Determining cardinality in categorical variables

The number of unique categories in a variable is called cardinality. For example, the cardinality of the Gender variable, which takes values of female and male , is 2 , whereas the cardinality of the Civil status variable, which takes values of married , divorced ,singled , and widowed , is 4.Here we will learn how to quantify and create plots of the cardinality of categorical variables using pandas and Matplotlib.

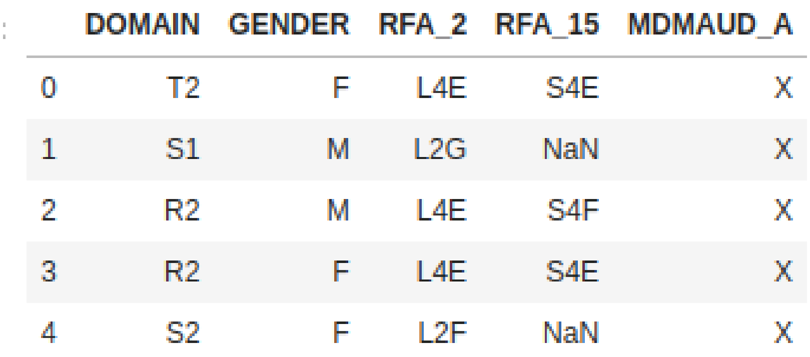
```
# import the required python libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# we will use the selected variables from a dataset
cols = ['GENDER', 'RFA_2', 'MDMAUD_A', 'RFA_2', 'DOMAIN', 'RFA_15']

# the dataset contains empty stings
# which are in essence missing values
# i replace these here
data = data.replace(' ', np.nan)

# loading dataset
data = pd.read_csv('data/cup98LRN.txt',usecols=cols)
data.head()
```

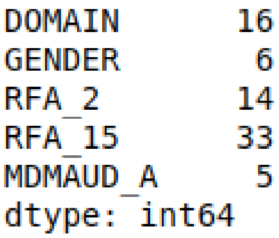
output:



```
   DOMAIN  GENDER  RFA_2  RFA_15  MDMAUD_A
0      T2       F    L4E     S4E         X
1      S1       M    L2G     NaN         X
2      R2       M    L4E     S4F         X
3      R2       F    L4E     S4E         X
4      S2       F    L2F     NaN         X
```

```
# let's determine the number of unique categories in each variable
data.nunique()
```

output:

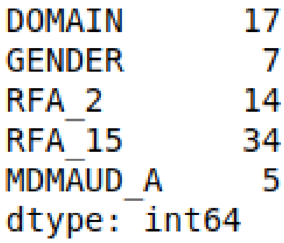


```
DOMAIN       16
GENDER        6
RFA_2        14
RFA_15       33
MDMAUD_A      5
dtype: int64
```

TIP: The nunique() method ignores missing values by default. If we want to consider missing values as an additional category, we should set the dropna argument to False: **data.nunique(dropna=False)**.

```
data.nunique(dropna=False)
```

output:



```
DOMAIN       17
GENDER        7
RFA_2        14
RFA_15       34
MDMAUD_A      5
dtype: int64
```

```
# let's print the different unique labels
data['GENDER'].unique()
```

output:

```
array(['F', 'M', nan, 'C', 'U', 'J', 'A'], dtype=object)
```

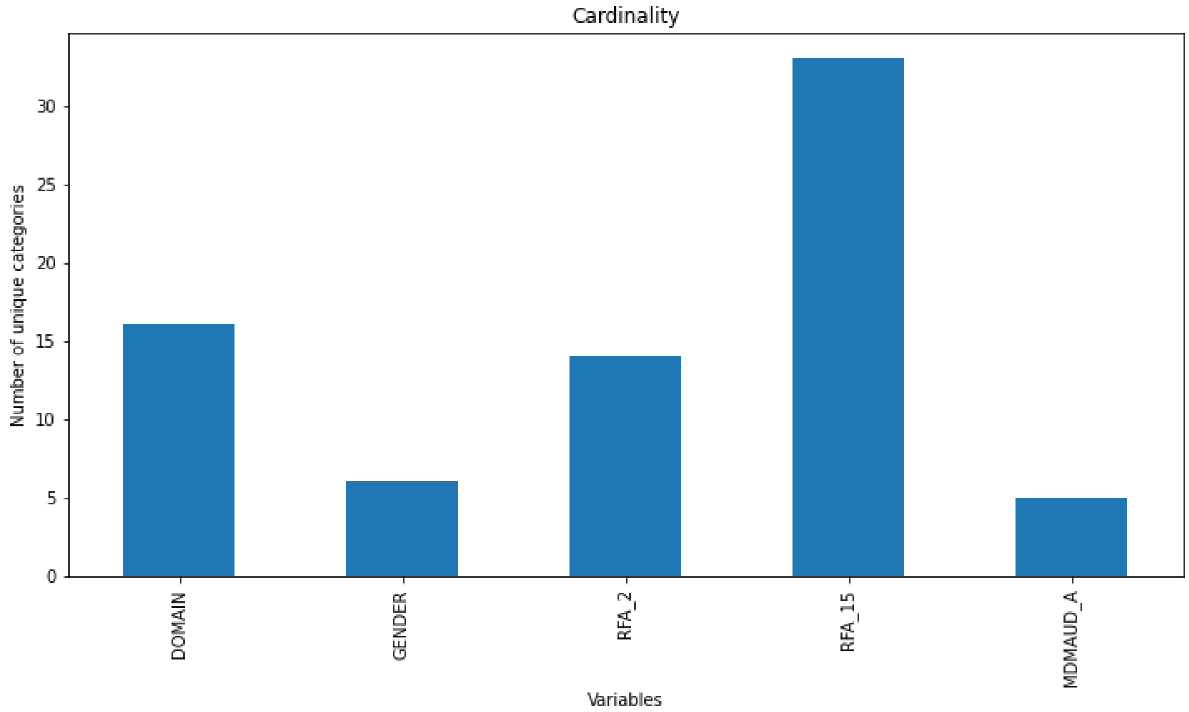
TIP: pandas **nunique**() can be used in the entire dataframe. pandas **unique**() , on the other hand, works only on a pandas Series. Thus, we need to specify the column name that we want to return the unique values for.

```
# let's plot the cardinality of the variables

data.nunique().plot.bar(figsize=(12,6))

# add labels and title
plt.ylabel('Number of unique categories')
plt.xlabel('Variables')
plt.title('Cardinality')
```

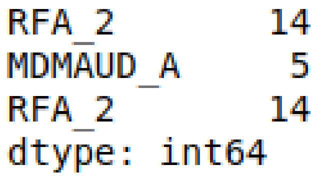
output:



```
# if we want to evaluate the cardinality of only a subset
# of columns from a data set, we can do so by passing the
# columns of interest as follows:

# evaluate cardinality of variables of choice
data[['RFA_2', 'MDMAUD_A', 'RFA_2']].nunique()
```

output:



```
RFA_2        14
MDMAUD_A      5
RFA_2        14
dtype: int64
```