating copy and keep original data safe

d = df.copy(deep=True)

df1 = df.copy(deep=True)

df1.shape

### Here we considered "Type" column as output variable, so eliminate that perticular column from analysis

df1.drop(['Types'], axis = 1, inplace=True)

df1.dtypes ## Data type of each column

### counts of each unique category

df1["Homly"].value\_counts()

df1["Animals"].value\_counts()

df1["Gender"].value\_counts()

### Here every independed variables "Homly","Animals & "Gender" are nominal datas. So there have not any order for them. Son instead of label encoding we are gonna apply "One Hot Encoding" here.

from sklearn.preprocessing import OneHotEncoder

#creating OneHotEncoder

OHE = OneHotEncoder(sparse = False, drop = 'first').fit(df1) ## initializing OneHotEncoder as OHE ### sparse = False=>Will return sparse matrix form output ### drop = first=> drop first column each dummy variables after One Hot Encoding

df2 = OHE.fit\_transform(df1[['Animals','Gender','Homly']])

df3=pd.DataFrame(df2,columns=[ 'Animals\_Dog', 'Animals\_Gout', 'Animals\_Lion', 'Animals\_Mouse', 'Gender\_Male','Homly\_Yes'])

### complete Data set

d4= pd.DataFrame(d[["Index","Types"]])

d\_complete = pd.concat([d4,df3],axis=1)

**Hints:**

For each assignment, the solution should be submitted in the below format

1. Work on each feature to create a data dictionary as displayed in the image displayed below:
2. Refer to animal\_category.csv data set.
3. Research and perform all possible steps for obtaining solution
4. All the codes (executable programs) should execute without errors
5. Code modularization should be followed
6. Each line of code should have comments explaining the logic and why you are using that function