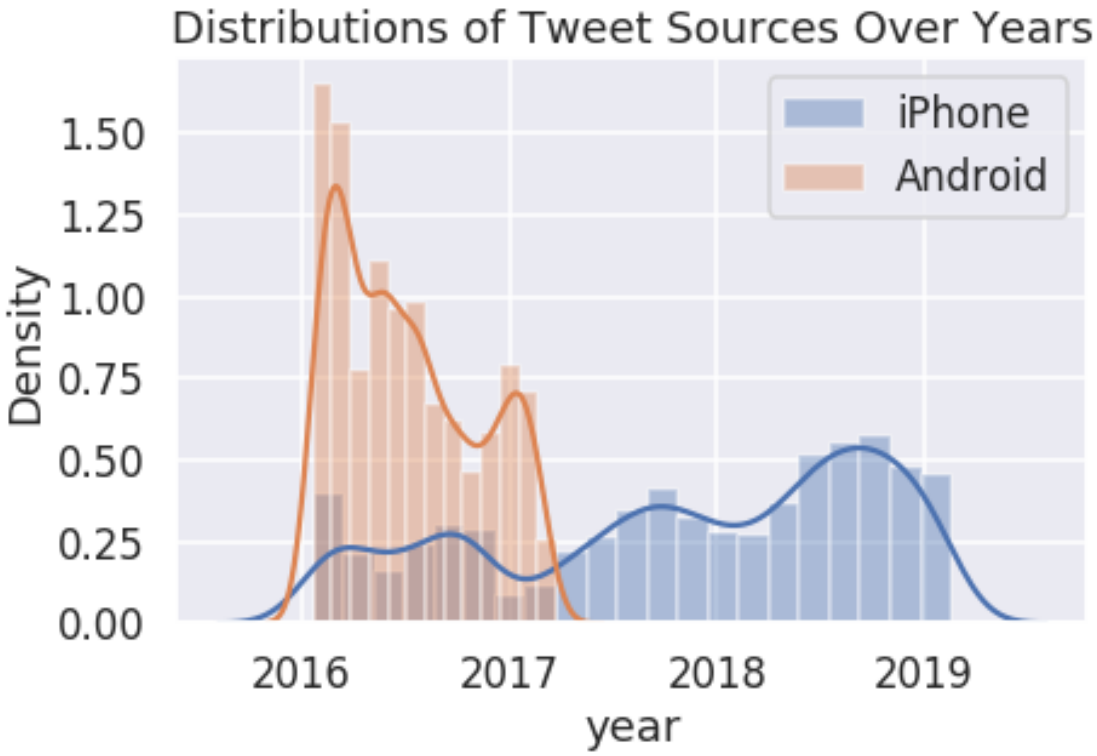
0.1 Question 0

There are many ways we could choose to read the President's tweets. Why might someone be interested in doing data analysis on the President's tweets? Name a kind of person or institution which might be interested in this kind of analysis. Then, give two reasons why a data analysis of the President's tweets might be interesting or useful for them. Answer in 2-3 sentences.

News media might be interested. They were able to analyze his reaction in his tweets in order to judge his next step.

Now, use `sns.distplot` to overlay the distributions of Trump's 2 most frequently used web technologies over the years. Your final plot should look similar to the plot below:

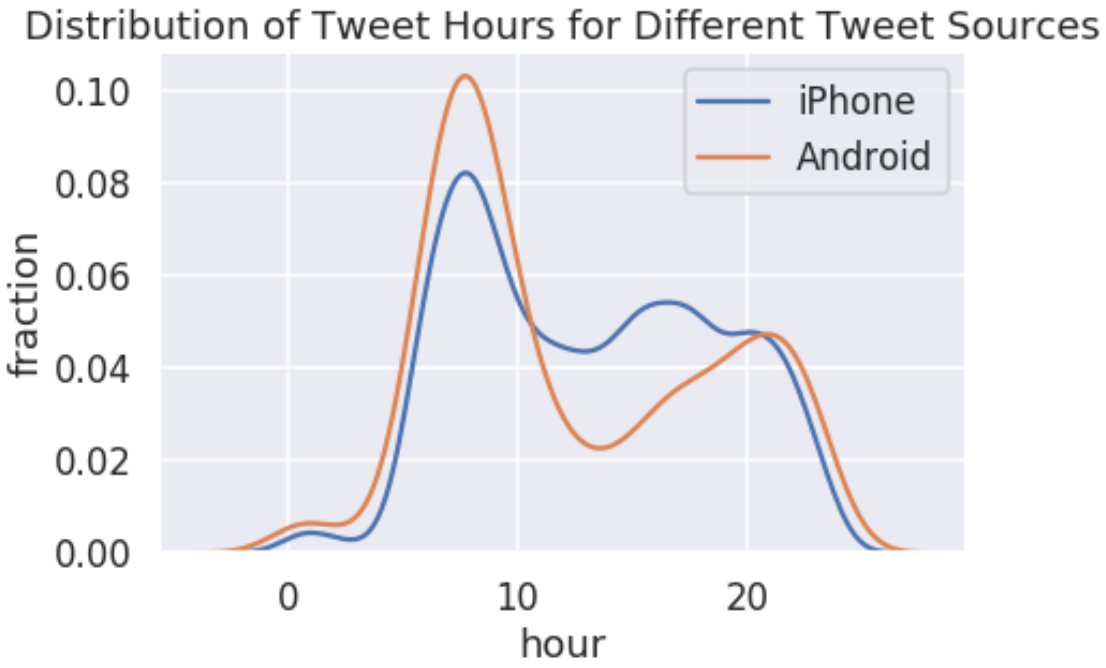
```
In [14]: sns.distplot(trump[trump['source'] == 'Twitter for iPhone']['year'], label= 'iPhone')
         sns.distplot(trump[trump['source'] == 'Twitter for Android']['year'], label= 'Android')
         plt.legend()
         plt.title('Distributions of Tweet Sources Over Years')
         plt.show()
```



0.1.1 Question 4b

Use this data along with the seaborn `distplot` function to examine the distribution over hours of the day in eastern time that Trump tweets on each device for the 2 most commonly used devices. Your final plot should look similar to the following:

```
In [19]: ### make your plot here
         sns.distplot(trump[trump['source'] == 'Twitter for iPhone']['hour'], label= 'iPhone', hist = F
         sns.distplot(trump[trump['source'] == 'Twitter for Android']['hour'], label= 'Android', hist =
         plt.legend()
         plt.ylabel('fraction')
         plt.title('Distribution of Tweet Hours for Different Tweet Sources')
         plt.show()
```

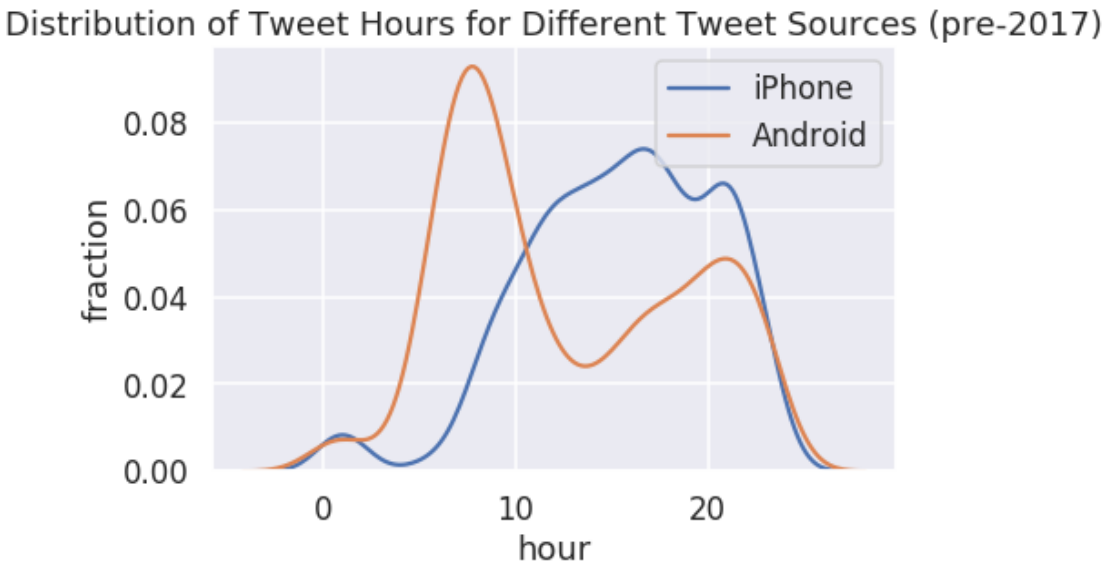


0.1.2 Question 4c

According to this Verge article, Donald Trump switched from an Android to an iPhone sometime in March 2017.

Let's see if this information significantly changes our plot. Create a figure similar to your figure from question 4b, but this time, only use tweets that were tweeted before 2017. Your plot should look similar to the following:

```
In [20]: ### make your plot here
         sns.distplot(trump[(trump['source'] == 'Twitter for iPhone') & (trump['year'] < 2017)]['hour']
         sns.distplot(trump[(trump['source'] == 'Twitter for Android') & (trump['year'] < 2017)]['hour']
         plt.legend()
         plt.ylabel('fraction')
         plt.title('Distribution of Tweet Hours for Different Tweet Sources (pre-2017)')
         plt.show()
```



0.1.3 Question 4d

During the campaign, it was theorized that Donald Trump's tweets from Android devices were written by him personally, and the tweets from iPhones were from his staff. Does your figure give support to this theory? What kinds of additional analysis could help support or reject this claim?

Yes, because we can learn from 4b graph that Trump usually tweeting in morning around 8~10 and the Android line from 4c graph meets the above, but the peak of the iphone line is obviously in the afternoon or evening, which does not match the 4c graphics. This is likely to be done by his team. Working hours of Trump staff can be used as additional analysis.

0.2 Question 5

The creators of VADER describe the tool's assessment of polarity, or "compound score," in the following way:

"The compound score is computed by summing the valence scores of each word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). This is the most useful metric if you want a single unidimensional measure of sentiment for a given sentence. Calling it a 'normalized, weighted composite score' is accurate."

As you can see, VADER doesn't "read" sentences, but works by parsing sentences into words assigning a preset generalized score from their testing sets to each word separately.

VADER relies on humans to stabilize its scoring. The creators use Amazon Mechanical Turk, a crowdsourcing survey platform, to train its model. Its training set of data consists of a small corpus of tweets, New York Times editorials and news articles, Rotten Tomatoes reviews, and Amazon product reviews, tokenized using the natural language toolkit (NLTK). Each word in each dataset was reviewed and rated by at least 20 trained individuals who had signed up to work on these tasks through Mechanical Turk.

0.2.1 Question 5a

Please score the sentiment of one of the following words: - police - order - Democrat - Republican - gun - dog - technology - TikTok - security - face-mask - science - climate change - vaccine

What score did you give it and why? Can you think of a situation in which this word would carry the opposite sentiment to the one you've just assigned?

I will give security 0.8 point because security can ensure that our property and life are not in danger, is a positive word. The opposite of this word is insecurity.

0.2.2 Question 5b

VADER aggregates the sentiment of words in order to determine the overall sentiment of a sentence, and further aggregates sentences to assign just one aggregated score to a whole tweet or collection of tweets. This is a complex process and if you'd like to learn more about how VADER aggregates sentiment, here is the info at this link.

Are there circumstances (e.g. certain kinds of language or data) when you might not want to use VADER? What features of human speech might VADER misrepresent or fail to capture?

Yes, it would be difficult to use VADER when the text is in the sarcastic or the words have multiple different meanings.

0.3 Question 5h

Read the 5 most positive and 5 most negative tweets. Do you think these tweets are accurately represented by their polarity scores?

Yes, I can see lots of dark or unpleasant words in the 5 most negative tweets and lots of pleasant or inspiring words in the 5 most positive tweets.

0.4 Question 6

Now, let's try looking at the distributions of sentiments for tweets containing certain keywords.

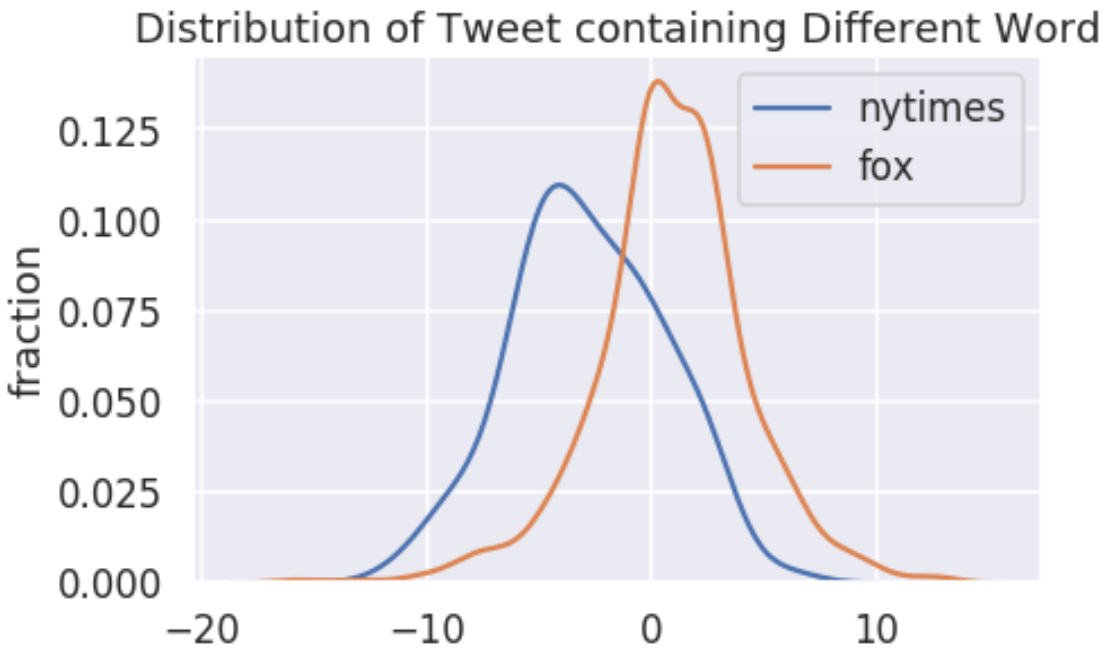
0.4.1 Question 6a

In the cell below, create a single plot showing both the distribution of tweet sentiments for tweets containing `nytimes`, as well as the distribution of tweet sentiments for tweets containing `fox`.

Be sure to label your axes and provide a title and legend. Be sure to use different colors for `fox` and `nytimes`.

```
In [34]: sns.distplot(trump[trump['text'].str.contains("nytimes")][['polarity']], label = 'nytimes', his
         sns.distplot(trump[trump['text'].str.contains("fox")][['polarity']], label = 'fox', hist = Fals
         plt.legend()
         plt.ylabel('fraction')
         plt.title('Distribution of Tweet containing Different Word')
```

```
Out[34]: Text(0.5, 1.0, 'Distribution of Tweet containing Different Word')
```



0.4.2 Question 6b

Comment on what you observe in the plot above. Can you find another pair of keywords that lead to interesting plots? Describe what makes the plots interesting. (If you modify your code in 6a, remember to change the words back to `nytimes` and `fox` before submitting for grading).

Tweet containing fox seem to have higher polarity than nytimes. It can be seen from his tweets to these two media that he likes fox more than nytime.

What do you notice about the distributions? Answer in 1-2 sentences.

I noticed that hashtag or link line has higher density than none line and both peak are located at 0. This means the tweet has hashtag or link has higher neutrality.