spamfilter_template

October 17, 2024

0.1 Import the dataset

[1]:
```
from ucimlrepo import fetch_ucirepo

# fetch dataset
spambase = fetch_ucirepo(id=94)

# data (as pandas dataframes)
X_all_features = spambase.data.features

# only select features that interest us
selected_features = ['word_freq_will', 'word_freq_remove', 'word_freq_you',␣
 ↪'word_freq_free', 'char_freq_!', 'char_freq_$']
X = X_all_features[selected_features]

# define y which contains the information whether the email is spam
y = spambase.data.targets
```

[2]:
```
# print the features
X
```

[2]:

	word_freq_will	word_freq_remove	word_freq_you	word_freq_free	\
0	0.64	0.00	1.93	0.32	
1	0.79	0.21	3.47	0.14	
2	0.45	0.19	1.36	0.06	
3	0.31	0.31	3.18	0.31	
4	0.31	0.31	3.18	0.31	
…	…	…	…	…	
4596	1.88	0.00	0.62	0.00	
4597	0.00	0.00	6.00	0.00	
4598	1.80	0.00	1.50	0.00	
4599	0.32	0.00	1.93	0.00	
4600	0.00	0.00	4.60	0.00	

	char_freq_!	char_freq_$
0	0.778	0.000
1	0.372	0.180
2	0.276	0.184

```
3           0.137        0.000
4           0.135        0.000
…           …            …
4596        0.000        0.000
4597        0.353        0.000
4598        0.000        0.000
4599        0.000        0.000
4600        0.125        0.000

[4601 rows x 6 columns]
```

[3]:
```
# print whether spam or not: 1 = spam, 0 = non-spam
y
```

[3]:
```
      Class
0         1
1         1
2         1
3         1
4         1
…       …
4596      0
4597      0
4598      0
4599      0
4600      0

[4601 rows x 1 columns]
```

0.2 Split data into training and test data

[4]:
```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=42)
```

[5]:
```
X_train_spam = X_train[y_train["Class"] == 1]
X_train_nonspam = X_train[y_train["Class"] == 0]
X_test_spam = X_test[y_test["Class"] == 1]
X_test_nonspam = X_test[y_test["Class"] == 0]

n_training = len(X_train)
n_test = len(X_test)
```

0.3 Exploratory histograms of training data

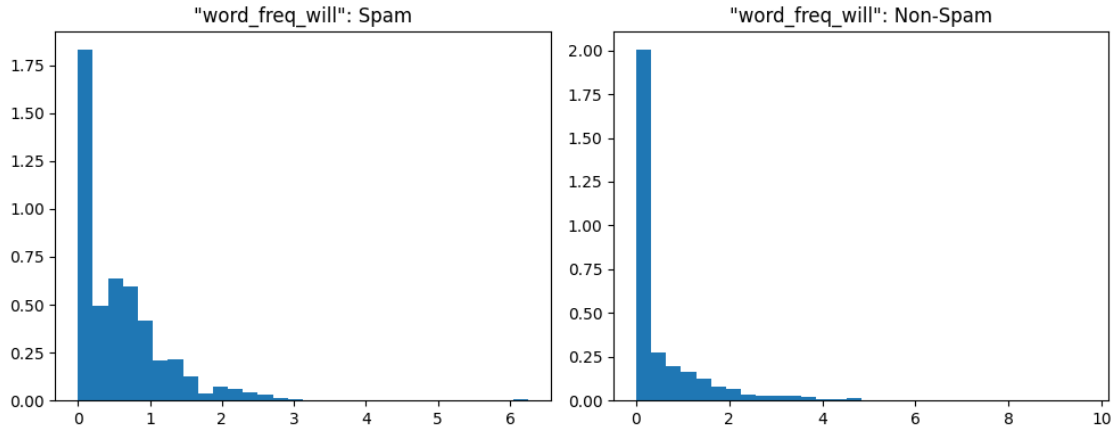
[6]:
```
import matplotlib.pyplot as plt
```

Define a function to plot histograms.

[7]:
```
def show_histogram(feature):
    # Create a figure with 2 subplots (1 row, 2 columns)
    fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 4))

    # Plot the first histogram with spam
    ax1.hist(X_train_spam[feature], bins=30, density=True) # density=True so␣
 ↪that the area under the histogram sums to 1
    ax1.set_title(f"\"{feature}\": Spam")

    # Plot the second histogram with non-spam
    ax2.hist(X_train_nonspam[feature], bins=30, density=True) # density=True so␣
 ↪that the area under the histogram sums to 1
    ax2.set_title(f"\"{feature}\": Non-Spam")

    plt.tight_layout()
    plt.show()
```

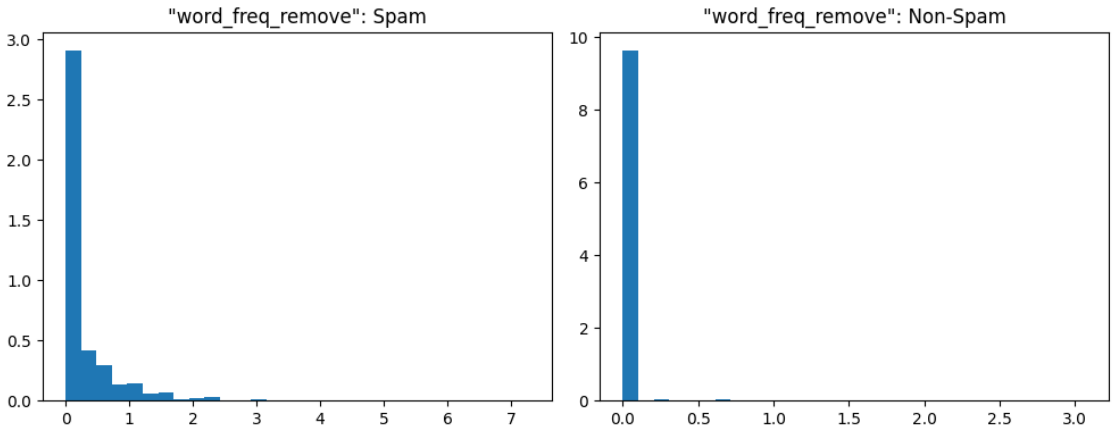
Occurences of “will”

[8]:
```
show_histogram('word_freq_will')
```



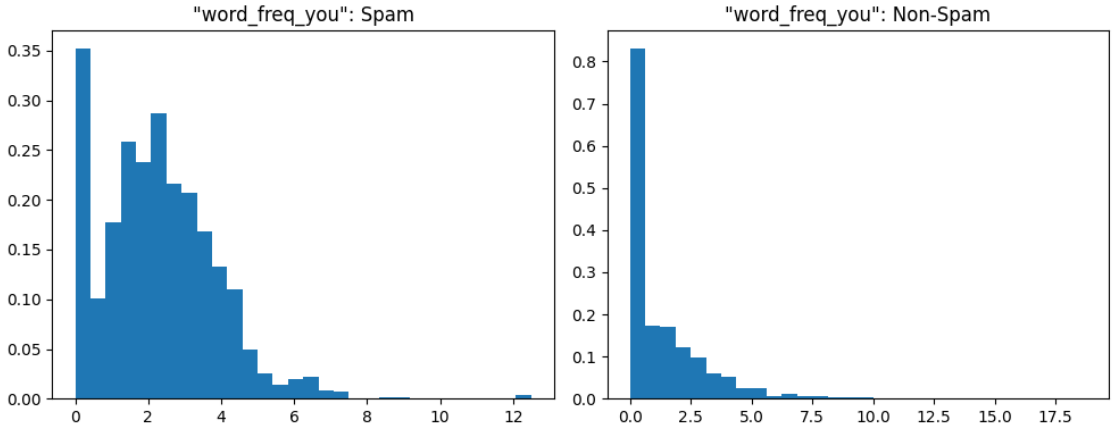
Occurences of “remove”

[9]:
```
show_histogram('word_freq_remove')
```



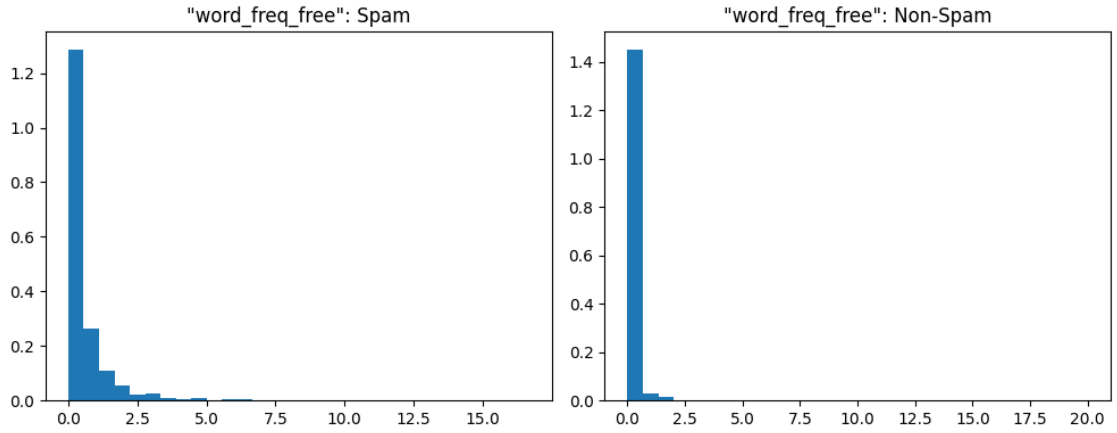
Occurences of "you"

[10]:
```
show_histogram('word_freq_you')
```



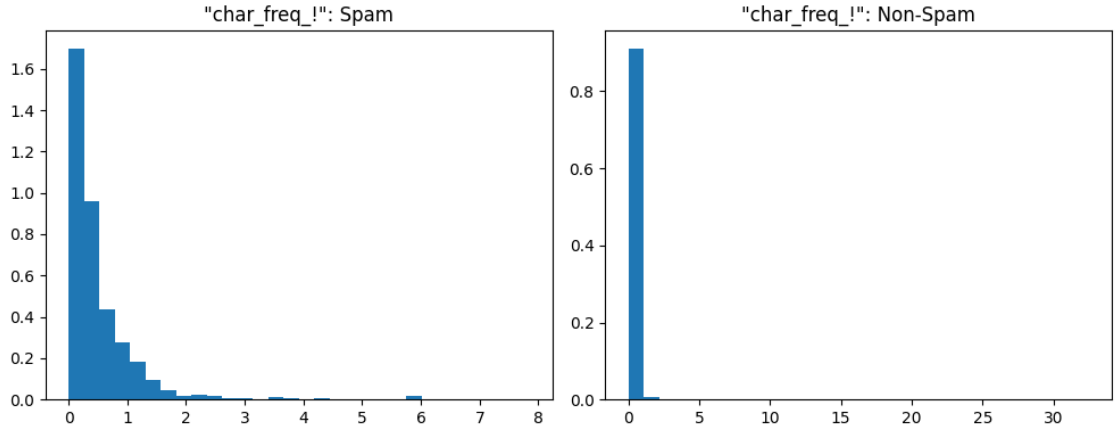
Occurences of "free"

[11]:
```
show_histogram('word_freq_free')
```



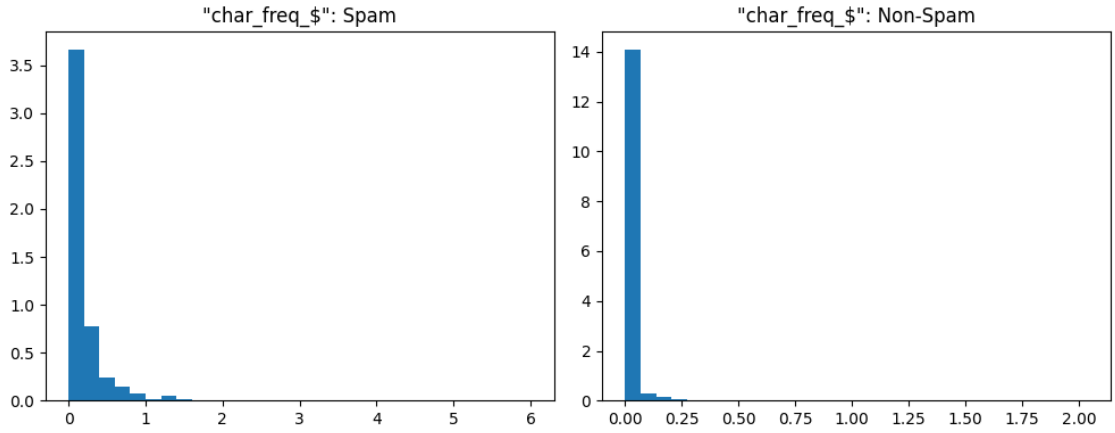
Occurences of "!"

[12]:
```
show_histogram('char_freq_!')
```



Occurences of "$"

[13]:
```
show_histogram('char_freq_$')
```


"char_freq_$": Spam
"char_freq_$": Non-Spam
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0
1
2
3
4
5
6
0
2
4
6
8
10
12
14
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00