ASDS2 Exercise 1.2 Preprocessing

April 23, 2025

[40]:
```
# Importing packages

import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.feature_extraction.text import CountVectorizer
import re
import string
import matplotlib.pyplot as plt
import datetime as dt

import nltk

#You may need to download the following to run this code:
nltk.download('stopwords')
nltk.download('wordnet')
nltk.download('punkt_tab')
nltk.download('averaged_perceptron_tagger_eng')

from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from nltk.stem import LancasterStemmer
from nltk.stem import WordNetLemmatizer
from nltk import word_tokenize, pos_tag
from nltk.corpus import wordnet

from collections import defaultdict
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\norag\AppData\Roaming\nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\norag\AppData\Roaming\nltk_data…
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package punkt_tab to
[nltk_data]     C:\Users\norag\AppData\Roaming\nltk_data…
[nltk_data]   Package punkt_tab is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger_eng to
[nltk_data]     C:\Users\norag\AppData\Roaming\nltk_data…
```

```
[nltk_data]   Unzipping taggers\averaged_perceptron_tagger_eng.zip.
```

1 Advanced Social Data Science 2 (ASDS2) Exercises

1.1 April 21: Preprocessing

1.1.1 1: Importing data without preprocessing

1. Download the data set available here, which contains the nearly 6,000 times Donald Trump insulted someone on Twitter: https://www.kaggle.com/ayushggarg/all-trumps-twitter-insults-20152021
2. Load the csv as a data frame using pandas.

[2]:
```
trump = pd.read_csv("data/trump_insult_tweets_2014_to_2021.csv")
print(trump.shape)
trump.head()
```

```
(10360, 5)
```

[2]:

	Unnamed: 0	date	target	insult	tweet
0	1	2014-10-09	thomas-frieden	fool	Can you believe this fool, Dr. Thomas Frieden …
1	2	2014-10-09	thomas-frieden	DOPE	Can you believe this fool, Dr. Thomas Frieden …
2	3	2015-06-16	politicians	all talk and no action	Big time in U.S. today - MAKE AMERICA GREAT AG…
3	4	2015-06-24	ben-cardin	It's politicians like Cardin that have destroy…	Politician @SenatorCardin didn't like that I s…
4	5	2015-06-24	neil-young	total hypocrite	For the nonbeliever, here is a photo of @Neily…

3. The variable 'target' has an indicator for the target of the insult. The data reveals that Trump's most frequent insult target is 'the media' ('the-media' in the data). Create a binary indicator for whether Trump targets the media. Fit a linear regression with this binary indicator as the dependent variable and the date of the tweet as the independent variable. Does Trump become more or less likely to insult the media over time? Why might this be?

[3]:
```
# get most frequent insult targets
trump['target'].value_counts()
```

[3]:
```
target
the-media                   1287
democrats                    647
hillary-clinton              625
trump-russia                 441
joe-biden                    402
                            …
mccabe-memos                   1
state-department               1
us-mexico-trade-surplus        1
us-court-system                1
mike-pence                     1
Name: count, Length: 866, dtype: int64
```

[4]:
```
# create binary variable
trump["media_target"] = [1 if x == "the-media" else 0 for x in trump["target"]]

# transfrom string to datetime type
trump["date"] = pd.to_datetime(trump["date"])
print(type[trump["date"][0]])

trump["date_from_0"] = (trump["date"] - trump["date"].min()).dt.days

trump.head()
```

```
type[Timestamp('2014-10-09 00:00:00')]
```

[4]:

	Unnamed: 0	date	target	insult	tweet	media_target	date_from_0
0	1	2014-10-09	thomas-frieden	fool	Can you believe this fool, Dr. Thomas Frieden …	0	0
1	2	2014-10-09	thomas-frieden	DOPE	Can you believe this fool, Dr. Thomas Frieden …	0	0
2	3	2015-06-16	politicians	all talk and no action	Big time in U.S. today - MAKE AMERICA GREAT AG…	0	250
3	4	2015-06-24	ben-cardin	It's politicians like Cardin that have destroy…	Politician @SenatorCardin didn't like that I s…	0	258
4	5	2015-06-24	neil-young	total hypocrite	For the nonbeliever, here is a photo of @Neily…	0	258

```
   date_from_0
0            0
1            0
2          250
3          258
4          258
```

[10]:
```
# create feature and target
x = trump['date_from_0'].values.reshape(-1, 1)
y = trump['media_target'].values

# fit linear regression model
mod = LinearRegression()
mod.fit(x, y)

# output model
print(f"Intercept: {mod.intercept_}")
print(f"Coefficient: {mod.coef_[0]}")
print(f"R-squared: {mod.score(x, y)}")
```

```
Intercept: 0.062765594926291
Coefficient: 4.3098811591001204e-05
R-squared: 0.006042202102098382
```

Since the coefficient is sligthly positive, Trump becomes more likely to insult the media over time. Maybe he's getting more negative media attention over time and therefore insults the media more often. Another possible explanation is that him insulting the media gets more positive resonance from his followers.

4. Using the CountVectorizer from sklearn, convert the tweets to a document-feature matrix. What are the dimensions of the matrix?

[15]:
```
# creating and fitting a vectorizer
vectorizer = CountVectorizer(lowercase=False, ngram_range=(1,1),␣
 ↪analyzer="word")
matrix = vectorizer.fit_transform(trump['tweet'])

matrix.shape
```

[15]:
```
(10360, 12902)
```

1.1.2 2: Preprocessing steps

1. Remove all tagged users, i.e. words starting with the '@' character.
2. Lowercase all tweet text.
3. Remove numbers.
4. Remove extra whitespace.
5. Remove default stopwords.
6. Remove punctuation.

7. Stem words.
8. Lemmatize words. (Hint: lemmatization requires part-of-speech tags)

A couple of hints: The NLTK library has a stemmer and a lemmatizer, and other helpful lexical resources.

The text (string objects) in a dataframe column can be accessed using .str (eg. df.text_col.str) see here https://pandas.pydata.org/docs/user_guide/text.html#string-methods

For some steps it is a good idea to define a function that works on single string and then use the *apply* method from pandas

[25]:
```
# 1. Remove all tagged users (words starting with @)
trump['tweet_no_tags'] = trump['tweet'].str.replace(r'@\w+', '', regex=True)

# 2. lowercase all text
trump['tweet_lowercase'] = trump['tweet_no_tags'].str.lower()

# 3. remove numbers
trump['tweet_no_numbers'] = trump['tweet_lowercase'].str.replace(r'\d+', '',␣
 ↪regex=True)

# 4. remove extra whitespace
trump['tweet_no_whitespace'] = trump['tweet_no_numbers'].str.replace(r'\s+', '␣
 ↪', regex=True)
trump['tweet_no_whitespace'] = trump['tweet_no_whitespace'].str.strip()

trump.head()
```

[25]:

	Unnamed: 0	date	target	insult	tweet	media_target
0	1	2014-10-09	thomas-frieden	fool	Can you believe this fool, Dr. Thomas Frieden …	0
1	2	2014-10-09	thomas-frieden	DOPE	Can you believe this fool, Dr. Thomas Frieden …	0
2	3	2015-06-16	politicians	all talk and no action	Big time in U.S. today - MAKE AMERICA GREAT AG…	0
3	4	2015-06-24	ben-cardin	It's politicians like Cardin that have destroy…	Politician @SenatorCardin didn't like that I s…	0
4	5	2015-06-24	neil-young	total hypocrite	For the nonbeliever, here is a photo of @Neily…	0

```
   date_from_0                                        tweet_no_tags  \
0            0  Can you believe this fool, Dr. Thomas Frieden …
1            0  Can you believe this fool, Dr. Thomas Frieden …
2          250  Big time in U.S. today - MAKE AMERICA GREAT AG…
3          258  Politician  didn't like that I said Baltimore …
4          258  For the nonbeliever, here is a photo of  in my…

                                    tweet_lowercase  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician  didn't like that i said baltimore …
4  for the nonbeliever, here is a photo of  in my…

                                    tweet_no_numbers  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician  didn't like that i said baltimore …
4  for the nonbeliever, here is a photo of  in my…

                                   tweet_no_whitespace
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician didn't like that i said baltimore n…
4  for the nonbeliever, here is a photo of in my …
```

[31]:
```
# 5 remove default stopwords
print(stopwords.words('english'))

# define function to remove stopwords
def remove_stopwords(text):

    patterns = set(stopwords.words('english'))  # set removes duplicates and is␣
 ↪more time-efficient

    for pattern in patterns:
        if re.search(r'\b'+pattern+r'\b', text):    # search for exact stopword␣
 ↪match in text
            text = re.sub(r'\b'+pattern+r'\b', '', text)    # substitute␣
 ↪stopword with empty string
    text = re.sub(r'\s+',' ', text)    # remove extra whitespace from removing␣
 ↪stop words
    text = text.strip()    # remove whitespace at beginning or end
```

```
    return text

trump['tweet_no_stopwords'] = trump['tweet_no_whitespace'].apply(lambda x:␣
 ↪remove_stopwords(x))

trump.head()
```

```
['a', 'about', 'above', 'after', 'again', 'against', 'ain', 'all', 'am', 'an',
'and', 'any', 'are', 'aren', "aren't", 'as', 'at', 'be', 'because', 'been',
'before', 'being', 'below', 'between', 'both', 'but', 'by', 'can', 'couldn',
"couldn't", 'd', 'did', 'didn', "didn't", 'do', 'does', 'doesn', "doesn't",
'doing', 'don', "don't", 'down', 'during', 'each', 'few', 'for', 'from',
'further', 'had', 'hadn', "hadn't", 'has', 'hasn', "hasn't", 'have', 'haven',
"haven't", 'having', 'he', "he'd", "he'll", 'her', 'here', 'hers', 'herself',
"he's", 'him', 'himself', 'his', 'how', 'i', "i'd", 'if', "i'll", "i'm", 'in',
'into', 'is', 'isn', "isn't", 'it', "it'd", "it'll", "it's", 'its', 'itself',
"i've", 'just', 'll', 'm', 'ma', 'me', 'mightn', "mightn't", 'more', 'most',
'mustn', "mustn't", 'my', 'myself', 'needn', "needn't", 'no', 'nor', 'not',
'now', 'o', 'of', 'off', 'on', 'once', 'only', 'or', 'other', 'our', 'ours',
'ourselves', 'out', 'over', 'own', 're', 's', 'same', 'shan', "shan't", 'she',
"she'd", "she'll", "she's", 'should', 'shouldn', "shouldn't", "should've", 'so',
'some', 'such', 't', 'than', 'that', "that'll", 'the', 'their', 'theirs',
'them', 'themselves', 'then', 'there', 'these', 'they', "they'd", "they'll",
"they're", "they've", 'this', 'those', 'through', 'to', 'too', 'under', 'until',
'up', 've', 'very', 'was', 'wasn', "wasn't", 'we', "we'd", "we'll", "we're",
'were', 'weren', "weren't", "we've", 'what', 'when', 'where', 'which', 'while',
'who', 'whom', 'why', 'will', 'with', won', "won't", 'wouldn', "wouldn't", 'y',
'you', "you'd", "you'll", 'your', "you're", 'yours', 'yourself', 'yourselves',
"you've"]
```

[31]:

	Unnamed: 0	date	target	insult	tweet	media_target
0	1	2014-10-09	thomas-frieden	fool	Can you believe this fool, Dr. Thomas Frieden …	0
1	2	2014-10-09	thomas-frieden	DOPE	Can you believe this fool, Dr. Thomas Frieden …	0
2	3	2015-06-16	politicians	all talk and no action		
3	4	2015-06-24	ben-cardin	It's politicians like Cardin that have destroy…		
4	5	2015-06-24	neil-young	total hypocrite		

```
2  Big time in U.S. today - MAKE AMERICA GREAT AG…              0
3  Politician @SenatorCardin didn't like that I s…              0
4  For the nonbeliever, here is a photo of @Neily…              0

   date_from_0                                        tweet_no_tags  \
0            0  Can you believe this fool, Dr. Thomas Frieden …
1            0  Can you believe this fool, Dr. Thomas Frieden …
2          250  Big time in U.S. today - MAKE AMERICA GREAT AG…
3          258  Politician  didn't like that I said Baltimore …
4          258  For the nonbeliever, here is a photo of  in my…

                                      tweet_lowercase  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician  didn't like that i said baltimore …
4  for the nonbeliever, here is a photo of  in my…

                                       tweet_no_numbers  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician  didn't like that i said baltimore …
4  for the nonbeliever, here is a photo of  in my…

                                    tweet_no_whitespace  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician didn't like that i said baltimore n…
4  for the nonbeliever, here is a photo of in my …

                                     tweet_no_stopwords
0  believe fool, dr. thomas frieden cdc, stated, …
1  believe fool, dr. thomas frieden cdc, stated, …
2  big time u.. today - make america great ! poli…
3  politician ' like said baltimore needs jobs & …
4  nonbeliever, photo office $$ request-total hyp…
```

[32]:
```
# 6 remove punctuation
trump['tweet_no_punctuation'] = trump['tweet_no_stopwords'].str.translate(str.
 ↪maketrans('', '', string.punctuation))
```

[36]:
```
# 7 stem words
def stemmer(text, stemmer = PorterStemmer()):
```

```
    text = word_tokenize(text)        #Tokenizing, as stemmer only takes␣
 ↪tokenized sentences
    text_stemmed = [stemmer.stem(word) for word in text]       #Stemming each␣
 ↪word in the sentence with list comprehension
    return ' '.join(text_stemmed)    #Joining the stemmed words back into a␣
 ↪sentence

trump['tweet_stemmed'] = trump['tweet_no_punctuation'].apply(lambda x:␣
 ↪stemmer(x))
```

[42]:
```
# 8 lemmatize words
def lemmatize(text, lemmatizer = WordNetLemmatizer()):

    tag_map = defaultdict(lambda : wordnet.NOUN)
    tag_map['J'] = wordnet.ADJ
    tag_map['V'] = wordnet.VERB
    tag_map['R'] = wordnet.ADV

    text = word_tokenize(text)              #Tokenizing, as lemmatizer only␣
 ↪takes tokenized sentences
    text_lemmatized = [lemmatizer.lemmatize(word, tag_map[tag[0]]) for word,␣
 ↪tag in pos_tag(text)]

    return ' '.join(text_lemmatized)

trump['tweet_lemmatized'] = trump['tweet_no_punctuation'].apply(lambda x:␣
 ↪lemmatize(x))
```

[43]:
```
trump.head()
```

[43]:

	Unnamed: 0	date	target	insult	tweet	media_target
0	1	2014-10-09	thomas-frieden	fool	Can you believe this fool, Dr. Thomas Frieden …	0
1	2	2014-10-09	thomas-frieden	DOPE	Can you believe this fool, Dr. Thomas Frieden …	0
2	3	2015-06-16	politicians	all talk and no action		
3	4	2015-06-24	ben-cardin	It's politicians like Cardin that have destroy…		
4	5	2015-06-24	neil-young	total hypocrite		

```
2  Big time in U.S. today - MAKE AMERICA GREAT AG…             0
3  Politician @SenatorCardin didn't like that I s…             0
4  For the nonbeliever, here is a photo of @Neily…             0

   date_from_0                                        tweet_no_tags  \
0            0  Can you believe this fool, Dr. Thomas Frieden …
1            0  Can you believe this fool, Dr. Thomas Frieden …
2          250  Big time in U.S. today - MAKE AMERICA GREAT AG…
3          258  Politician  didn't like that I said Baltimore …
4          258  For the nonbeliever, here is a photo of  in my…

                                      tweet_lowercase  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician  didn't like that i said baltimore …
4  for the nonbeliever, here is a photo of  in my…

                                       tweet_no_numbers  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician  didn't like that i said baltimore …
4  for the nonbeliever, here is a photo of  in my…

                                    tweet_no_whitespace  \
0  can you believe this fool, dr. thomas frieden …
1  can you believe this fool, dr. thomas frieden …
2  big time in u.s. today - make america great ag…
3  politician didn't like that i said baltimore n…
4  for the nonbeliever, here is a photo of in my …

                                     tweet_no_stopwords  \
0  believe fool, dr. thomas frieden cdc, stated, …
1  believe fool, dr. thomas frieden cdc, stated, …
2  big time u.. today - make america great ! poli…
3  politician ' like said baltimore needs jobs & …
4  nonbeliever, photo office $$ request-total hyp…

                                    tweet_no_punctuation  \
0  believe fool dr thomas frieden cdc stated anyo…
1  believe fool dr thomas frieden cdc stated anyo…
2  big time u today  make america great  politici…
3  politician  like said baltimore needs jobs  sp…
4  nonbeliever photo office  request-total hypocr…

                                         tweet_stemmed  \
```

```
0  believ fool dr thoma frieden cdc state anyon f…
1  believ fool dr thoma frieden cdc state anyon f…
2  big time u today make america great politician…
3  politician like said baltimor need job spirit …
4  nonbeliev photo offic request-tot hypocrit htt…
```

```
                                    tweet_lemmatized
0  believe fool dr thomas frieden cdc state anyon…
1  believe fool dr thomas frieden cdc state anyon…
2  big time u today make america great politician…
3  politician like say baltimore need job spirit …
4  nonbeliever photo office request-total hypocri…
```

1.1.3 3: Consequences of preprocessing

Create a new document-feature matrix with the preprocessed tweets. How do the dimensions of this matrix compare with those of the matrix you created in 1.3?

[44]:
```
matrix_stemmed = vectorizer.fit_transform(trump['tweet_stemmed'])
matrix_stemmed.shape
```

[44]:
```
(10360, 7004)
```

[45]:
```
matrix_lemmatized = vectorizer.fit_transform(trump['tweet_lemmatized'])
matrix_lemmatized.shape
```

[45]:
```
(10360, 7989)
```