

Desafio do Módulo 3

Entrega 27 jun em 23:59
Limite de tempo Nenhum

Pontos 40

Perguntas 14

Disponível até 27 jun em 23:59

Instruções

O Desafio do Módulo 3 está disponível!

1. Instruções para realizar o desafio

Consulte a data de entrega no teste em seu calendário.

Reserve um tempo para realizar a atividade, leia as orientações e enunciados com atenção. Em caso de dúvidas utilize o "Fórum de dúvidas do Desafio".

Para iniciá-lo clique em "Fazer teste". Você tem somente **uma** tentativa e não há limite de tempo definido para realizá-lo. Caso precise interromper a atividade, apenas deixe a página e, ao retornar, clique em "Retomar teste".

Clique em "Enviar teste" **somente** quando você concluí-lo. Antes de enviar confira todas as questões.

O gabarito será disponibilizado partir de sábado, **27/06/2020**, às 23h59.

Bons estudos!

2. O arquivo abaixo contém o enunciado do desafio

[Enunciado do Desafio do Módulo 3 - Bootcamp Analista de Dados.pdf](#) 

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	Tentativa 1	2.667 minutos	40 de 40

⚠ As respostas corretas estarão disponíveis em 27 jun em 23:59.

Pontuação deste teste: **40** de 40

Enviado 26 jun em 13:52

Esta tentativa levou 2.667 minutos.


Pergunta 1

2,85 / 2,85 pts

A função **mean** do pacote **numpy** foi utilizada no desafio para apoiar a análise de sentimento realizada. Considere o código abaixo escrito em Python. Qual o valor que a função **mean** do pacote **numpy** irá retornar?

```
1 import numpy
2 import random
3
4 valores=[] # Lista vazia para armazenar valores
5
6 i=0
7 while i < 10:
8     valores.append(random.randrange(0, 10)) #Inclui na lista um valor
9     i=i+1                                     #randomico entre 0 e 10
10
11 print(valores) #imprime a lista de valores
12
```

[8, 3, 4, 8, 3, 4, 3, 2, 0, 8]

1 numpy.mean(valores) 

☐ 3.7

☐ 3.5

☒ 4.3

☐ 2.79

Pergunta 2**2,85 / 2,85 pts**

Em Python, a função ***sentiment.polarity*** do pacote *TextBlob* foi usada para calcular a polaridade do tweets. A função retorna um valor entre -1 e 1 sobre um determinado texto. Sobre a polaridade calculada por esta função, podemos afirmar que:



Quando uma polaridade for maior que 0(zero), indica que o texto tem uma tendência indicar um sentimento positivo.



Uma polaridade for maior que 0(zero), indica que o texto tem uma tendência indicar um sentimento negativo.



As polaridades -1 e 1 podem ser consideradas polaridades neutras.



Uma polaridade for menor que 0(zero), indica que o texto tem uma tendência indicar um sentimento positivo.

Pergunta 3**2,85 / 2,85 pts**

Qual das bibliotecas abaixo pode ser usada no Python para acessar a API do Twitter?



rtweet.



textblob.



tweepy.



numpy.

Pergunta 4**2,85 / 2,85 pts**

Considere a função ***search*** do pacote *tweepy* para coletar tweets que atendam a condição de possui uma determinada palavra chave conforme exemplo abaixo. Qual das assertivas abaixo representam um valor inválido que parâmetro `result_type` da função pode assumir?

`search(q=keyword, result_type=?)`



mixed



popular



trend



recent

Pergunta 5**2,85 / 2,85 pts**

Considere a função **search** do pacote *tweepy* para coletar tweets que atendam a condição de possui uma determinada palavra chave conforme exemplo abaixo. Qual das assertivas abaixo está CORRETA?



O parâmetro `tweet_mode='extended'` indica que deve ser recuperado o texto total (280 caracteres) do tweet e não o texto truncado (140 caracteres).



Para recuperar os trend tweets deve-se usar o parâmetro `tweet_mode='extended'` na função `search`.



Quando usado o parâmetro `tweet_mode='extended'` será recuperado, além dos tweets, as respostas ou comentários a cada um dos tweets.



O parâmetro `tweet_mode='extended'` indica que deve ser recuperado o texto truncado (140 caracteres) do tweet.

Pergunta 6

2,85 / 2,85 pts

Sobre a API do Twitter para coletar de dados, qual das assertivas abaixo está INCORRETA?



Após receber o acesso de desenvolvedor é necessário criar uma aplicação para receber as chaves de acesso.



Para solicitar uma conta de desenvolvedor é necessário ser um usuário cadastrado, ou seja, ter uma conta no Twitter.



Não é necessário ter uma conta no Twitter, basta cadastrar um e-mail no Twitter e solicitar uma conta de desenvolvedor.



São necessárias 4 chaves de acesso para usar a API, são elas: API key, API secret key, Access token e Access token secret.

Pergunta 7

2,85 / 2,85 pts

Considere o trecho de código abaixo no Python referente à análise de sentimento dos tweets. Tal exemplo foi apresentado na videoaula. Marque a alternativa INCORRETA.

```
1 polarities = []
2
3 for tweet in tweets:
4     analysis = tb(tweet.text)
5
6     if analysis.detect_language() != 'en':
7
8         traducao = tb(str(analysis.translate(to='en')))
9
10        polarity = traducao.sentiment.polarity
11    else:
12        polarity = analysis.sentiment.polarity
13
14    polarities.append(polarity)
```



A linha 4 realiza a análise de polaridade do texto do tweet e passa o valor encontrado para a variável analysis.



A linha 14 refere-se a uma estrutura que insere valores em uma lista existente.



Na linha 10, a variável polarity recebe o valor da polaridade do texto do tweet já traduzido, no qual é obtido pela função sentimenty.polarity.



A linha 6 representa uma estrutura condicional e análise o idioma no qual o tweet foi escrito.

Pergunta 8

2,85 / 2,85 pts

Considere o trecho de código abaixo e marque a alternativa CORRETA.

```
1  #Definição do dataframe com informaçãoe básicas dos tweets
2  tweets_df = pd.DataFrame(tweets, columns=['Tweets'])
3
4  tweets_df['len'] = np.array([len(tweet) for tweet in tweets])
5  tweets_df['ID'] = np.array([tweet.id for tweet in info])
6  tweets_df['USER'] = np.array([tweet.user.screen_name for tweet in info])
7  tweets_df['userName'] = np.array([tweet.user.name for tweet in info])
8  tweets_df['User Location'] = np.array([tweet.user.location for tweet in info])
9  tweets_df['Language'] = np.array([tweet.user.lang for tweet in info])
10 tweets_df['Date'] = np.array([tweet.created_at for tweet in info])
11 tweets_df['Source'] = np.array([tweet.source for tweet in info])
12 tweets_df['Likes'] = np.array([tweet.favorite_count for tweet in info])
13 tweets_df['Retweets'] = np.array([tweet.retweet_count for tweet in info])
14 tweets_df['Geo'] = np.array([tweet.geo for tweet in info])
15 tweets_df['Coordinates'] = np.array([tweet.coordinates for tweet in info])
16 tweets_df['Place'] = np.array([tweet.place for tweet in info])
17
18 likes_max = np.max(tweets_df['Likes'])
19 retweet_max = np.max(tweets_df['Retweets'])
20
```



Nas linhas 18 e 19 a função max do numpy identifica respectivamente a média de valores das colunas Likes e Retweets do dataframe.



Nas linhas 18 e 19 a função max do numpy identifica respectivamente o valor máximo das colunas Likes e Retweets do dataframe.



Nas linhas 18 e 19 a função max do numpy identifica respectivamente o valor mínimo das colunas Likes e Retweets do dataframe.



Nas linhas 18 e 19 a função max do numpy identifica respectivamente o valor total de dados existentes nas colunas Likes e Retweets do dataframe, ignorando aquelas no qual o valor é 0(zer0).

Pergunta 9

2,85 / 2,85 pts

Considere o trecho de código abaixo e marque a alternativa CORRETA:

```
In [11]: 1 sources = [] #Lista para armazenar a fontes
2 for source in tweets_df['Source']:
3     if source not in sources:
4         sources.append(source)
5
6 percent = np.zeros(len(sources))
7
8 for source in tweets_df['Source']:
9     for index in range(len(sources)):
10        if source == sources[index]:
11            percent[index] += 1
12        pass

In [12]: 1 sourceDF = pd.DataFrame({
2     'source':percent,
3     }, index=sources)
4
5 sources_sorted = sourceDF.sort_values('source',ascending=False)
6 ax = sources_sorted.source.plot(kind='barh',color='#B8860B')
7 ax.get_xaxis().set_major_formatter(plt.FuncFormatter(lambda x, loc: "{:,}".format(int(x))
```

☐

Este trecho de código pode ser usado para identificar a fonte (origem) do tweet e gerar um gráfico apontando se o tweet da respectiva fonte é positivo ou negativo.

☐

Este trecho de código pode ser usado para identificar o sentimento expresso pelo tweet.

☐

Este trecho de código pode ser usado para identificar a localização geográfica de onde o tweet foi postado.

☒

Este trecho de código pode ser usado para identificar a fonte (origem) do tweet e gerar um gráfico apontando a quantidades de tweets por fonte.

Pergunta 10

2,85 / 2,85 pts

Considere o trecho de código abaixo e marque a alternativa CORRETA:

```
1
2 for tweet in tweets_df['Tweets']:
3     analysis = tb(tweet)
4
5     polarity = analysis.sentiment.polarity
6
7     polarities.append(polarity)
8
```

☐

Este trecho de código pode ser usado para identificar a localização geográfica de onde o tweet foi postado.

☒

Este trecho de código pode ser usado para identificar o sentimento expresso pelo tweet.

☐

Este trecho de código pode ser usado para identificar a fonte (origem) do tweet e gerar um gráfico apontando a quantidades de tweets por fonte.

☐

Este trecho de código pode ser usado para identificar a fonte (origem) do tweet e gerar um gráfico apontando se o tweet da respectiva fonte é positivo ou negativo.

Pergunta 11

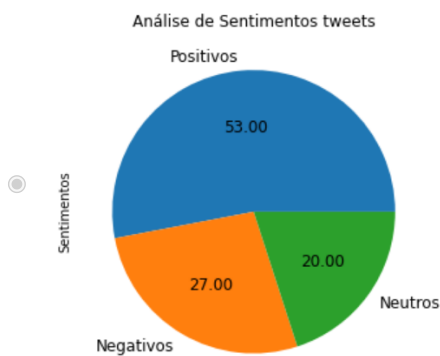
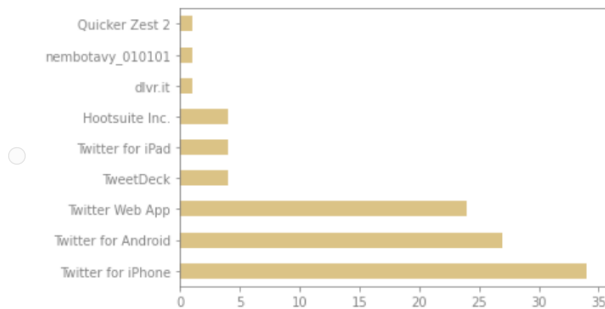
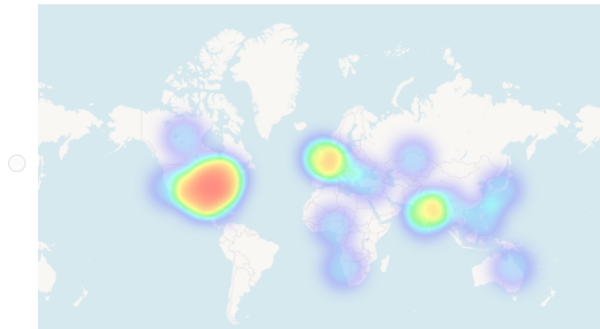
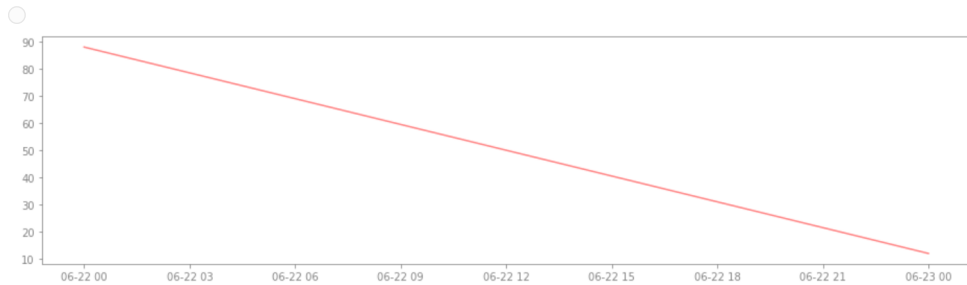
2,85 / 2,85 pts

Considere o trecho de código abaixo e marque a alternativa que indica o gráfico gerado pelo código:

```
In [32]: 1 sentiments = ['Positivos', 'Negativos', 'Neutros']
          2 percents = [pos_pct, neg_pct, neu_pct]

In [33]: 1 pie_chart = pd.Series(percents, index=sentiments, name='Sentimentos')
          2 pie_chart.plot.pie(fontsize=12, autopct='%2f', figsize=(5, 5), title="Análise de Sentimentos tweets")
```

Observação: os valores do gráfico podem ser diferentes, pois cada coletor de tweet pode coletar dados diferentes. O objetivo é identificar qual tipo de gráfico gerado.



Pergunta 12

2,85 / 2,85 pts

Considere o trecho de código abaixo e marque a alternativa que indica o gráfico gerado pelo código:

```

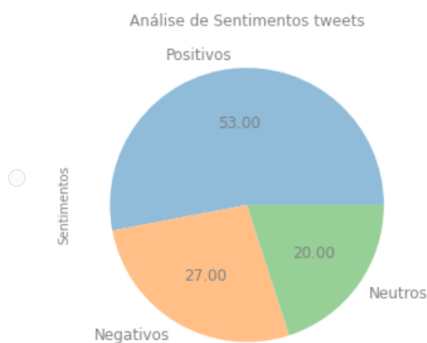
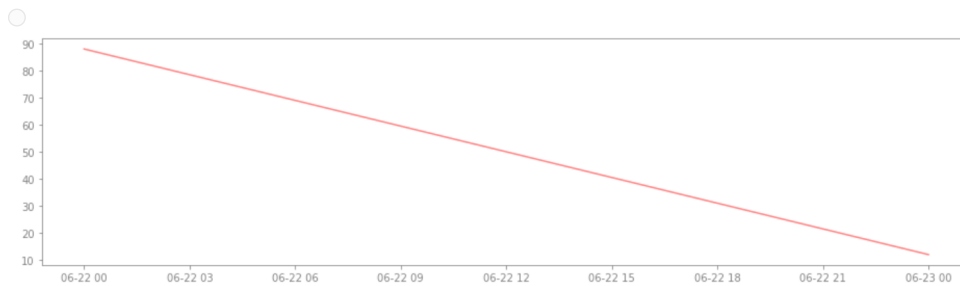
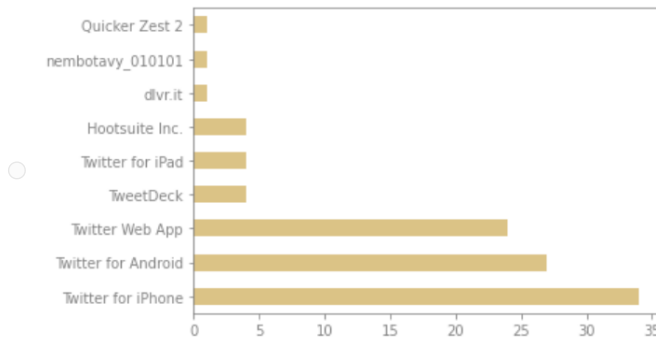
1 geolocator = Nominatim(user_agent="TweeterSentiments")
2
3 latitude = []
4 longitude = []
5
6 for user_location in tweets_df['User Location']:
7     try:
8         location = geolocator.geocode(user_location)
9         latitude.append(location.latitude)
10        longitude.append(location.longitude)
11    except:
12        continue

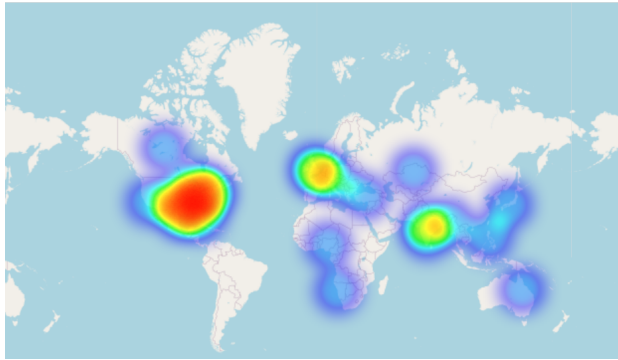
```

```

1 coordenadas = np.column_stack((latitude, longitude))
2
3 mapa = folium.Map(zoom_start=3.)
4 mapa.add_child(plugins.HeatMap(coordenadas))
5 mapa.save('covid.html')
6 mapa

```

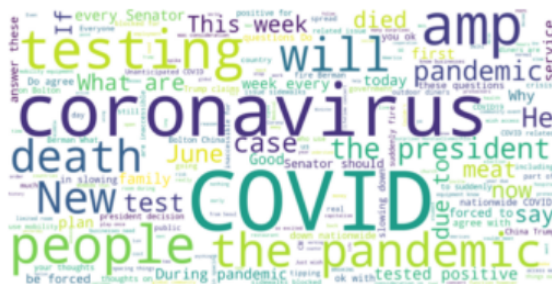


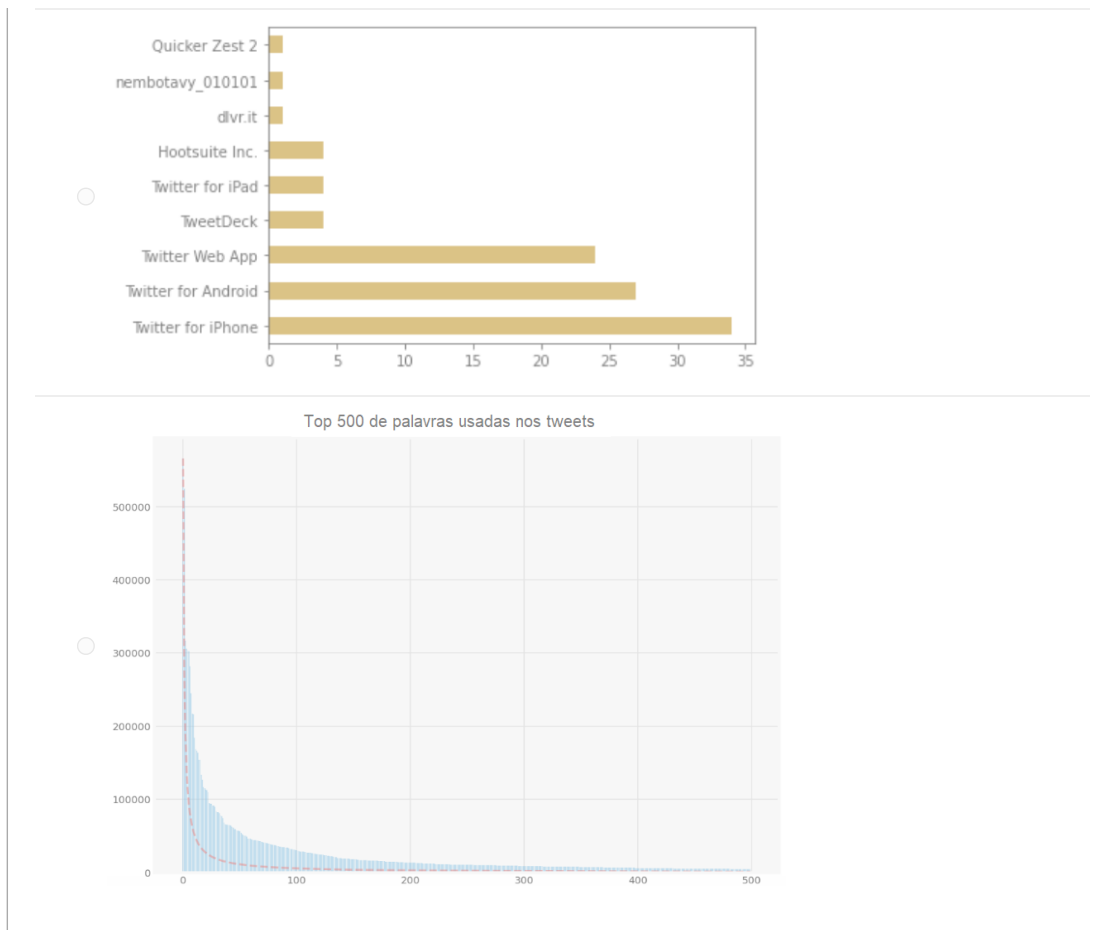


2,85 / 2,85 pts

Considere o trecho de código abaixo e marque a alternativa que melhor representa o que é gerado pelo código:

```
1 import warnings
2
3 warnings.simplefilter('ignore')
4
5 wc = WordCloud(min_font_size=10,
6                 max_font_size=300,
7                 background_color='white',
8                 mode="RGB",
9                 width=2000,
10                 height=1000,
11                 normalize_plurals= True).generate(words_clean)
12
13 plt.imshow(wc, interpolation="bilinear")
14 plt.axis("off")
15 plt.savefig('covid_ccloud.png', dpi=300)
16 plt.show()
```



**Pergunta 14****2,95 / 2,95 pts**

Para realizar o desafio utilizamos, entre outras, as seguintes bibliotecas do Python: tweepy, wordcloud, textblob, geopy.

Considere as funcionalidades utilizadas de cada uma destas bibliotecas e marque a assertiva CORRETA.

☐

Geopy - Utilizada para realizar a análise de frequência de uma palavra no tweet e gerar uma nuvem de palavras com esta frequência.

☒

Tweepy – Utilizada para fazer a interface com a API do Twitter e assim, coletar os tweets.

☐

Textblob - Usada para criar uma dataframe que armazena todos os tweets coletados.

☐

Wordcloud – Utilizada para realizar a análise de polaridade do Twitter.

Pontuação do teste: 40 de 40