Exploratory Data Analysis Project Report on Chocolate Bar Ratings

Group Members

Elliot Kojo Attipoe

Isaac Armah-Mensah

Franklin Kome Amoo

**Introduction**

Exploratory Data Analysis (EDA) generally refers to the process of performing initial investigations on data with the aim of discovering patterns, anomalies, testing hypothesis or checking assumptions with the aid of descriptive statistics and graphical representations. EDA is part of the data science lifecycle. See figure 1.



Figure 1: Data science lifecycle, source: <http://sudeep.co/data-science/Understanding-the-Data-Science-Lifecycle/>

EDA is primarily used to see what data can reveal beyond the formal modeling or hypothesis testing task and provides a better understanding of data set variables and the relationships between them. It can also help determine if the statistical techniques you are considering for data analysis are appropriate. The main purpose of EDA is to help look at data before making any assumptions. It can help identify obvious errors, as well as better understand patterns within the data, detect outliers or anomalous events, find interesting relations among the variables.

**Steps involved in EDA**

Figure 2: EDA steps

The EDA process is all about using data exploratory techniques to understand the various aspects of a dataset. There are three steps in the EDA cycle which is must be followed and they are:

1. **Understand the data**

There is a need to understand the variables in the dataset. You must know the number of columns and rows you are dealing with their corresponding data types.

1. **Cleaning the data**

The data is clean of any redundancies and this could be any irregular variable or missing variables or null variables. These kinds of variables are not necessary for making any meaningful conclusions or interpretations. The data is again checked for duplicates and outliers as this has the potential to affect the accuracy of conclusions drawn from the dataset.

1. **Analysis of relationship between variables**

The final step in the EPA process is to check where there are any possible relationships between the variables in the dataset. Going through this cycle will help gain insight about the dataset.

**Source of dataset**

The dataset used for this project is the Chocolate Bar Ratings which is compiled by Brady Brelinski, Founding Member of the Manhattan Chocolate Society.

**Context**

Chocolate is one of the most popular candies in the world. Each year, residents of the United States collectively eat more than 2.8 billions pounds. However, not all chocolate bars are created equal! This dataset contains expert ratings of over 1,700 individual chocolate bars, along with information on their regional origin, percentage of cocoa, the variety of chocolate bean used and where the beans were grown.

**Flavors of Cacao Rating System:**

* 5= Elite (Transcending beyond the ordinary limits)
* 4= Premium (Superior flavor development, character and style)
* 3= Satisfactory(3.0) to praiseworthy(3.75) (well made with special qualities)
* 2= Disappointing (Passable but contains at least one significant flaw)
* 1= Unpleasant (mostly unpalatable)

Each chocolate is evaluated from a combination of both objective qualities and subjective interpretation. A rating here only represents an experience with one bar from one batch. Batch numbers, vintages and review dates are included in the database when known.

**Tools/libraries used**

The python Integrated development environment used for the project is the Jupyter Notebook and the main packages or libraries where Pandas, Seaborn and matplotlib.

**Hypothesis**

Over time, there has been a correlation between cocoa ratings and the proportion of cocoa in a chocolate bar, as the quality of cocoa beans and chocolate bars has improved.

### 

**Hypothetical Questions.**

* + - 1. Which countries produces the best cocoa beans?
      2. Which countries have the highest-rated chocolate bars?
      3. Is there relationship between the proportion of cocoa in chocolate bar and the rating?
      4. What are the top ten companies with the highest rating?
      5. What is the pattern over the years with respect to rating?
      6. Is there a correlation between a bean’s origin and the average rating of bars?
      7. Which chocolate beans have the highest ratings?

**Understanding the data**

First of all, the dataset is was loaded and afterward printed and this gave a fair idea of the dataset.

*The Flavors of Cacao dataset consists of 1795 rows and 9 columns.*

*The Columns are:*

| **Column** | **Description** |
| --- | --- |
| Company (Maker-if known) | Name of the company manufacturing the bar. |
| Specific Bean Originor Bar Name | The specific geo-region of origin for the bar. |
| REF | A value linked to when the review was entered in the database. Higher = more recent. |
| ReviewDate | Date of publication of the review. |
| CocoaPercent | Cocoa percentage (darkness) of the chocolate bar being reviewed. |
| CompanyLocation | Manufacturer base country. |
| Rating | Expert rating for the bar. |
| BeanType | The variety (breed) of bean used, if provided. |
| Broad BeanOrigin | The broad geo-region of origin for the bean. |

The nest step of the cleaning process was to understand the data types of the variables and the output is shown below:

---------------------------------------------

Data Types of all Columns CompanyÂ \n(Maker-if known) object

Specific Bean Origin\nor Bar Name object

REF int64

Review\nDate int64

Cocoa\nPercent object

Company\nLocation object

Rating float64

Bean\nType object

Broad Bean\nOrigin object

dtype: object

---------------------------------------------

The data type information reveals something about our data. The column names contain the "\n" or "newline" character, which will result in unidentifiable errors we have to go through each column name and rename it explicitly.

We went ahead to run some descriptive statistics about our dataset and following were realized:

* The mean review date is 2012 which shows that the data is centered around 2012.
* The most current review date is 2017 and the oldest date is 2006.
* This shows the data was collected for the period 2006-2017(a 11-year period)
* The average rating of chacolate bar 3.18, which is satisfactory (represent a well made bar with special qualities.)
* The rating of the bars is within a range of 1 to 5

**Cleaning the data**

The purpose of cleaning any dataset is to look of missing values and possible remove or replace them based on its effect on the data. We first did a count of null values and below was the output:

Missing Values:

---------------------------------------------

CompanyÂ \n(Maker-if known) 0

Specific Bean Origin\nor Bar Name 0

REF 0

Review\nDate 0

Cocoa\nPercent 0

Company\nLocation 0

Rating 0

Bean\nType 1

Broad Bean\nOrigin 1

dtype: int64

the results showed that the columns Bean\nType and Broad Bean\nOrigin had some missing values in the dataset. A printout of the last 10 records revealed that those two columns had some special character stored in some of the fields. Details shown below:

Bean\nType Broad Bean\nOrigin

1785 Criollo, Trinitario Papua New Guinea

1786 Â  Bolivia

1787 Â  Dominican Republic

1788 Forastero Congo

1789 Â  Â

1790 Â  Peru

1791 Forastero Congo

1792 Forastero India

1793 Â  India

1794 Â  Brazil

Several steps were taken to clean the dataset. The first was to rename the column header and secondly, to replace the missing values with none.

The following function was defined to replace all newline characters and spaces with an underscore. Details shown below:

*#cleaning column names*

cols\_names = list(data.columns)

*# Function to replace newline characters and spaces in the feature names*

**def** replace\_columnNames(column\_names):

replace\_names = []

**for** f **in** column\_names:

replace\_names.append(((f.casefold()).replace("**\n**","\_")).replace(" ","\_"))

**return** replace\_names

print("Column Names before Cleaning:")

print(cols\_names)

print("**\n**Column Names after Cleaning:")

print(replace\_columnNames(cols\_names))

in order to get a better understanding of the missing values, we probe further. We decided to look at the individual columns with the missing data. With the following lines of code:

*# Probing the Â character further*

print('Checking special character in bean\_type data')

print('----------------------------------------------')

print(data['bean\_type'].value\_counts().head())

print("Missing Spaces encoded as:")

list(data['bean\_type'][0:10])

Checking special character in bean\_type data

----------------------------------------------

Â  887

Trinitario 419

Criollo 153

Forastero 87

Forastero (Nacional) 52

Name: bean\_type, dtype: int64

Missing Spaces encoded as:

['Â\xa0',

'Â\xa0',

'Â\xa0',

'Â\xa0',

'Â\xa0',

'Criollo',

'Â\xa0',

'Criollo',

'Criollo',

'Â\xa0']

It was revealed that, *we have* ***887*** *instances in which "bean\_type" is encoded as a special character Â and space or Â\xa0.*

*The missing values for both columns were subsequently replaced with ‘none’ and a displayed of the dataset after is shown below:*

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 1795 entries, 0 to 1794

Data columns (total 9 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Company 1795 non-null object

1 specific\_bean\_origin\_or\_bar\_name 1795 non-null object

2 ref 1795 non-null int64

3 review\_date 1795 non-null int64

4 cocoa\_percent 1795 non-null object

5 company\_location 1795 non-null object

6 rating 1795 non-null float64

7 bean\_type 1795 non-null object

8 broad\_bean\_origin 1795 non-null object

dtypes: float64(1), int64(2), object(6)

memory usage: 126.3+ KB

The indication was that there was no missing data. The final process of the cleaning the dataset was to delete columns which in no way had any impact in the dataset. The ‘ref’ column was identified not to have any impact on the dataset and was subsequently dropped.

**Analyzing relationships between variables**

A correlation matrix and pairplot showed that there no relationships between the variables. Figure 3 show the output of the correlation matrix.

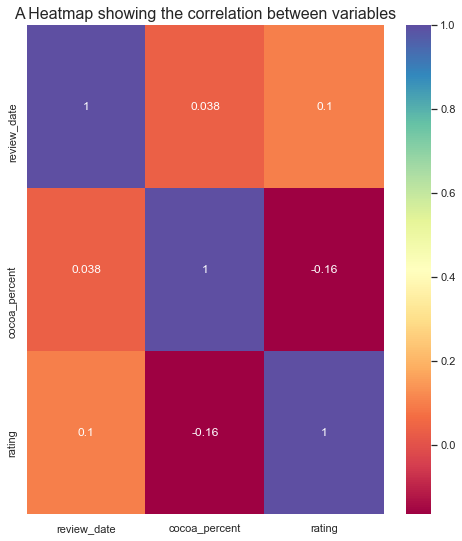


Figure 3: correlation matrix

Again a subsequent analysis using pairplot showed a similar relationship as the correlation matrix as shown in figure 4.

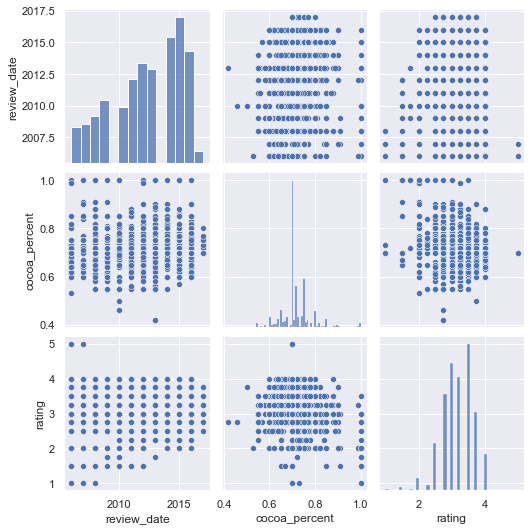


Figure 4: Pairplot

**Questions**

**Recommendation**

**Conclusions**