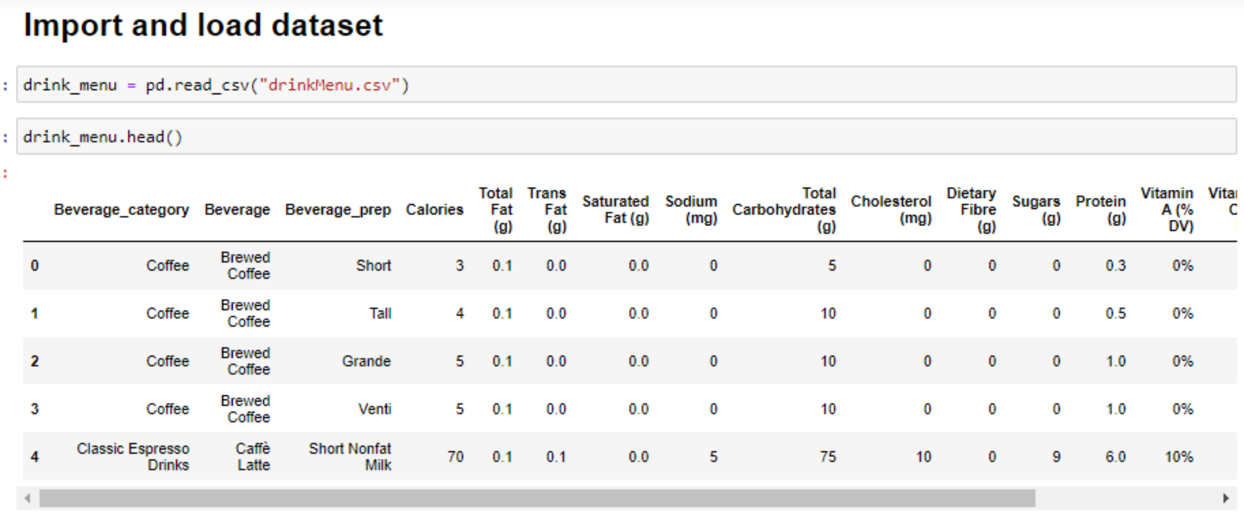
ET3 Internship 2022

Data Science task

Abdallah Medhat Toballa

Contents

1. Load Dataset



Import and load dataset

```
: drink_menu = pd.read_csv("drinkMenu.csv")
```

```
: drink_menu.head()
```

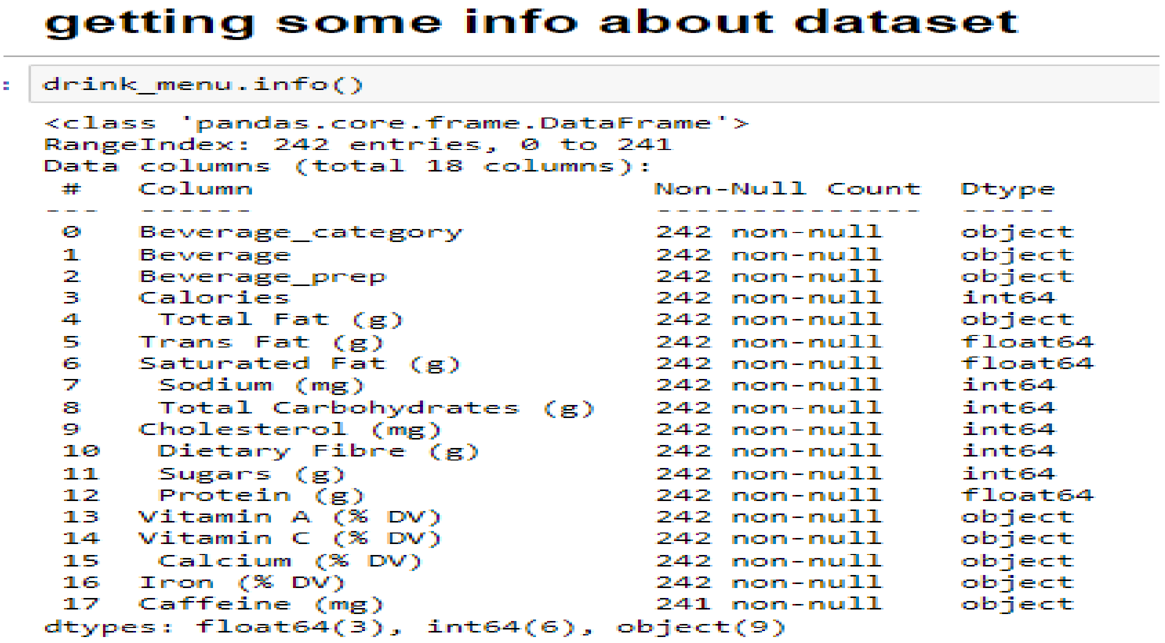
:

	Beverage_category	Beverage	Beverage_prep	Calories	Total Fat (g)	Trans Fat (g)	Saturated Fat (g)	Sodium (mg)	Total Carbohydrates (g)	Cholesterol (mg)	Dietary Fibre (g)	Sugars (g)	Protein (g)	Vitamin A (% DV)	Vitai C
0	Coffee	Brewed Coffee	Short	3	0.1	0.0	0.0	0	5	0	0	0	0.3	0%	
1	Coffee	Brewed Coffee	Tall	4	0.1	0.0	0.0	0	10	0	0	0	0.5	0%	
2	Coffee	Brewed Coffee	Grande	5	0.1	0.0	0.0	0	10	0	0	0	1.0	0%	
3	Coffee	Brewed Coffee	Venti	5	0.1	0.0	0.0	0	10	0	0	0	1.0	0%	
4	Classic Espresso Drinks	Caffè Latte	Short Nonfat Milk	70	0.1	0.1	0.0	5	75	10	0	9	6.0	10%	

Figure 1: loading the dataset

Firstly, I load dataset by using pandas then showing some rows from dataset and showing all dataset. I found dataset consist of 242 rows and 18 columns.

2. Getting some info about dataset



getting some info about dataset

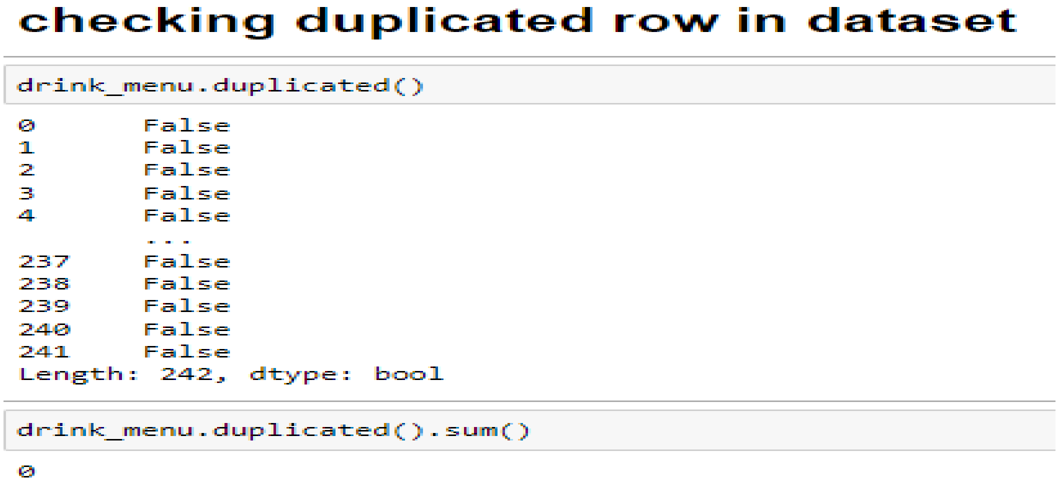
```
: drink_menu.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 242 entries, 0 to 241
Data columns (total 18 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Beverage_category          242 non-null    object
 1   Beverage                   242 non-null    object
 2   Beverage_prep              242 non-null    object
 3   Calories                   242 non-null    int64
 4    Total Fat (g)             242 non-null    object
 5   Trans Fat (g)              242 non-null    float64
 6   Saturated Fat (g)          242 non-null    float64
 7    Sodium (mg)               242 non-null    int64
 8    Total Carbohydrates (g)   242 non-null    int64
 9   Cholesterol (mg)           242 non-null    int64
 10   Dietary Fibre (g)         242 non-null    int64
 11   Sugars (g)                242 non-null    int64
 12   Protein (g)               242 non-null    float64
 13  Vitamin A (% DV)           242 non-null    object
 14  Vitamin C (% DV)           242 non-null    object
 15   Calcium (% DV)            242 non-null    object
 16  Iron (% DV)                242 non-null    object
 17  Caffeine (mg)              241 non-null    object
dtypes: float64(3), int64(6), object(9)
```

Figure 2 : Getting info about dataset

After getting some info about dataset, I found caffeine column contain on null value, type of 9 columns are object, type of 6 columns are int64 and type of 3 columns are float64.

3. Checking duplicated row in dataset



checking duplicated row in dataset

```
drink_menu.duplicated()
```

```
0      False
1      False
2      False
3      False
4      False
       ...
237    False
238    False
239    False
240    False
241    False
Length: 242, dtype: bool
```

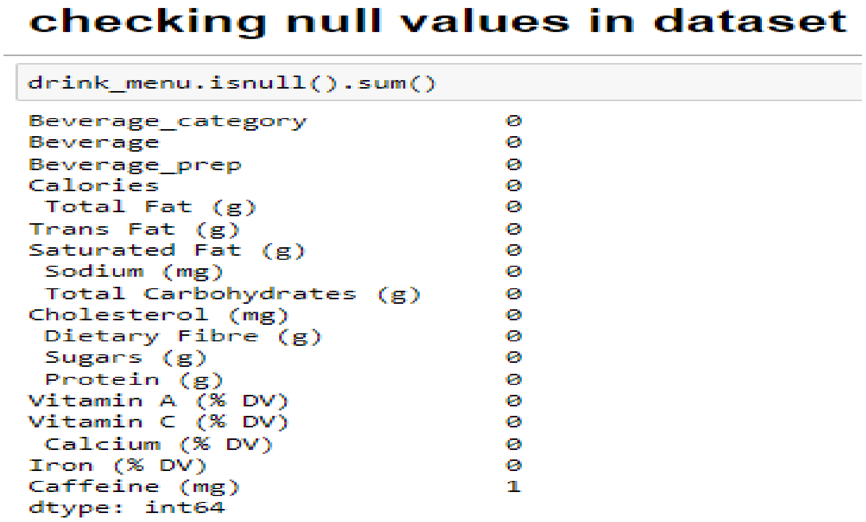
```
drink_menu.duplicated().sum()
```

```
0
```

Figure 3: Checking duplicated row in dataset

After checking duplicated row by using duplicated function, I found no row is duplicated.

4. Checking null values in dataset



checking null values in dataset

```
drink_menu.isnull().sum()
```

```
Beverage_category             0
Beverage                      0
Beverage_prep                 0
Calories                      0
 Total Fat (g)                0
Trans Fat (g)                 0
Saturated Fat (g)             0
 Sodium (mg)                  0
 Total Carbohydrates (g)      0
Cholesterol (mg)              0
 Dietary Fibre (g)            0
 Sugars (g)                   0
 Protein (g)                  0
Vitamin A (% DV)              0
Vitamin C (% DV)              0
 Calcium (% DV)               0
Iron (% DV)                   0
Caffeine (mg)                 1
dtype: int64
```

Figure 4: Checking null values in dataset

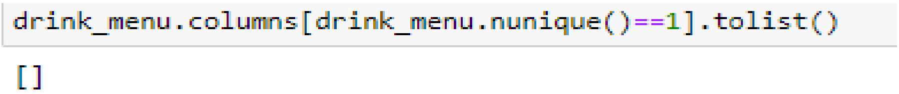
After checking null values in dataset and duplicated row in dataset once again, I found one null values in caffeine so dataset is clear because it contain on one null value and has not duplicated row. I fill null value in caffeine column by mode strategy because type of caffeine column is object and contain on varies which meaning is different from all values in caffeine column not specific value so I cannot deal with this value so as not to effect on machine learning phase.

Note

We can deal with caffeine column in machine learning phase by one hot encoding or mapping.

5. Dropping unnecessary Columns

1. Checking all columns in dataset which have one value using columns and unique functions.



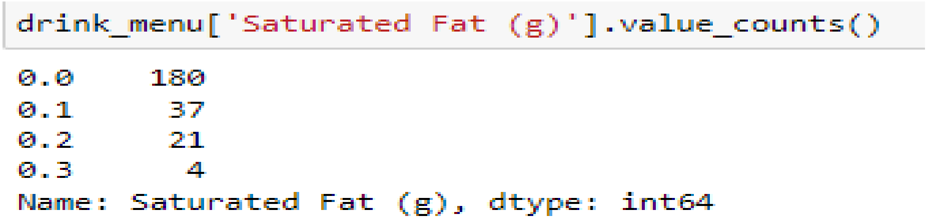
```
drink_menu.columns[drink_menu.nunique()==1].tolist()

[]
```

Figure 5: Checking all columns in dataset which have one value

I found no columns have one value so all columns is important but I tray another strategy to check unnecessary columns.

2. checking some columns which have huge difference between count values such as Dietary Fibre (g), Total Fat (g), Trans Fat (g), Saturated Fat (g), Vitamin A (% DV) and Beverage_prep.



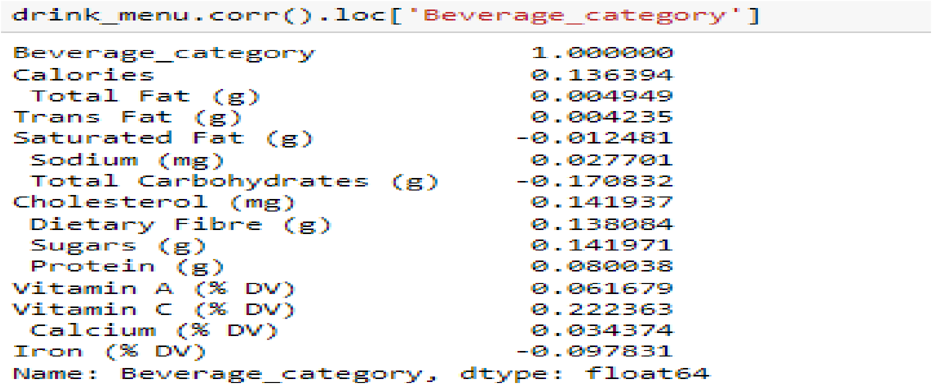
```
drink_menu['Saturated Fat (g)'].value_counts()

0.0    180
0.1     37
0.2     21
0.3      4
Name: Saturated Fat (g), dtype: int64
```

Figure 6: Example of value count for Saturated Fat (g) column

After checking some columns, I found no huge difference between count values in columns so all columns is important but I tray another strategy to check unnecessary columns.

3. Assuming this dataset is used to classify Beverage_category so Beverage_category is label column.



```
drink_menu.corr().loc['Beverage_category']

Beverage_category                 1.000000
Calories                          0.136394
 Total Fat (g)                    0.004949
Trans Fat (g)                     0.004235
Saturated Fat (g)                -0.012481
 Sodium (mg)                      0.027701
 Total Carbohydrates (g)         -0.170832
Cholesterol (mg)                  0.141937
 Dietary Fibre (g)                0.138084
 Sugars (g)                       0.141971
 Protein (g)                      0.080038
Vitamin A (% DV)                  0.061679
Vitamin C (% DV)                  0.222363
 Calcium (% DV)                   0.034374
Iron (% DV)                      -0.097831
Name: Beverage_category, dtype: float64
```

Figure 7: correlation between columns and label column

After my assuming about dataset, removing Beverage column because of my assuming which is classification for Beverage category and I prepare dataset to check correlation between columns with label column such as checking values in Total Fat (g) column, converting type of Total Fat (g) column to float type, removing % in Vitamin A (% DV), Vitamin C (% DV), Calcium (% DV) and Iron (% DV) columns by using regex, converting type of these columns to float type and using label encoder with label column (Beverage_category) to convert categorical values to numerical values.

After preparing dataset and checking correlation between columns with label column, I found correlation is very close between columns and target column therefore it is possible to delete some columns such as Total Fat (g) and Trans Fat (g) because Total Fat (g) and Trans Fat (g) have least correlation with label column. In finally, I delete three columns which are Beverage, Total Fat (g) and Trans Fat (g) based on my assuming.

Note

Before machine learning phase, we can apply more algorithms from feature selection to make sure about importance of all columns such as features importance by decision tree and Recursive Feature Elimination.

6. Data Visualizations

1. Visualizing bar plot to know which drinks have the highest calories between two columns which are Beverage and calories columns.

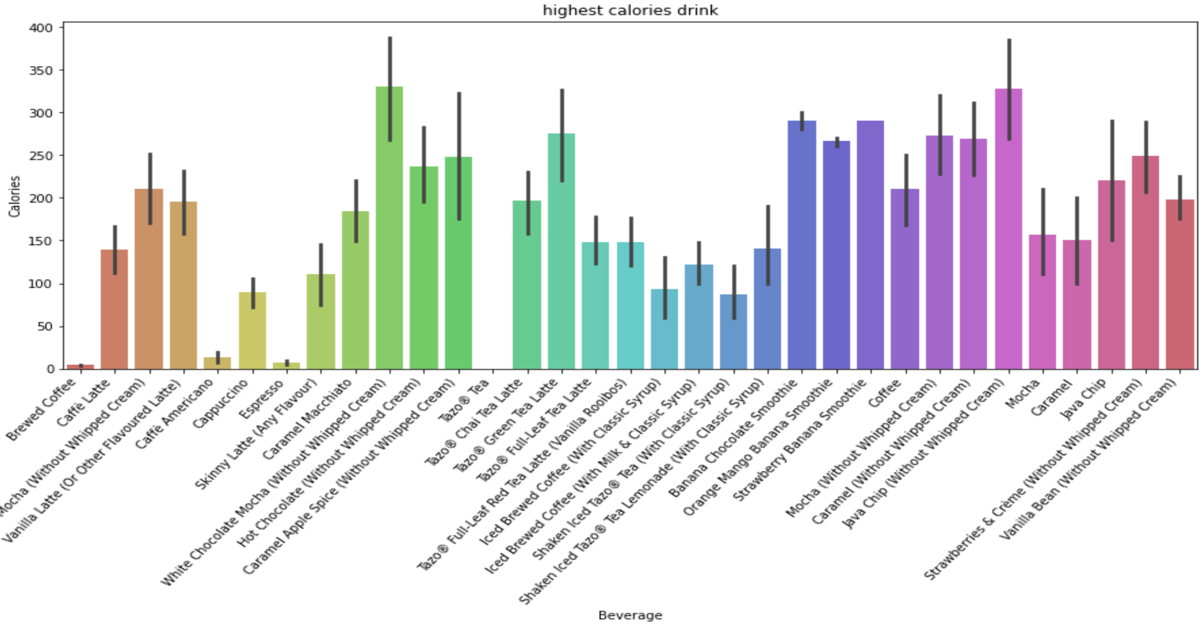


Figure 8: bar plot between Beverage and calories columns

After plotting bar plot between Beverage and calories columns, I found chocolate mocha (without whipped cream) to have highest calories then Java Chip (Without Whipped Cream) the second highest calories.

2. Visualizing bar plot to know which drinks have the highest Sugar between two columns which are Beverage and Sugars (g) columns.

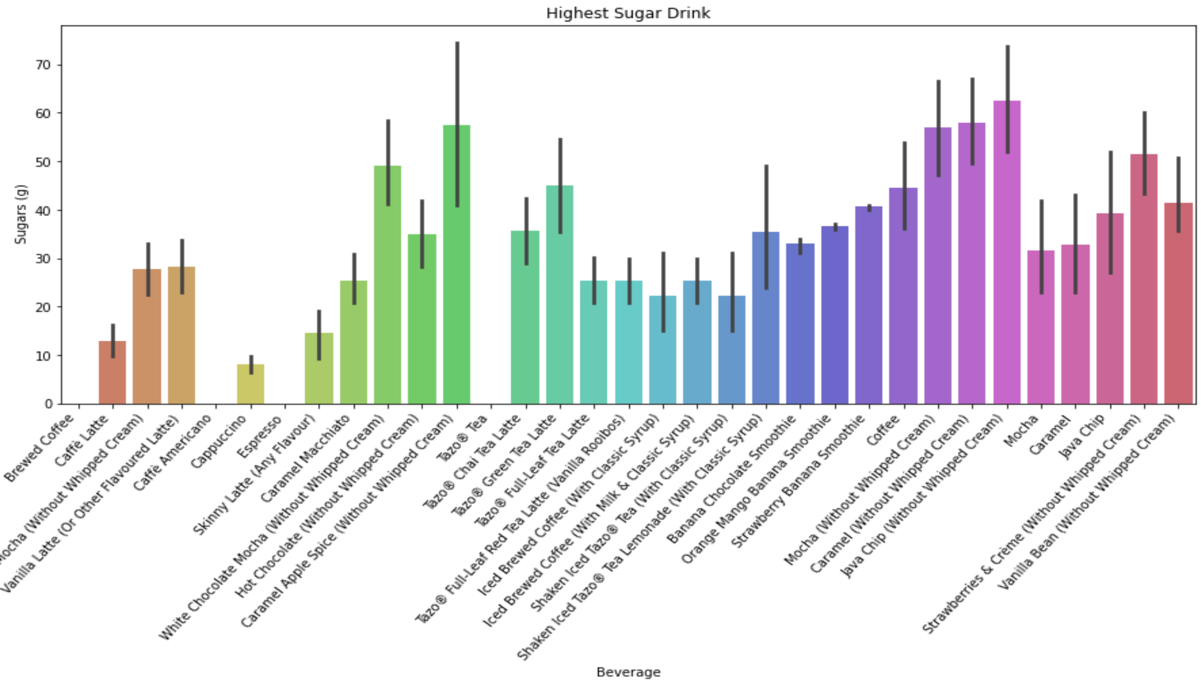


Figure 9: bar plot between Beverage and Sugars (g) columns

After plotting bar plot between Beverage and Sugars (g) columns, I found Java Chip (Without Whipped Cream) to have highest Sugars then Caramel Apple Spice (Without Whipped Cream) the second highest Sugars.

Notes

1. Using bar plot because it is useful in display relationship between a numeric and a categorical variable.
2. Using Beverage because you want drink but if you want category, I will use Beverage _category column.

7. Conclusion

I found dataset is clear because I found one null value in dataset and dataset has not duplicated row, it is possible to delete three columns from dataset based on my assuming. After Visualizing, I found chocolate mocha (without whipped cream) to have highest calories then Java Chip (Without Whipped Cream) the second highest calories and Java Chip (Without Whipped Cream) to have highest Sugars then Caramel Apple Spice (Without Whipped Cream) the second highest Sugars.

8. Instructions on how to run my solutions

1. Download task from my GitHub.

2. Check all requirements are installed such as Jupyter notebook, python3 and libraries of python such as numpy, pandas, matplotlib, seaborn and sklearn.
3. Open cmd in directory of task and write commend jupyter notebook.
4. After opening jupyter notebook, open Data Science task.ipynb file then run each cell or all cell in Data Science task.ipynb file.