NAME: JANHAVI CHALAK

DIV:ET-2

ROLL NO.:ET2-35

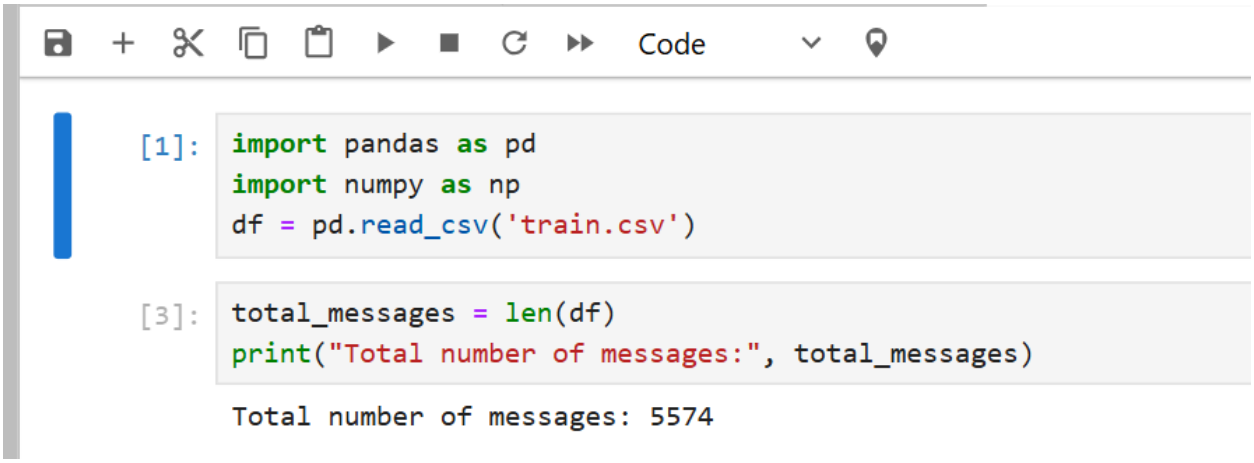
PRN:202401070194

TOPIC: SMS DATA COLLECTION

DATASET LINK:
https://drive.google.com/file/d/1_oz5N8tHOHJ9945AoYxq6NdVxBVnMu2n/view?usp=drivesdk

PROBLEM STATEMENT AND SOLUTIONS:

1. Total number of messages

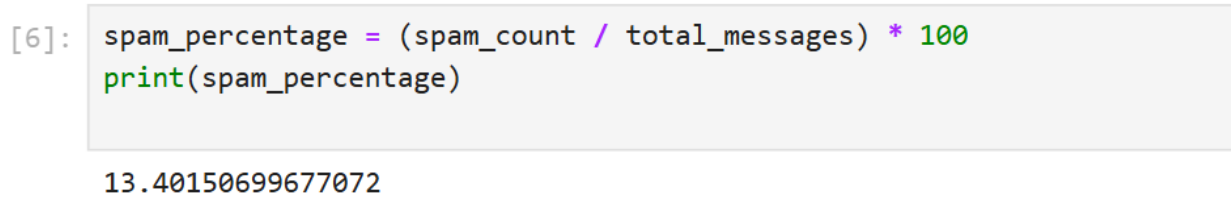


```
[1]: import pandas as pd
     import numpy as np
     df = pd.read_csv('train.csv')

[3]: total_messages = len(df)
     print("Total number of messages:", total_messages)

     Total number of messages: 5574
```

2. Number of spam and ham messages

```
[5]: spam_count = df['label'].sum()
     ham_count = total_messages - spam_count
     print("Spam:", spam_count, "Ham:", ham_count)

     Spam: 747 Ham: 4827
```

3. Percentage of spam messages



```
[6]: spam_percentage = (spam_count / total_messages) * 100
     print(spam_percentage)

     13.40150699677072
```

4. Average length of SMS messages

```
df['length'] = df['sms'].apply(len)
average_length = df['length'].mean()
print(average_length)
```

```
81.47829207032652
```

5. Average length of spam vs ham messages

```
average_length_spam = df[df['label'] == 1]['length'].mean()
average_length_ham = df[df['label'] == 0]['length'].mean()
print("Spam:", average_length_spam, "Ham:", average_length_ham)
```

```
Spam: 139.6760374832664 Ham: 72.47192873420344
```

6. Maximum length of an SMS message

```
max_length = df['length'].max()
print(max_length)
```

```
911
```

7. Minimum length of an SMS message

```
min_length = df['length'].min()
print(min_length)
```

```
3
```

8. SMS with maximum number of characters

```
sms_max = df[df['length'] == max_length]['sms'].values[0]
print(sms_max)
```

```
For me the love should start with attraction.i should feel that I need her every time around me.she should be the first thing w
hich comes in my thoughts.I would start the day and end it with her.she should be there every time I dream.love will be then wh
en my every breath has her name.my life should happen around her.my life will be named to her.I would cry for her.will give all
my happiness and take all her sorrows.I will be ready to fight with anyone for her.I will be in love when I will be doing the c
raziest things for her.love will be when I don't have to proove anyone that my girl is the most beautiful lady on the whole pla
net.I will always be singing praises for her.love will be when I start up making chicken curry and end up makiing sambar.life w
ill be the most beautiful then.will get every morning and thank god for the day because she is with me.I would like to say a lo
t..will tell later..
```

9. SMS with minimum number of characters

```
sms_min = df[df['length'] == min_length]['sms'].values[0]
print(sms_min)
```

```
Ok
```

10. Number of messages with more than 100 characters

```
messages_above_100 = df[df['length'] > 100].shape[0]
print(messages_above_100)
```

```
1775
```

11. Most common word across all SMS

```
import re
from collections import Counter

def get_words(text_series):
    all_text = ' '.join(text_series).lower()
    words = re.findall(r'\b\w+\b', all_text)
    return words

all_words = get_words(df['sms'])
most_common_word = Counter(all_words).most_common(1)[0]
print(most_common_word)
```

```
('i', 3021)
```

12. Average number of words in spam messages

```
]: spam_words = df[df['label'] == 1]['sms'].apply(lambda x: len(x.split()))
   average_words_spam = spam_words.mean()
   print(average_words_spam)
```

```
23.91164658634538
```

13. Average number of words in ham messages

```
ham_words = df[df['label'] == 0]['sms'].apply(lambda x: len(x.split()))
average_words_ham = ham_words.mean()
print(average_words_ham)
```

```
14.304122643463849
```

14. Number of messages that contain the word "free"

```
messages_with_free = df['sms'].str.contains(r'\bfree\b', case=False).sum()
print(messages_with_free)
```

```
229
```

15. Number of spam messages that contain the word "free"

```
spam_with_free = df[(df['label'] == 1) & (df['sms'].str.contains(r'\bfree\b', case=False))].shape[0]
print(spam_with_free)
```

```
170
```

16. Percentage of messages that contain numbers

```
messages_with_numbers = df['sms'].str.contains(r'\d').sum()
percentage_with_numbers = (messages_with_numbers / total_messages) * 100
print(percentage_with_numbers)
```

```
26.264800861141012
```

17. Longest word across all SMS messages

```
longest_word = max(all_words, key=len)
print(longest_word)
```

```
hypotheticalhuagauahahuagahyuhagga
```

18. Number of unique words across all messages

```
unique_words_count = len(set(all_words))
print(unique_words_count)
```

```
8753
```

19. Top 10 most common words in spam messages

```
spam_messages_words = get_words(df[df['label'] == 1]['sms'])
top_10_spam_words = Counter(spam_messages_words).most_common(10)
print(top_10_spam_words)
```

```
[('to', 691), ('a', 380), ('call', 355), ('you', 297), ('your', 264), ('free', 224), ('2', 207), ('the', 206), ('for', 204)
('now', 199)]
```

20. Average number of special characters per message

```
]: special_characters_count = df['sms'].apply(lambda x: len(re.findall(r'[^A-Za-z0-9\s]', x)))
   average_special_chars = special_characters_count.mean()
   print(average_special_chars)
```

```
4.308575529242914
```