42047: Data Processing with Python

Assignment (Part C)

Report: Data Analysis and Visualization

Student Name and ID: NIPUNN KHURANA 25009200

Date: 30/10/2023

Table of Contents

ABSTRACT……………………………………………..3

1. Introduction and Background…………………………3

2. Overview of Data Analysis Pipeline…………………...4

3. Discussion and Conclusion…………………………….16

ABSTRACT

The Spam base dataset provides a comprehensive collection of features extracted from emails, facilitating in the identification of spam messages. Our analysis dives deep into patterns and correlations among these features, revealing distinct behaviours between spam and non-spam emails. By visualizing word frequencies, special character usage, and capital letter sequences, we derive insights that extend beyond mere spam classification. Such insights can potentially guide email marketing strategies and content creation, ensuring messages echo genuinely with recipients without triggering spam flags.

1. INTRODUCTION AND BACKGROUND

1.1 Problem Solved in this paper

The agenda of this paper was to solve was to determine the relationship of words, characters, and capital letters with each other given that the email is spam or not.

1.2 Business Question

Through Exploratory Data Analysis (EDA), we seek to answer the pressing business question: **What inherent patterns and features distinguish spam from genuine emails**? By dissecting word frequencies, character usage, and other embedded features, we aim to unravel the underlying characteristics of these emails. The insights drawn will not only enhance spam detection mechanisms but also provide guidance for crafting emails that effectively reach their intended audience without being misconstrued as spam.

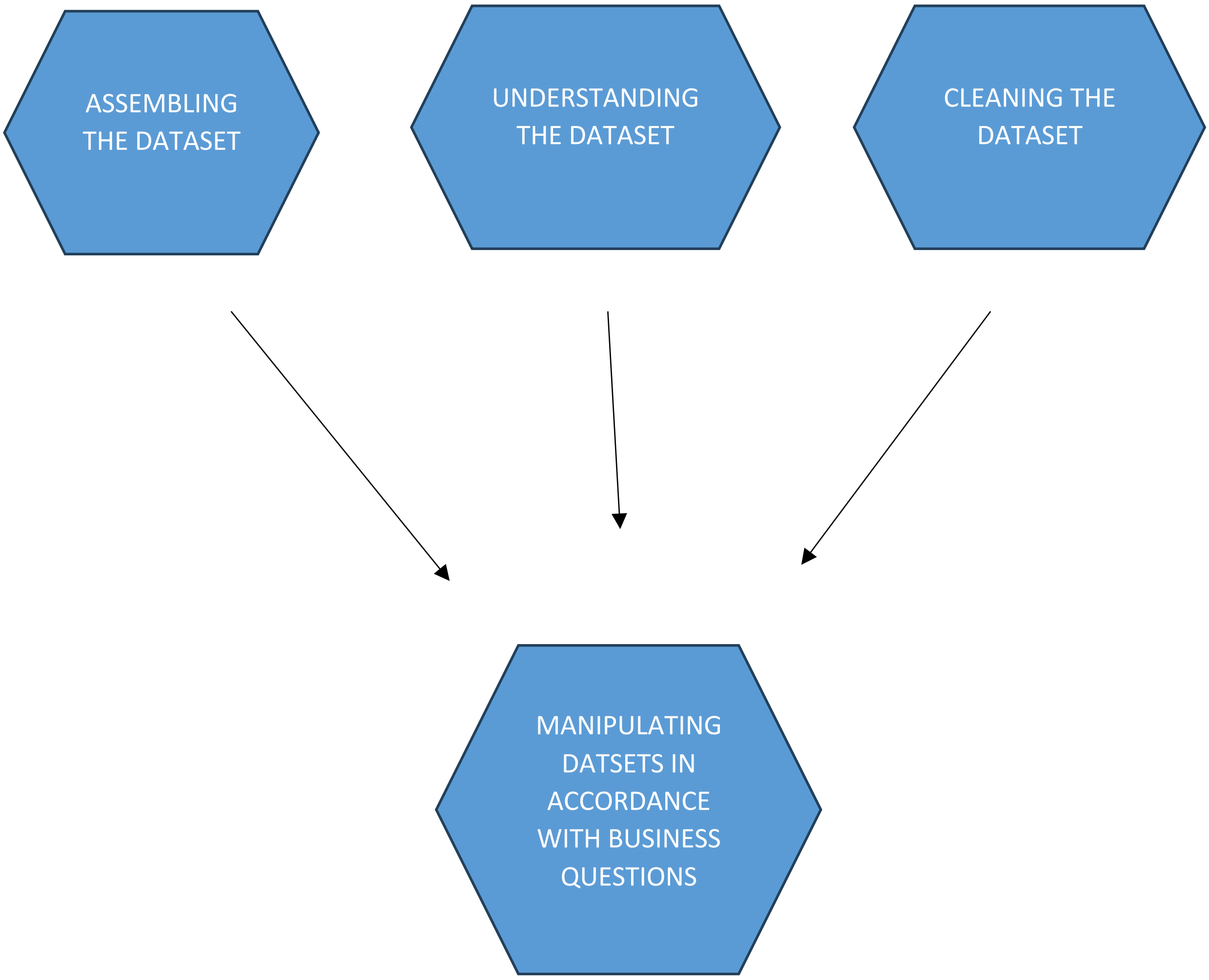
1.3 Dataset

The Spambase dataset is a curated collection designed to facilitate spam detection research. It contains 4,601 samples, each characterized by 57 attributes—primarily word frequencies and character usage within email content. The features encapsulate common terms and patterns observed in emails, providing a comprehensive view of their structure. This dataset was sourced from the UCI Machine Learning Repository, a trusted resource for the data science community. Further details and attributions can be found in the repository's publication: https://archive.ics.uci.edu/dataset/94/spambase

To address the business question pertaining to the Spambase dataset, we primarily focused on two groups of attributes: the frequency variables and character variables. The frequency variables capture the occurrence rates of specific words within emails, providing insights into common terminologies associated with spam and non-spam content.

2. Overview of the Data Analysis Pipeline

2.1 Flow Diagram/Flowchart/Workflow


ASSEMBLING THE DATASET
UNDERSTANDING THE DATASET
CLEANING THE DATASET
MANIPULATING DATSETS IN ACCORDANCE WITH BUSINESS QUESTIONS

2.2 Data Preparation

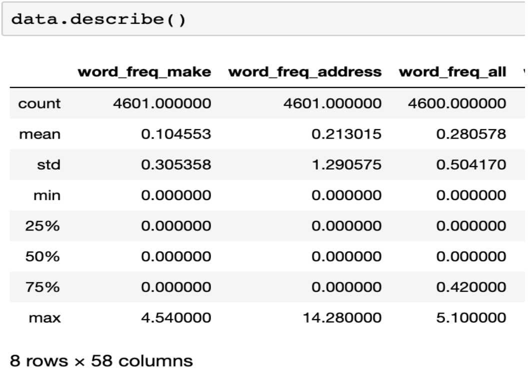
1. **Renaming Columns for Readability and Accessibility**: The process we undertook began with renaming the columns. By shifting the previous column names to the first row, we ensured that each column is clearly labelled with its content, making the dataset more interpretable and easier to manipulate in subsequent analyses.

2. **Initial Inspection with df.head() and df.tail():** These functions are essential for a preliminary review of the dataset. While df.head() displays the first few rows, df.tail() provides the last few rows. This initial glance offers insights into the kind of data entries present, possible missing values, and the general structure of the dataset.

3. **Understanding Dataset Structure and Integrity with df.shape and df.info():** The df.shape function offers a quick view of the number of rows and columns, which is vital for understanding the dataset's dimensions. In contrast, df.info() provides a comprehensive summary, including data types of each column and the number of non-null values. It was through this method that we identified certain columns having non-numeric data types, which led to the next step.

4. **Data Type Conversion**: For many analytical processes, especially statistical methods, it's essential to ensure that data types are consistent and relevant to the nature of the data. Upon noticing that some columns were of the 'Object' data type, we converted them to a numeric type, ensuring compatibility with various analytical methods and enhancing calculative efficiency.

5. **Descriptive Analysis with df.describe():** This function is a powerhouse, providing a quick summary of the central tendencies, dispersion, and shape of the dataset’s distribution. It returns valuable statistics such as mean, median, standard deviation, and the minimum and maximum values for each column. This step is particularly crucial, as understanding the distribution and spread of data can guide the choice of analytical methods and reveal potential outliers or anomalies in the data.



```
data.describe()
```

	word_freq_make	word_freq_address	word_freq_all
count	4601.000000	4601.000000	4600.000000
mean	0.104553	0.213015	0.280578
std	0.305358	1.290575	0.504170
min	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000
50%	0.000000	0.000000	0.000000
75%	0.000000	0.000000	0.420000
max	4.540000	14.280000	5.100000

8 rows × 58 columns

TABLE.1 Statistical Summary of Variables

2.3 MISSING VALUE EXPLORATION

2.3.1 MISSING VALUE HANDLING PROCEDURE

Identifying Missing Values: Initially, to detect the presence of any missing values in the dataset, we used the isnull() method. This method returns a similar structured DataFrame but with boolean values, indicating whether a value is missing (True) or not (False). By chaining this with the sum() method, we calculated the total number of missing values for each variable. The result, stored in missing_values_count, gave a clear picture of which variables contained missing data and how many missing entries they had.

Quantifying Missing Data: Beyond merely identifying the presence of missing values, it's crucial to understand their significance in relation to the dataset's size. The missing_percentage calculation achieved this, representing the proportion of missing data for each variable as a percentage of the dataset's total length. Such a metric is instrumental in deciding whether to retain or discard a variable, based on the extent of its missing data.

Visualizing Missing Values: To get a better and more intuitive grasp of the extent and distribution of missing values, we visualized the data using a bar graph. This visualization (FIG 1.) displayed all variables on the x-axis, with the height of the bars representing the number of missing values.

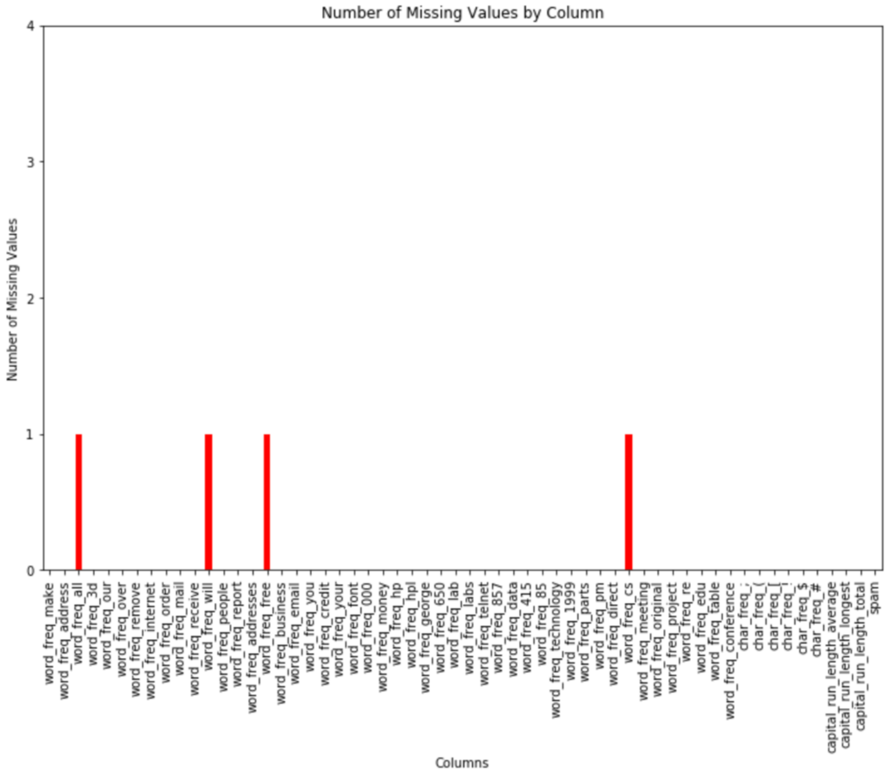


FIG 1. Visualisation of Missing Values

Addressing Missing Values: After identifying the columns, 'word_freq_all', 'word_freq_will', 'word_freq_free', and 'word_freq_cs' as having one missing value each, the decision was made to impute these missing values. Using the fillna() method, we filled the missing values with the mean of their respective columns. Since there was only one missing value in each identified column, the impact of this imputation on the dataset's overall statistics was minimal.

2.3.2 DROPPING DUPLICATE ROWS

After handling the missing values, the next step was to remove duplicate rows. First, we identified the duplicate rows by applying data.duplicated() method. We found out that there were 391 duplicate records. We removed them by using the data.drop_duplicate() method.

2.4 OUTLIER IDENTIFICATION

Visual Detection of Outliers Using Box Plots:

By plotting box plots for all 58 variables in the dataset, we were able to visually inspect the spread of data, interquartile range, and potential outliers. In FIG 2. we can see the variables 'capital_run_length_average', 'capital_run_length_longest', and 'capital_run_length_total' not only deviated from the percentage range but also contained potential outliers.

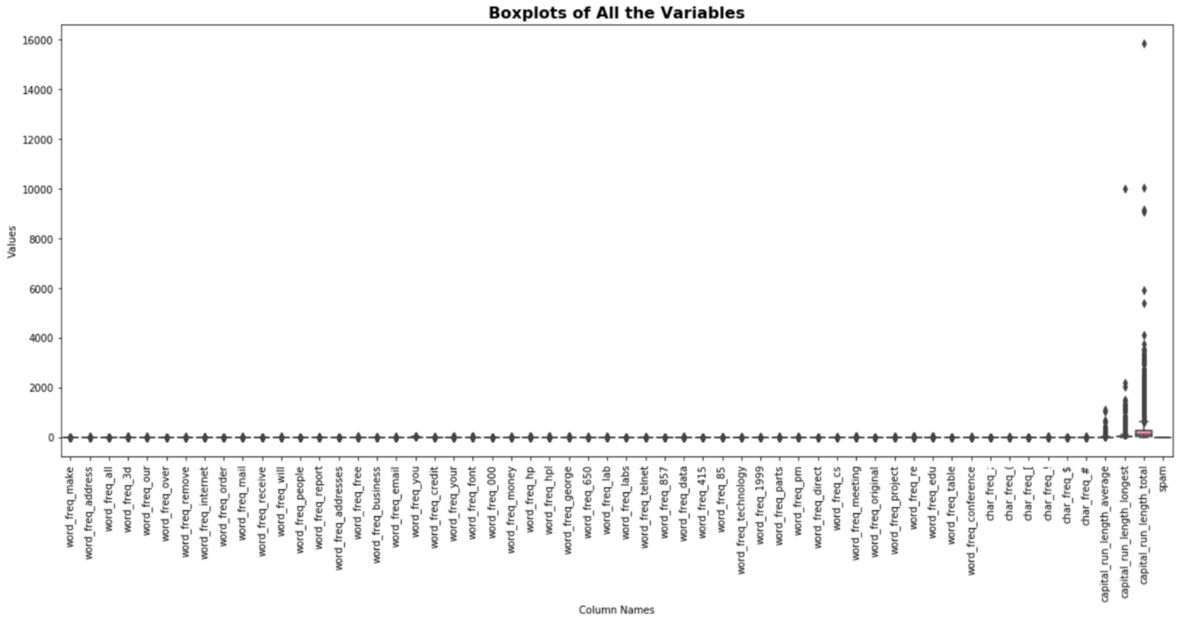


FIG 2. Box plots of all the variables in the Spambase dataset

Subsetting for Detailed Inspection:

We separated them from the rest to facilitate a more focused analysis. This subset was then visualized using a dedicated box plot, further corroborating the presence of outliers in Capital Letter Variables.

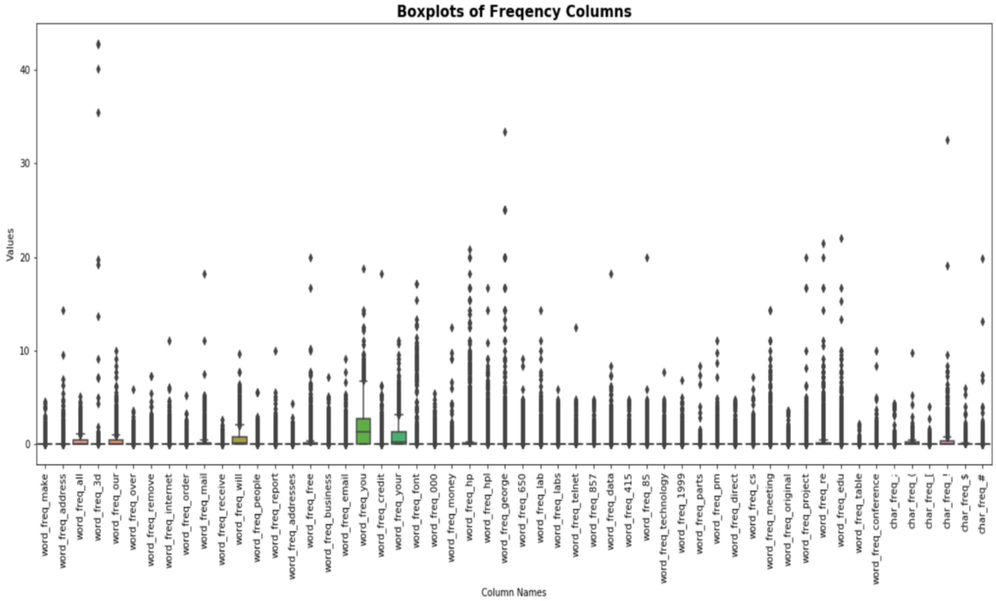


FIG 3. Boxplots of Frequency Columns

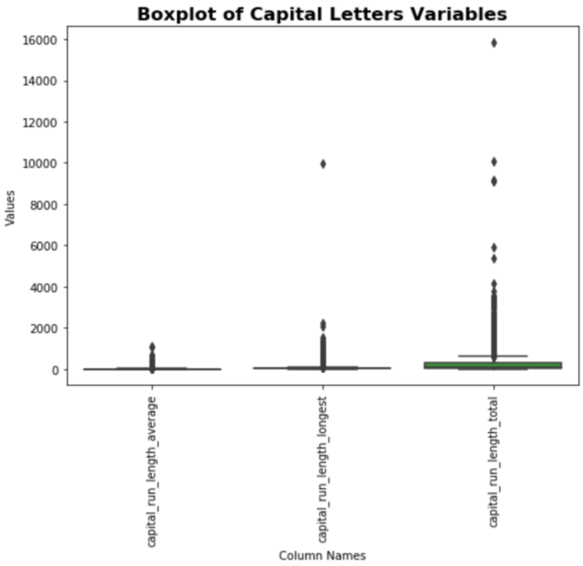


Fig 4. Box plot of Capital Letter Variables only

Quantitative Outlier Removal:

We created a function named outlier remover that employs the Interquartile Range (IQR) method. The function calculates the first (Q1) and third quartiles (Q3) of the data. The IQR is the difference between Q3 and Q1, which represents the middle 50% of the data. Any data point outside the range [(Q1 - 1.5 * IQR) to (Q3 + 1.5 * IQR)] is generally considered an outlier. The function then filters the dataset, retaining only those rows where the values of the specified columns lie within this acceptable range.

Implementation on the Dataset:

By passing the three columns to this function, we ensured that outliers present in these columns led to the removal of their respective rows from the dataset. This approach ensures that the final dataset, df_cleaned, is devoid of the outliers in these critical columns and is primed for robust analysis.

2.5 DATA VISUALISATION

<u>Q1. Which specific words are over-represented in the dataset, and can this tell us anything about common themes or topics in spam and non-spam emails?</u>

To solve this problem, we split the dataset into ‘spam emails’ and ‘non-spam emails’ and created two bar charts for the same for all the word frequency variables:

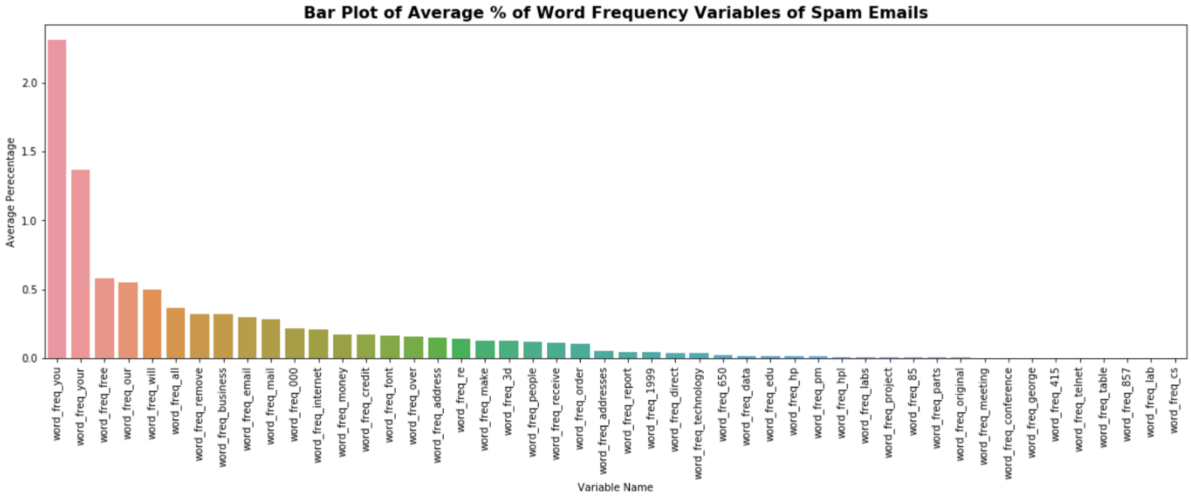


FIG 5. Bar Plot of Spam Email Dataset

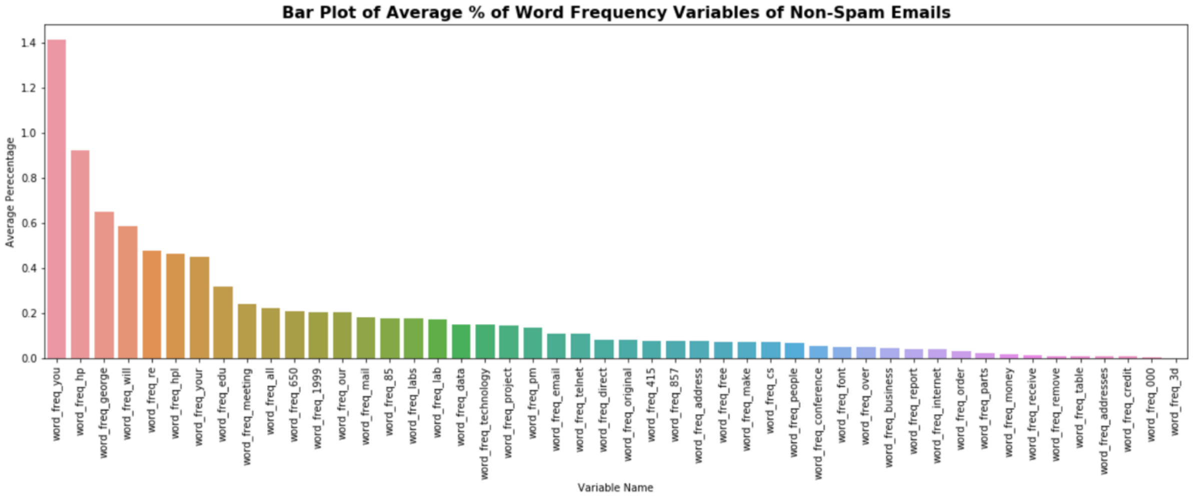


FIG 6. Bar Plot of Non-Spam Emails Dataset

From the above two charts we can observe the following:

1. Words like ‘you’, ‘your’, ‘free’, ‘our’ and ‘will’ have average frequency rate over 0.5% in the spam emails (Fig 5). We can comment about this finding that **“a spam email has at least 0.5% occurrence of the word ‘free’ to attract their potential victims”.**

2. Words like ‘you’, ‘hp’ & ‘george’ have average frequency rate over 0.6 % in the non-spam emails (Fig 6). A basic comment about this observation could be **“0.6% of an email has the word ‘george’ (or written by George) in a random non-spam email”.**

To understand the relationship of the spam email, we created two pair plots for top 3 and bottom 3 averages of occurrences of word variables for the spam emails subset.

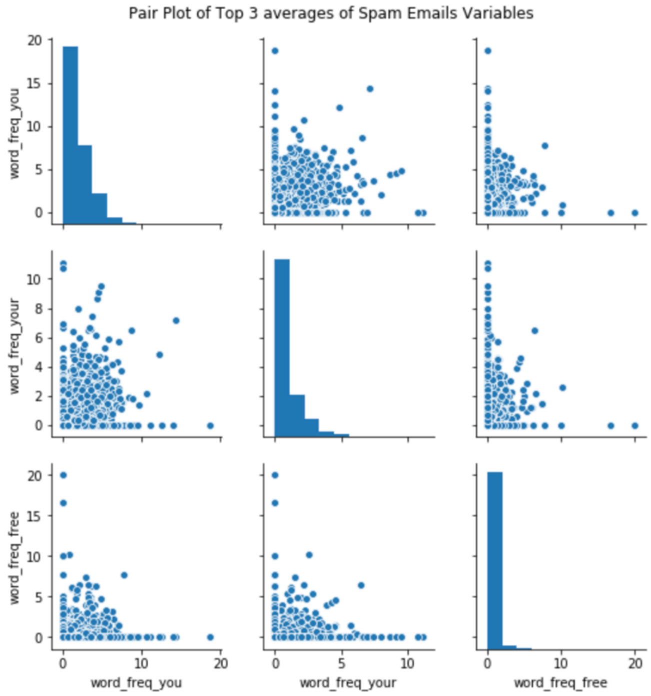


FIG 7. Pair Plot of Top 3 Word Variables Averages

From the above pair plot, we can comment that the density of scatter plot is high with the occurrence of words like ‘you’ and ‘your’ which means that these two words are commonly used together for any random spam email.

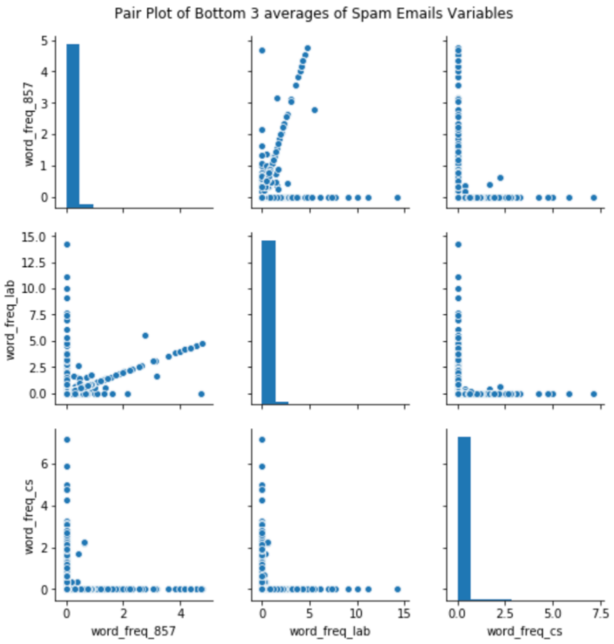
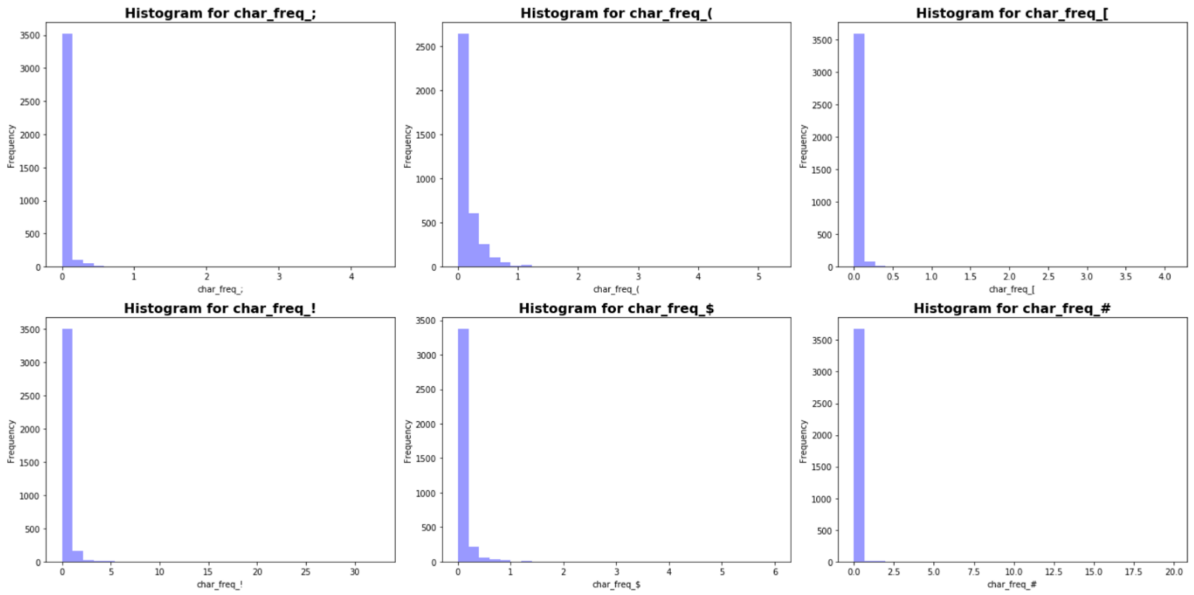


FIG 8. Pair Plot of Bottom 3 Word Variables Averages

A basic comment about the above graph could be there is a slight linear relationship between the words ‘lab’ and ‘857’ in few spam emails.

Q2. How does the frequency of special characters vary in the dataset, and what might this indicate about the content of the emails?

To conduct analysis of the above question, we will dive into the distribution of the ‘char’ value variables of the dataset.



Histogram for char_freq_;

Histogram for char_freq_(

Histogram for char_freq_[

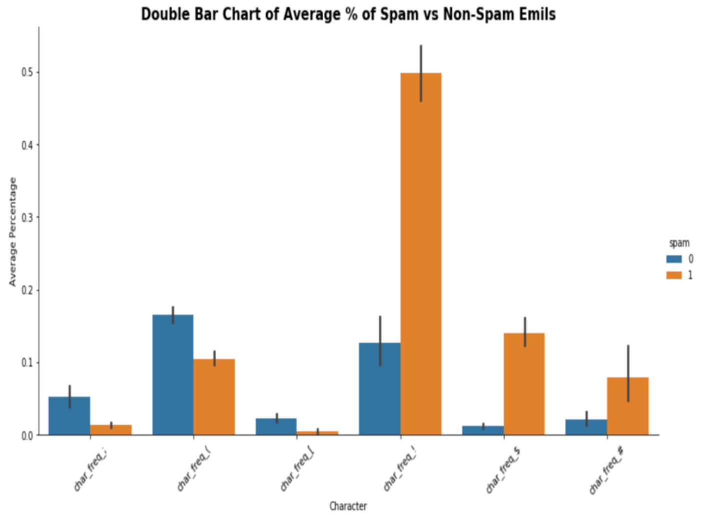
Histogram for char_freq_!

Histogram for char_freq_$

Histogram for char_freq_#

FIG 9 Distribution Plots for Char Freq Values

All the char variables in the dataset have most of their distribution between 0 to 1, but mostly the values are closer to zero.



Double Bar Chart of Average % of Spam vs Non-Spam Emils

FIG 10. Double Bar Chart of Spam Vs Non-Spam Email

We created a double bar chart of char values in comparison to spam and non-spam emails. In Fig 10 we can see that the bars showing true spam email (spam = 1) have more than 3 times the average frequency % for the characters “!”, “$” & “#” in comparison to the non-spam emails. A basic comment about the occurrence of "$" in spam emails could be that **"scammers use dollar sign ($) character more in their spam emails which shows an intent to attract the potential victim in their financial scam"**.

Q3. Is there a pattern between the frequency of capital letters and other specific words in emails?

To answer this question, we have created a heatmap (correlation matrix) for the variable 'capital_run_length_total' (it tells the total nubmer of capital letters in an email) against all the word variables.

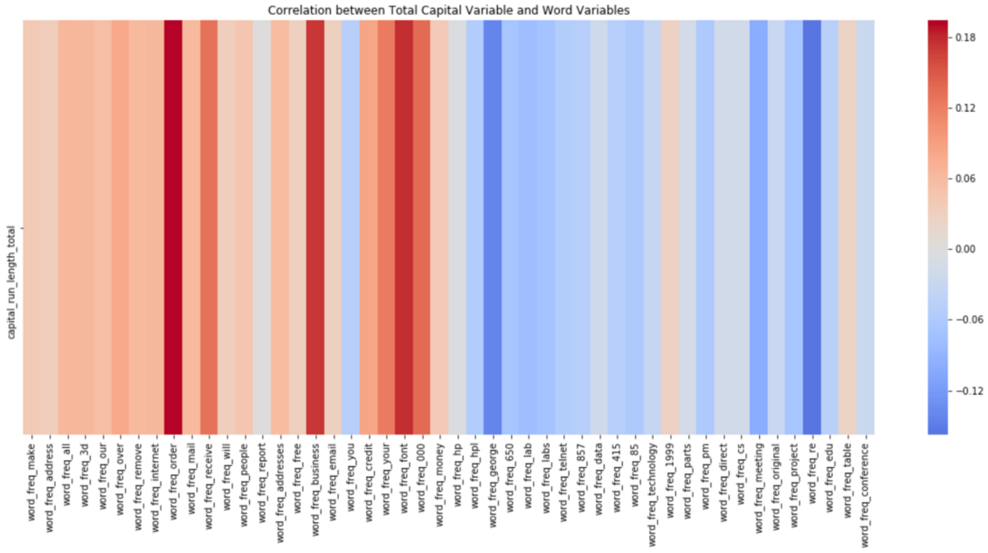
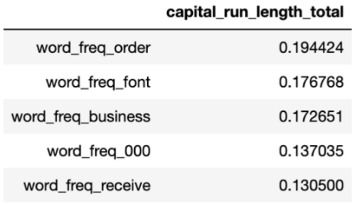
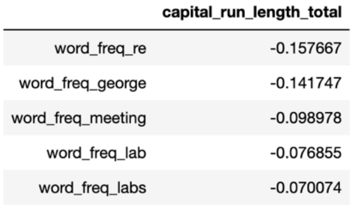


FIG 11. Heatmap

The above Fig 11. shows a range of approximately (-0.12, 0.18), the variables with positive correlation are shown in red and the variables with negative correlation are shown in blue.



	capital_run_length_total
word_freq_order	0.194424
word_freq_font	0.176768
word_freq_business	0.172651
word_freq_000	0.137035
word_freq_receive	0.130500



	capital_run_length_total
word_freq_re	-0.157667
word_freq_george	-0.141747
word_freq_meeting	-0.098978
word_freq_lab	-0.076855
word_freq_labs	-0.070074

TABLE 2.1 Top 5 correlations

TABLE 2.2 Bottom 5 correlations

Table 2.1. shows the most 5 and Table 2.2 shows least 5 correlated values between word variables and ‘capital_run_length_total’. It can be summarized that **whenever the frequency of words like ‘order’, ‘font’, ‘business’, ‘000’ and ‘receive’ are used in the email, the total number of capital letters increases.**

3. DISCUSSION AND CONCLUSION

The Spambase dataset underwent a thorough examination. Columns were renamed for clarity, revealing inconsistent data types that were rectified. Missing values in specific columns were filled with column means, and 391 duplicate records were removed. A comprehensive box plot analysis pinpointed potential outliers in 'capital_run_length_average', 'capital_run_length_longest', and 'capital_run_length_total'. An outlier removal function (IQR method) was built to address these differences.

It was observed that spam emails often contained words like 'free', 'will', and 'you', possibly as tactics to attract victims. Conversely, non-spam emails commonly featured words like 'hp', 'george', and 'you', suggesting frequent mentions by someone named George. In essence, the analysis meticulously unearthed patterns and insights critical to discerning spam from non-spam emails.

Upon segregating data into spam and non-spam emails, double bar charts were plotted. Notably, spam emails displayed a thrice higher frequency of characters "!", "$", and "#". This observation underscores the potential lure of financial scams in spam emails. Correlation tables indicated a relationship between certain word frequencies and the total number of capital letters, implying that words like 'order', 'font', and 'business' correlate with increased capital letter usage.