Imputation of missing values in DataFrame

February 3, 2021

0.0.1 Problem statement:

It is always important to impute the missing values in dataframe. The are different approaches to impute the missing values. In pandas DataFrame, missing values is represented by NaN but sometimes we see the ?, ?? and ??? (question marks) in the DataFrame. In this case we can do the following procedures to identify the missing values before imputing. This project tries to find out the possible approaches to fill the missing values. I will be going to apply mean and median in order to fill the numeric variables and impute by mode in case of catagorical variables.

[1]:
```
import os
import pandas as pd
import numpy as np
```

[6]:
```
df=pd.read_csv('Toyota.csv', index_col=0)
```

[9]:
```
df.head(n=10)
```

[9]:

	Price	Age	KM	FuelType	HP	MetColor	Automatic	CC	Doors	Weight
0	13500	23.0	46986	Diesel	90	1.0	0	2000	three	1165
1	13750	23.0	72937	Diesel	90	1.0	0	2000	3	1165
2	13950	24.0	41711	Diesel	90	NaN	0	2000	3	1165
3	14950	26.0	48000	Diesel	90	0.0	0	2000	3	1165
4	13750	30.0	38500	Diesel	90	0.0	0	2000	3	1170
5	12950	32.0	61000	Diesel	90	0.0	0	2000	3	1170
6	16900	27.0	??	Diesel	????	NaN	0	2000	3	1245
7	18600	30.0	75889	NaN	90	1.0	0	2000	3	1245
8	21500	27.0	19700	Petrol	192	0.0	0	1800	3	1185
9	12950	23.0	71138	Diesel	????	NaN	0	1900	3	1105

[13]:
```
#We have ?? and ???? in the columns so we have to replace those by applying␣
 ↪na_values
df=pd.read_csv('Toyota.csv', index_col=0, na_values=["??", "????"])
```

[29]:
```
df.isnull().sum()
```

[29]:
```
Price          0
Age          100
KM            15
FuelType     100
```

```
HP              6
MetColor      150
Automatic       0
CC              0
Doors           0
Weight          0
dtype: int64
```

[41]:
```
data_cars.describe()
```

[41]:
```
              Price          Age             KM          HP     MetColor  \
count   1436.000000  1436.000000    1436.000000  1430.000000  1286.000000
mean   10730.824513    55.672156   68594.873259   101.478322     0.674961
std     3626.964585    17.930380   37140.890566    14.768255     0.468572
min     4350.000000     1.000000       1.000000    69.000000     0.000000
25%     8450.000000    44.000000   43505.750000    90.000000     0.000000
50%     9900.000000    59.000000   63634.000000   110.000000     1.000000
75%    11950.000000    68.000000   86916.000000   110.000000     1.000000
max    32500.000000    80.000000  243000.000000   192.000000     1.000000

         Automatic           CC      Weight
count  1436.000000  1436.000000  1436.00000
mean      0.055710  1566.827994  1072.45961
std       0.229441   187.182436    52.64112
min       0.000000  1300.000000  1000.00000
25%       0.000000  1400.000000  1040.00000
50%       0.000000  1600.000000  1070.00000
75%       0.000000  1600.000000  1085.00000
max       1.000000  2000.000000  1615.00000
```

[30]:
```
data_cars=df.copy()
```

[32]:
```
data_cars['Age'].fillna(data_cars['Age'].mean(), inplace=True)
```

[33]:
```
data_cars['Age'].isna().sum()
```

[33]:
```
0
```

[34]:
```
data_cars['KM'].fillna(data_cars['KM'].median(), inplace=True)
```

[35]:
```
data_cars['KM'].isna().sum()
```

[35]:
```
0
```

[36]:
```
#imputing missing values in catagorical columns
data_cars.dtypes=='O'
```

[36]:
```
Price         False
Age           False
KM            False
FuelType       True
HP            False
MetColor      False
Automatic     False
CC            False
Doors          True
Weight        False
dtype: bool
```

[37]:
```
data_cars['FuelType'].value_counts()
```

[37]:
```
Petrol    1177
Diesel     144
CNG         15
Name: FuelType, dtype: int64
```

[38]:
```
data_cars['FuelType'].mode()
```

[38]:
```
0    Petrol
dtype: object
```

[39]:
```
#Thus fuel types and Doors are catagorical variables
data_cars['FuelType'].fillna(data_cars['FuelType'].mode().index[0],␣
 ↪inplace=True)
```

[40]:
```
data_cars['FuelType'].isna().sum()
```

[40]:
```
0
```

[42]:
```
data_cars['MetColor'].value_counts()
```

[42]:
```
1.0    868
0.0    418
Name: MetColor, dtype: int64
```

[43]:
```
data_cars['MetColor'].mode()
```

[43]:
```
0    1.0
dtype: float64
```

[44]:
```
data_cars['MetColor'].fillna(data_cars['MetColor'].value_counts().index[0],␣
 ↪inplace=True)
```

[45]:
```
data_cars['MetColor'].isna().sum()
```

[45]:
```
0
```

[47]:
```
data_cars['HP'].fillna(data_cars['HP'].mean(), inplace=True)
```

[48]:
```
#Now check the missing values in the dataframes
data_cars.isnull().sum()
```

[48]:
```
Price        0
Age          0
KM           0
FuelType     0
HP           0
MetColor     0
Automatic    0
CC           0
Doors        0
Weight       0
dtype: int64
```

0.0.2 Conclusion:

Hence, we are able to do the missing values both in the numeric and catagorical variables. It is important to fill the missing values before Machine Learning model building. The accuracy of our ML model depends on how wisely we imputed the missing values in the dataframe. So, the descriptive statistics helps us to impute the missing values by mean, median and mode. Normally, the missing values in catagorical variables is imputed by the mode and numeric variables by the mean and median. I always recommend to impute the missing values in numeric variables by mean or median(which value is minimum and not so much differences among each other)

[ ]:
```
# Reference: https://www.youtube.com/watch?v=FdudxZN6rIo
```