ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.6

Data Analysis on ‘Nutrition Facts for McDonald's Menu’ Data-set using Python

Neha Tiwari[1], Prof. Vaishali Gatty[2]
Student[1], Professor[2]
Department of MCA
Vivekanand Education Society’s Institute of Technology, India

Abstract:
Python is now-a-days easy to go programming language which is so popular due to its multiple features and applications. Python has become the language choice for most of data scientists now-a-days for data & its operations like visualization, analysis, manipulation, retrieval, cleaning, and machine learning. It uses open source platform and libraries such as NumPy, Scipy, matplotlib, pandas, scikit-learn etc. This paper aims to highlight data analysis of ' Nutrition Facts for McDonald's Menu' dataset using Python. The Indian food industry has risen as a high-development and high-benefit area because of its huge potential for esteem expansion, especially inside the food processing industry. This dataset is used to analyze nutritious and non-nutritious food items in the menu. It uses various python libraries to analyze this dataset to represent the data in the form of different charts.

Keywords: Chart Diagrams, Data Analysis, Data-set, Nutritious, Non-nutritious, Python.

1. INTRODUCTION

The Python programming language is very popular today because of its features and use. So in this paper the data analysis of a ‘Nutrition Facts for McDonald's Menu’ data-set is done using Python language. There are total 9 sections in this paper which are as follows: section 2 represents Introduction to Python, section 3 represents Why python is used for Data Analysis, section 4 represents Applications of Python, section 5 represents Introduction to data-set, section 6 represents Analysis Performed on Data-set, section 7 represents Result using different chart diagrams, section 8 represents Conclusion, while section 9 represents References used.

2. INTRODUCTION TO PYTHON

The Python programming language was conceived in the late 1980s, and its implementation was started in December 1989 by Guido van Rossum at CWI in the Netherlands as a successor to the ABC programming language capable of exception handling and interfacing with the Amoeba operating system [1]. Python is an translated, object-oriented, high-level programming language with dynamic semantics. It’s high-level built in data structures, consolidated with dynamic typing and dynamic binding make it very exceptionally appealing for Rapid Application Development. Python supports modules and packages, which supports program seclusion and code reuse. The Python translator and the broad standard library are accessible in source or parallel frame without charge for every single significant stage, and can be uninhibitedly circulated.[2] Python has some one of the kind of elements so it can be utilized as a part of numerous applications. Some of these components are as per the following: Utilizes a rich language structure, making the projects you compose less demanding to peruse. It is a simple to-utilize language that makes it easy to get your program working. This makes Python perfect for model development and other specially appointed programming tasks, without trading off viability. It accompanies a huge standard library that backings numerous normal programming errands, for example, associating with web servers, searching content with regular expressions, reading and altering files. Python's intuitive mode makes it simple to test short scraps of code. There's likewise a packaged improvement condition called IDLE. It is effortlessly stretched out by including new modules executed in a gathered language, for example, C or C++. It can likewise be inserted into an application to give a programmable interface. It runs anywhere, including Mac OS X, Windows, Linux, and Unix. It is free programming in two detects. It doesn't cost anything to download or utilize Python, or to incorporate it in your application. Python can likewise be uninhibitedly altered and re-distributed, on the grounds that while the language is copyrighted it's accessible under an open source license[3].

3. WHY PYTHON IS USED FOR DATA ANALYSIS

The scripting language Python currently available in 2 different versions, python 3.4.3 released in February 2015 while python released in December 2014.Many data analyst use python for analysis of data-sets. So python has certain features which enables it to be used for data analysis purpose.

1. Purpose -Python focuses on productivity and code readability.
2. Used by -It is used by programmers that want to dive into data analysis Or apply statistical / mathematical techniques And by developers that turn to data science
3. Usability -Coding and debugging is much easier to do in Python because of simple syntax and terminology. The indentation of code affects its meaning.
4. Flexibility -It is flexible for doing something that has never been done before. Developers can use Python for scripting a website or other applications.

5. Ease of learning -Python makes learning curve relatively low and gradual, So it good for starting programmers.
6. Set of Libraries -In python there are many libraries which we can use as per use for extracting analysis from data-sets. There are many libraries , some of main libraries that are most commonly used libraries are NumPy (Numerical Python), SciPy (Scientific Python), Matplotlib, Pandas, Scikit Learn, Scrapy, Bokeh, Pygal etc.
7. Python IDE's -There are many Python IDE's, most popular are Spyder and IPython Notebook
8. Python Testing Framework -Python's testing framework guarantee that code is reusable and dependable.
9. Open Source -Python is free to download for everyone so good for developers, programmers and data analyst [4].

4. APPLICATIONS OF PYTHON

Python is utilized as a part of numerous application spaces. The Python Package Index records a huge number of outsider modules for Python. Here's a listing.

4.1. Web and Internet Development

Python offers numerous decisions for web advancement:
Frameworks, for example, Django and Pyramid.
Miniaturized scale systems, for example, Flask and Bottle.
Advanced content administration frameworks, for example, Plone and django CMS.
Python's standard library supports numerous Internet conventions:
HTML and XML, JSON, Email preparing. Support for FTP, IMAP, and other Internet conventions,
Simple to-utilize attachment interface.
Furthermore, the Package Index has yet more libraries: Demands, an intense HTTP customer library. Beautiful Soup, a HTML parser that can deal with a wide range of oddball HTML.
Feed parser for parsing RSS/Atom sustains.
Paramiko, executing the SSH2 convention.
Twisted Python, a system for offbeat system programming.

4.2. Logical and Numeric

Python is generally utilized as a part of logical and numeric figuring:
SciPy is an accumulation of packages for arithmetic, science, and building.
Pandas is an information investigation and displaying library.
IPython is an intense intuitive shell that components simple altering and recording of a work session, and supports visual representations and parallel processing.

4.3. Education

Python is a great language for showing programming, both at the early on level and in more propelled courses.
Books, for example, How to Think Like a Computer Scientist, Python Programming: An Introduction to Computer Science, and Practical Programming.
The Education Special Interest Group is a decent place to talk about instructing issues.

4.4. Desktop GUIs

The Tk GUI library is incorporated with most paired dispersions of Python.
Some toolkits that are usable on a few stages are accessible independently:
wxWidgets Kivy, for composing multitouch applications.
Qt by means of pyqt or pyside
Stage particular toolboxs are likewise accessible:
GTK+
Microsoft Foundation Classes through the win32 augmentations

4.4.1. Image Processing and Graphic Design Applications: Python has been utilized to make 2D imaging programming, for example, Inkscape, GIMP, Paint Shop Pro and Scribus. Further, 3D movement bundles, similar to Blender, 3ds Max, Cinema 4D, Houdini, Light wave and Maya, additionally utilize Python in factor extents.

4.4.2. Logical and Computational Applications: The higher paces, profitability and accessibility of devices, for example, Scientific Python and Numeric Python, have brought about Python turning into a basic piece of uses required in calculation and preparing of logical information. 3D modeling software, for example, FreeCAD, and limited component method software, for example, Abaqus, are coded in Python.

4.5 Games: Python has different modules, libraries and stages that supports development of games. For instance, PySoy is a 3D game motor supporting Python 3, and PyGame gives usefulness and a library for game advancement. There have been various recreations constructed utilizing Python including Civilization-IV, Disney's Toontown Online, Vega Strike and so forth.

4.6 Operating Systems: Python is frequently a integral part of Linux distributions. For example, Ubuntu's Ubiquity Installer, and Fedora's and Red Hat Enterprise Linux's Anaconda Installer are composed in Python. Gentoo Linux makes utilization of Python for Portage, its package administration framework.[5]

4.7 Programming Development

Python is regularly utilized as a support language for programming engineers, for assemble control and administration, testing, and in numerous different ways. SCons for manufacture control. Buildbot and Apache Gump for computerized persistent assemblage and testing.
Gathering or Trac for bug following and venture administration.

5. INTRODUCTION TO DATA-SET

Ray Kroc needed to fabricate an eatery system that would be acclaimed for giving food of reliably high caliber and uniform strategies for preparation. He needed to serve burgers, buns, fries and drinks that tasted only the same in Alaska as they did in Alabama. To accomplish this, he picked a one of a kind way: inducing both franchisees and providers to become tied up with his vision, working not for McDonald's but rather for themselves, together with McDonald's. Huge numbers of McDonald's most acclaimed menu things – like the Big Mac, Filet-O-Fish, and Egg McMuffin – were made by franchisees. The ‘Nutrition Facts for McDonald's Menu’ [6] dataset gives a nutrition examination of each menu thing on the US McDonald's menu, including breakfast, hamburger burgers, chicken and fish sandwiches, fries, servings of mixed greens, pop, espresso and tea, milkshakes, and desserts.

So there is lot of information of menu items which contains basically, Category, Item, Serving Size, Calories, Calories from Fat, Total Fat, Total Fat (% Daily Value), Saturated Fat, Saturated Fat (% Daily Value), Trans Fat, Cholesterol, Cholesterol (% Daily Value), Sodium, Sodium (% Daily Value), Carbohydrates, Carbohydrates (% Daily Value), Dietary Fiber, Dietary Fiber (% Daily Value), Sugars, Protein, Vitamin A (% Daily Value), Vitamin C (% Daily Value), Calcium (% Daily Value), Iron (% Daily Value).

6. ANALYSIS PERFORMED ON DATA-SET

- Import csv file in python

In Python:

```
>>> import csv
>>> with open('C:\\Users\\Bappa\\Pictures\\menu.csv', encoding='utf-8', newline='') as f:
        reader = csv. reader(f)
        for row in reader:
                print(', '.join(row))
                print(row)
```

Result : It will import Menu.csv file of data-set

-To get first 10 lines of dataset with specific columns

In Python:

```
>>> import csv, itertools
>>> with open('C:\\Users\\Bappa\\Pictures\\menu.csv', encoding='utf-8', newline='') as csvfile:
        for row in itertools. Islice (csv.DictReader(csvfile), 10):
                print(row['Category'], row['Item'], row['Serving Size'])
```

Result : Here function islice() will create an iterator from the iterable object you pass and it will allow you iterate till the limit, you pass as the second parameter.

-**Import all necessary files**

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.tools as tls
import warnings
warnings. filter warnings('ignore')
```

- Sugar content in Menu's items

Create a new Data Frame with the columns Item and Sugars and find first 10 items containing high sugar content value.

In Python:

```
df_sugars = pd.DataFrame(columns=('Item','Sugars'))
df_sugars['Item'] = menu['Item']
df_sugars['Sugars'] = menu['Sugars']
print("Let's sort them by the amount of sugar they have in a ascending order: ")
df_sugars = df_sugars.sort_values('Sugars', ascending=[True])
print(df_sugars.head(10))
```

Result:

```
Let's sort them by the amount of sugar they have in a ascending order:

Item                                  Sugars
145    Coffee                  (Small)     0
99     Kids      French        Fries       0
96     Small     French        Fries       0
81     Chicken   McNuggets     (20piece)   0
114    Diet      Coke          (Small)     0
115    Diet      Coke          (Medium)    0
116    Diet      Coke          (Large)     0
117    Diet      Coke          (Child)     0
122    Diet      DrPepper      (Small)     0
123    Diet      Dr Pepper     (Medium)    0
```

-Check for item which contains no sugar.

In Python:

```
print("Number of items in the menu: "+str(len(menu.index)))
print("Number of items without sugar in the menu: "+str(len(df_sugars.loc[df_sugars['Sugars'] == 0])))
print(df_sugars.loc[df_sugars['Sugars'] == 0])
```

Result:

```
Number of items in the menu: 260
Number of items without sugar in the menu: 25
                Item                              Sugars
145             Coffee         (Small)               0
99         Kids      French       Fries              0
96         Small     French       Fries              0
81    Chicken    McNuggets    (20    piece)          0
114        Diet      Coke         (Small)            0
115        Diet      Coke         (Medium)           0
116        Diet      Coke         (Large)            0
117        Diet      Coke         (Child)            0
122      Diet    Dr    Pepper    (Small)             0
123      Diet    Dr    Pepper    (Medium)            0
124      Diet    Dr    Pepper    (Large)             0
98         Large     French       Fries              0
80    Chicken    McNuggets    (10    piece)          0
79    Chicken    McNuggets    (6     piece)          0
136        Dasani    Water        Bottle             0
137        Iced      Tea          (Small)            0
138        Iced      Tea          (Medium)           0
139        Iced      Tea          (Large)            0
140        Iced      Tea          (Child)            0
78    Chicken    McNuggets    (4     piece)          0
146             Coffee         (Medium)              0
38                Hash         Brown                 0
147             Coffee         (Large)               0
125      Diet    Dr    Pepper    (Child)             0
97      Medium French Fries      0
```

So only 25 elements of 260, which means that only the **9.61**% of the items in McDonalds doesn't have any amount of sugar.

7. RESULT USING DIFFERENT CHART DIAGRAMS

It is important to show the result in form of chart diagrams so that it is easily identified. There are many chart diagrams that can be drawn using libraries in python [7]. In this paper, bar diagram, pie chart, scatter diagram, heatmap diagram are shown with result and analysis.

7.1. Bar Diagram of Calories in Different Category of Menu Data-set

In Python:

```
mc_menu = read.csv("../input/menu.csv", header = T, sep = ",")
# PIVORT TABLE OF CATEGORY AND SUM OF CALORIES
aggregate(mc_menu$Calories, by=list(mc_menu$Category), sum)
calories_cat = as.data.frame(aggregate(mc_menu$Calories, by=list(mc_menu$Category), sum))
library(ggplot2)
ggplot(calories_cat ) + geom_col(aes(Group.1, x, fill=rainbow(9))) +
 geom_text(aes(x=Group.1 , y=x , label = x))+
 labs(title = "Each Category Containg number of Calories", x= "Categories", y= "Calories")+
 theme(
   plot.background = element_rect(fill="#F0F3F4"),
   panel.grid.major = element_line(colour = "#37474F"),
   panel.background = element_rect(fill="#F0F3F4"),
   axis.title.y = element_text(colour = "#3E2723", angle=90),
   axis.title.x = element_text(colour = "#3E2723", angle = 0),
   axis.text = element_text(colour = "#3E2723"),
   legend.position = "none")
```

Result:

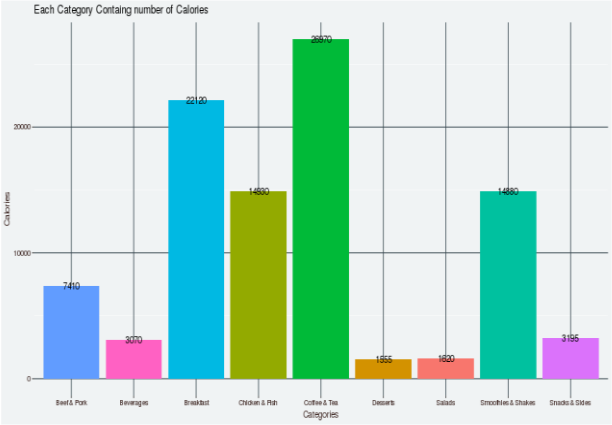

Figure.1. Bar Diagram 1

Analysis:

From above, Fig 1: Bar Diagram 1 it is found that menu item with calorie values as follows: beef & pork contains calories 7410, beverage contains calories 3070,

breakfast contains calories 22120, chicken & fish contains 14830, coffee & tea with highest calories 26970, desserts contains calories 1555, salad contains 1620, smoothies & shakes contains calories 14880, while snacks and sides contains calories 3196.

7.2. Pie-Chart of Menu Category with Calories

In Python:

```
var=df.groupby(['Calorie']).sum().stack()
temp=var.unstack()
type(temp)
x_list = temp['Category']
label_list = temp.index
pyplot.axis("equal") #The pie chart is oval by default. To make it a circle use pyplot.axis("equal")
plt.pie(x_list,labels=label_list,autopct="%1.1f%%")
plt.title("Pie-chart of Menu Category with Calories")
plt.show()
```

Result:

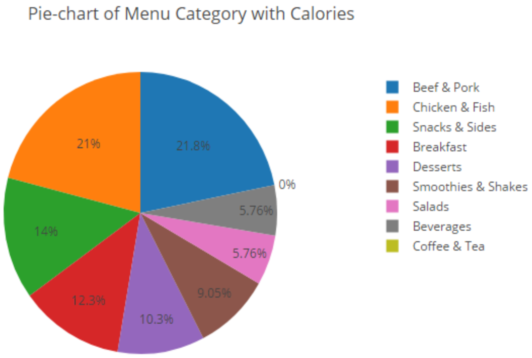

Figure.2. Pie Chart 1

Analysis:

From above, Fig 2: Pie Chart 1 we found that different category of menu in McDonald's menu dataset with their calorie values in percentage (%). So highest value of Calorie found in Category Beef & Pork with value 21.8%.While other categories Chicken & Fish with 21%, Snacks and Sides with 14%, Breakfast 12.3%, Desserts 10.3%, Smoothies & Shakes with 9.05%, Salads with 5.76% , Beverages with 5.76%.

7.3. Pie-Chart for Category with Cholesterol

In Python:

```
var=df.groupby(['Choleterol']).sum().stack()
temp=var.unstack()
type(temp)
x_list = temp['Category']
label_list = temp.index
pyplot.axis("equal") #The pie chart is oval by default. To make it a circle use pyplot.axis("equal")
plt.pie(x_list,labels=label_list,autopct="%1.1f%%")
plt.title("Pie-chart for Category with Cholesterol (% Daily Value)")
plt.show()
```

Result:

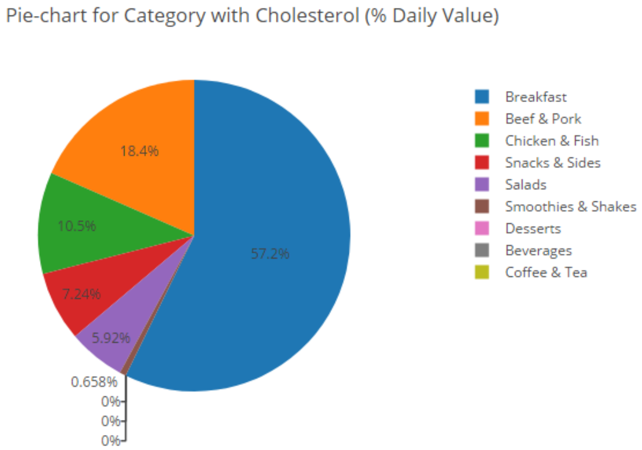

Figure.3. Pie Chart 2

Analysis:

From above, Fig 3:Pie Chart 2 it is found that different category of menu in McDonald’s menu dataset with their cholesterol values in percentage (%). So highest value found in Breakfast category with highest value 57.2%.

7.4. Bar Chart for Category highest Range

In Python:

```
import matplotlib.pyplot as plt
ax = df[['Category']].plot(kind='bar', title ="Categories in Menu Data-set of McDonald's Menu", figsize=(15, 10), fontsize=12)
ax.set_xlabel("Different Categories in Menu", fontsize=12)
ax.set_ylabel("Range", fontsize=12)
plt.show()
```

Result:

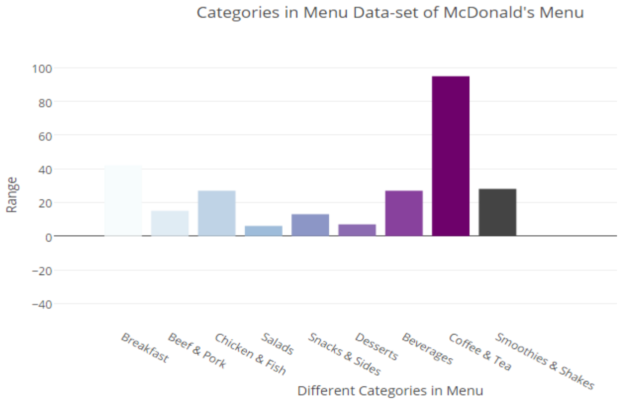

Figure.4. Bar Diagram 2

Analysis:

From the above Fig 4: Bar Diagram 2, it is found that, major categories of menu item found in McDonald’s menu is Coffee and Tea with almost 95% value.

7.5. Bar Chart for Vitamin A Range in % for Different Category

In Python:

```
import matplotlib.pyplot as plt
ax = df[['Category']].plot(kind='bar', title ="Vitamin A Range for Different Category", figsize=(15, 10), fontsize=12)
ax.set_xlabel("Vitamin A (% Daily Value)", fontsize=12)
ax.set_ylabel("Range in %", fontsize=12)
plt.show()
```

Result:

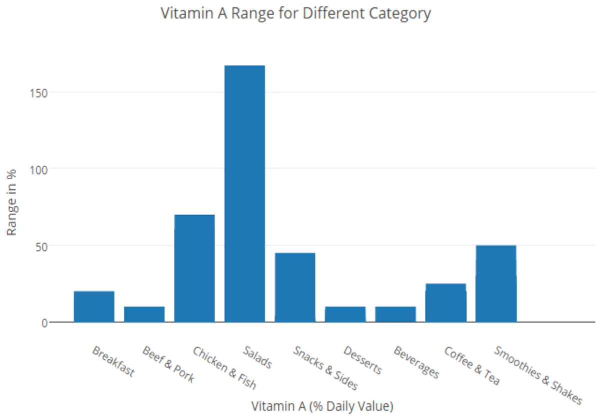

Figure.5. Bar Diagram 3

Analysis:

From above Fig 5: Bar Diagram 3, it is found Vitamin A is majorly (almost 170%) in salad category of McDonald’s Menu dataset. So salad is healthy nutritious food from this menu dataset.

7. 6. Bar Chart for Vitamin C Range in % for Different Category

In Python:

```
import matplotlib.pyplot as plt
ax = df[['Category']].plot(kind='bar', title ="Vitamin C Range for Different Category", figsize=(15, 10), fontsize=12)
ax.set_xlabel("Vitamin C (% Daily Value)", fontsize=12)
ax.set_ylabel("Range in %", fontsize=12)
plt.show()
```

Result:

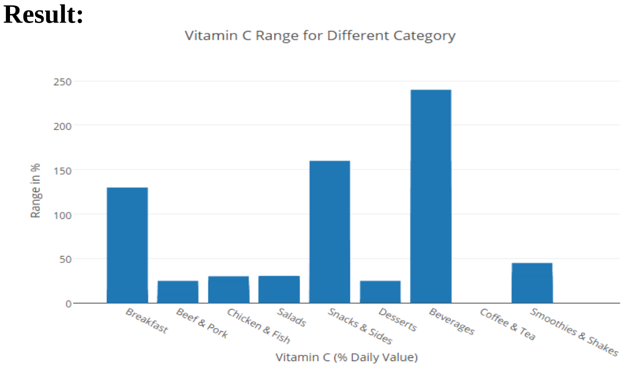

Figure.6. Bar Diagram 4

Analysis:

From above Fig 6: Bar Diagram 4, it is found that Vitamin C is majorly (almost 240%) in Beverages Category of McDonald’s Menu dataset. So Beverages is rich source of Vitamin C from this menu dataset.

7.7. Bar Chart For Sodium Range in % for Different Category

In Python:

```
import matplotlib.pyplot as plt
ax = df[['Category']].plot(kind='bar', title ="Sodium Range for Different Category", figsize=(15, 10), fontsize=12)
ax.set_xlabel("Sodium (% Daily Value)", fontsize=12)
ax.set_ylabel("Range in %", fontsize=12)
plt.show()
```

Result:

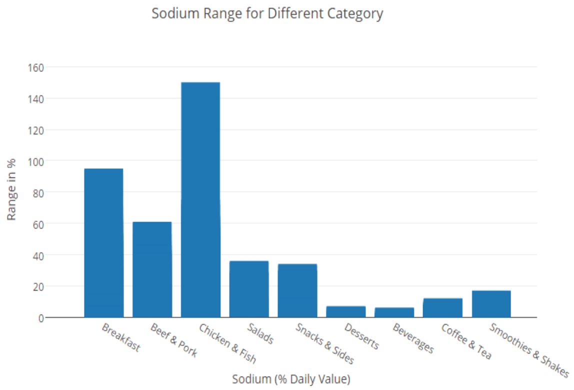

Figure.7. Bar Diagram 5

Analysis:

From above Fig 7: Bar Diagram 5, it is found that Sodium is majorly (almost 150%) found in Chicken and Fish Category of McDonald's Menu dataset. So Chicken and Fish is rich source of Sodium from this menu dataset.

7.8. Bar Chart For Dietary Fiber Range in % for different Category

In Python:

```
import matplotlib.pyplot as plt
ax = df[['Category']].plot(kind='bar', title ="Dietary Fiber Range for Different Category", figsize=(15, 10), fontsize=12)
ax.set_xlabel("Dietary Fiber (% Daily Value)", fontsize=12)
ax.set_ylabel("Range in %", fontsize=12)
plt.show()
```

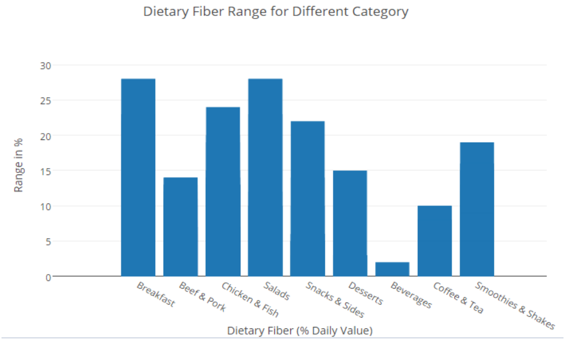

Figure.8. Bar Diagram 6

Analysis:

From above Fig 8: Bar Diagram 6, it is found that Dietary Fiber is majorly (almost 28%) in both Breakfast and Salad Category of McDonald's Menu dataset. Also Dietary fiber is found in most of menu categories from this menu dataset.

7.9. Bar Diagram of Sugar Content for Different Category

In Python:

```
import matplotlib.pyplot as plt
ax = df[['Category']].plot(kind='bar', title ="Sugar Content in Different Category", figsize=(15, 10), fontsize=12)
ax.set_xlabel("Category)", fontsize=12)
ax.set_ylabel("Range in %", fontsize=12)
plt.show()
```

Result:

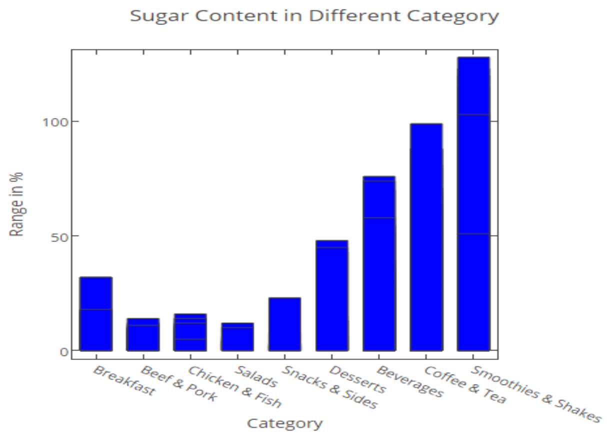

Figure.9. Bar Diagram 7

Analysis:

From above Fig 9: Bar Diagram 7, it is found that Sugar is majorly (almost 140%) in both Smoothies & Shakes Category of McDonald's Menu dataset. Also Sugar is found in most of menu categories from this menu dataset.

7.10. Scatter chart for Protein Range in % for different Category

In Python :

```
import matplotlib.pyplot as plt
plot_ly(x = ~Protein, y = ~Carbohydrates, color = ~Dietary_Fiber, text = ~paste("Item ",Item),type= "scatter", mode=" markers",size=~Total.Fat) %>%
 layout(title="Proteins, Dietary Fibre,Carbs and Total Fat (Size represents Total Fat) ")
```

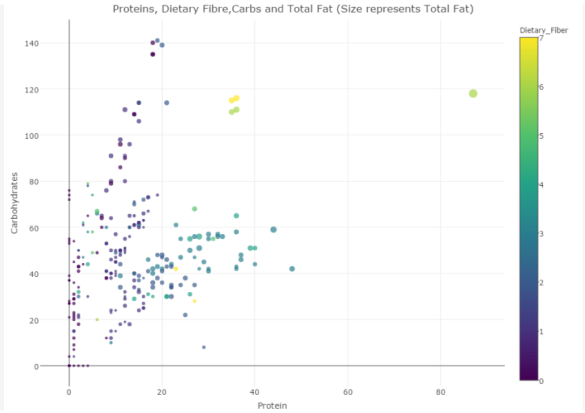

Figure.10. Scatter Diagram 1

Analysis:

From the above, Fig 10: Scatter Diagram 1, it is found that the food item with the maximum amount of protein content is the Chicken McNugget (40Pieces). So this item has the highest fat content overall.

7.11. Selecting nutritious and non-nutritious food from menu dataset

It is very important to know about nutritious food and non-nutritious food items from this menu data-set.

In Python:

```
 neturicious = ['Protein', 'Vitamin A (% Daily Value)','Dietary Fiber', 'Vitamin C (% Daily Value)', 'Calcium (% Daily Value)', 'Iron (% Daily Value)']
nonneturicious = ['Total Fat', 'Calories from Fat', 'Saturated Fat', 'Trans Fat', 'Cholesterol']
data['neturicious'] = data['Protein'] + data['Vitamin A (% Daily Value)'] + data['Vitamin C (% Daily Value)'] + data['Dietary Fiber'] + data['Calcium (% Daily Value)'] + data['Iron (% Daily Value)']
data['nonneturicious'] = data['Total Fat'] + data['Calories from Fat'] + data['Saturated Fat'] + data['Trans Fat'] + data['Cholesterol']
```

Input:

```
# selecting Breakfast category to show neutricious food
df = data[data['Category'] == 'Breakfast']
df = df.groupby(['Item']).sum().sort_values(by='neturicious', ascending=False).head(10)
```

```
fig, ax = plt.subplots(figsize=(7, 5))
df = df[neturicious]
sns.heatmap(df, ax=ax, annot=True)
ax.set_title('What to choose for Breakfast')
plt.xticks(rotation=45)
```

Result:

Output :

```
(array([ 0.5,  1.5,  2.5,  3.5,  4.5,  5.5]),
<a list of 6 Text xticklabel objects>)
```

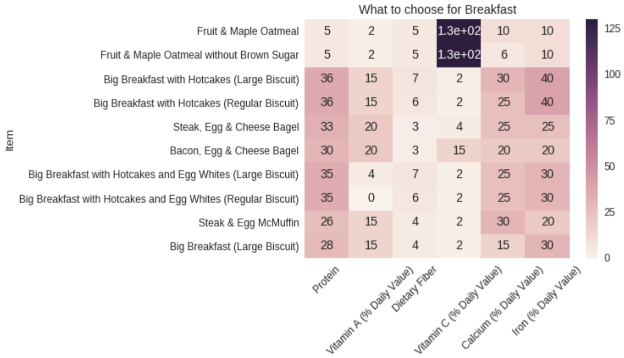

Figure.11. Heatmap Diagram 1

Input:

```
# selecting Breakfast category to show non-neutricious food
df = data[data['Category'] == 'Breakfast']
df = df.groupby(['Item']).sum().sort_values(by='nonneturicious',
ascending=False).head(10)
fig, ax = plt.subplots(figsize=(9, 5))
df = df[nonneturicious]
sns.heatmap(df, ax=ax, annot=True)
ax.set_title('What not to choose for Breakfast')
plt.xticks(rotation=45)
```

Result:

Output:

```
(array([ 0.5,  1.5,  2.5,  3.5,  4.5]), <a list of 5 Text xticklabel
objects>)
```

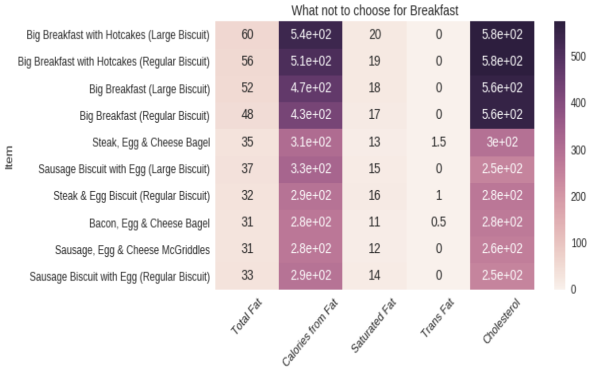

Figure.12. Heatmap Diagram 2

8. CONCLUSION

Python programming language used to demonstrate data analysis for ‘Nutrition Facts for McDonald's Menu’ dataset. It is analyzed that the items in menu dataset can be categorized as nutritious food and non-nutritious food based on different chart diagrams and range values in percentage obtained. So it is beneficial for demonstrating different range values for food nutrients such as vitamin A, vitamin B, vitamin C, sugar, dietary fibers, fats, carbohydrates, cholesterol, iron, sodium and protein for their proper consumption from menu items.

9. REFERENCES

[1] https://en.wikipedia.org/wiki/History_of_Python

[2]. https://www.python.org/ doc/essays/blurb/

[3]. https://wiki.python.org/moin/BeginnersGuide/ Overview

[4]http://blog.datacamp.com/wp-content/ uploads/2015/05/R-vs-Python-216-2.png

[5]. https://www.invensis.net/blog/it/applications-of-python-in-real-world/

[6]. https://www.kaggle.com/mcdonalds/nutrition-facts

[7]. http://www.randalolson.com/2014/06/28/how-to-make-beautiful-data-visualizations-in-python-with-matplotlib/