

# nlptweets

June 19, 2020

## 0.0.1 Twitter Sentiment Analysis - Gov CDMX

```
[1]: import matplotlib.dates as mdates
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from tensorflow.python.lib.io import file_io
```

```
[ ]: !export GOOGLE_APPLICATION_CREDENTIALS="/Users/admin/Downloads/
→gov-cdmx-twitter-sentiment-3479e766b2c0.json"
!export PATH="/usr/local/Caskroom/google-cloud-sdk/latest/google-cloud-sdk/bin:/
→usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin:/Library/TeX/texbin:/Users/admin/
→development/gov-cdmx-twitter-sentiment/streaming/env/bin:/usr/local/Caskroom/
→google-cloud-sdk/latest/google-cloud-sdk/bin:/usr/local/sbin:/Users/admin/
→golang/bin:/usr/local/Cellar/go/1.13/libexec/bin:/Users/admin/lib:/Users/
→admin/golang/bin:/usr/local/Cellar/go/1.13/libexec/bin:/Users/admin/lib:/
→Library/TeX/texbin"
```

```
[ ]: !pip install --upgrade google-api-python-client
```

```
[ ]: %!bq tables describe --name gov-cdmx-twitter-sentiment:
→gov_cdmx_twitter_sentiment.tweets
```

Now that the data is in bigquery we can save a section to cloud storage or grab it direct from bigquery.

```
[5]: with file_io.FileIO('gs://gov-cdmx-twitter-sentiment/nlpstorage/
→bq-results-20200619-114357-99zrfxhaozw3.csv', 'r') as f:
df = pd.read_csv(f)
```

```
[6]: df.head()
```

```
[6]:   tweet_timestamp   user_text \
0      1592514243  RT @SVictoriaDiaz: Disponible desde ya mis bel...
1      1592514243  RT @CruisingTlalnel: Que rica mamada y cogida ...
2      1592514243  Estamos trabajando de manera coordinada con la...
3      1592514243  RT @AlvaradoDarlyns: #CDMX #holis @AlvaradoDar...
4      1592514243  Esta foto de @DAGUILARFOTO es preciosa! Nuestr...
```

	user_screen_name	user_location	user_followers_count
0	fernandolovaco	México	50
1	outofbody4	NaN	10
2	Victori64369605	NaN	2
3	Set_Pornstars	México	43509
4	Letyludigital	México	1054

```
[7]: print(df['user_text'][0])
      print(len(df['user_text']))
```

RT @SVictoriaDiaz: Disponible desde ya mis bellos seguidores 5543748423

#Disponible  
 #CDMX <https://t.co/Qmu9qTU0uT>  
 1300

```
[8]: rts=df[df['user_text'].str.match('RT')]
      print("Retweets ", len(rts['user_text']))
      cdmx=df[df['user_text'].str.contains('#CDMX')]
      print("#CDMX", len(cdmx['user_text']))
```

Retweets 0  
 #CDMX 285

```
[9]: from google.cloud import language
      from google.cloud.language import enums
      from google.cloud.language import types
      from google.oauth2 import service_account
      from google.protobuf.json_format import MessageToDict

      score=[]
      magnitude=[]
      creds = service_account.Credentials.from_service_account_file('/Users/admin/
      ↳Downloads/gov-cdmx-twitter-sentiment-53dba8db6dbd.json')
      client = language.LanguageServiceClient(credentials=creds)

      for tweet in cdmx['user_text']:
          document = types.Document(
              content=tweet,
              type=enums.Document.Type.PLAIN_TEXT
          )
          analyze_sentiment_response = client.analyze_sentiment(document=document)
          message = MessageToDict(analyze_sentiment_response,
          ↳including_default_value_fields=True)
          score.append(message['documentSentiment']['score'])
          magnitude.append(message['documentSentiment']['magnitude'])
```

```
print(len(score))
# print('POLARITY=%s MAGNITUDE=%s for %s' % (score, magnitude, tweet))
```

285

```
[10]: cdmx['score']=score
      cdmx['magnitude']=magnitude
      cdmx.head()
```

```
/Users/admin/development/gov-cdmx-twitter-
sentiment/streaming/env/lib/python3.7/site-packages/ipykernel_launcher.py:1:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
"""Entry point for launching an IPython kernel.
/Users/admin/development/gov-cdmx-twitter-
sentiment/streaming/env/lib/python3.7/site-packages/ipykernel_launcher.py:2:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
[10]: tweet_timestamp      user_text \
0      1592514243  RT @SVictoriaDiaz: Disponible desde ya mis bel...
3      1592514243  RT @AlvaradoDarlyns: #CDMX #holis @AlvaradoDar...
4      1592514243  Esta foto de @DAGUILARFOTO es preciosa! Nuestr...
5      1592514243  Durante una sesión de la @CONAGO_oficial, el s...
6      1592514243  #Adopción #CDMX MIMI esta Hermosura 1 añito Ca...

      user_screen_name  user_location  user_followers_count  score  magnitude
0  fernandolovaco      México          50          0.5          1.1
3   Set_Pornstars      México        43509          0.6          0.6
4   Letyludigital      México         1054          0.5          1.7
5  semujmorenacmx  Ciudad de México         466          0.1          0.3
6    Flormar07        NaN           40          0.4          1.8
```

```
[11]: cdmx['datef']=pd.to_datetime(cdmx['tweet_timestamp'], unit='s',_
    ↪yearfirst='TRUE')
      cdmx['datef'].head()
```

```
/Users/admin/development/gov-cdmx-twitter-
```

```
sentiment/streaming/env/lib/python3.7/site-packages/ipykernel_launcher.py:1:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
"""Entry point for launching an IPython kernel.

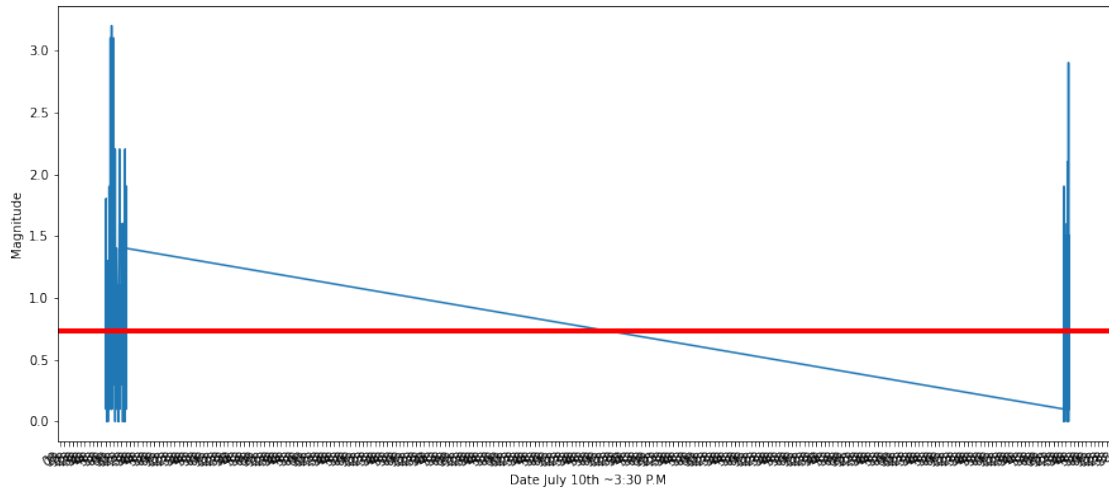
```
[11]: 0    2020-06-18 21:04:03
      3    2020-06-18 21:04:03
      4    2020-06-18 21:04:03
      5    2020-06-18 21:04:03
      6    2020-06-18 21:04:03
      Name: datef, dtype: datetime64[ns]
```

```
[12]: #averages
      score_avg=np.mean(cdmx['score'])
      magnitude_evg=np.mean(cdmx['magnitude'])
      print('score', score_avg, 'magnitude', magnitude_evg)
```

```
score 0.24105263157894743 magnitude 0.7343859649122807
```

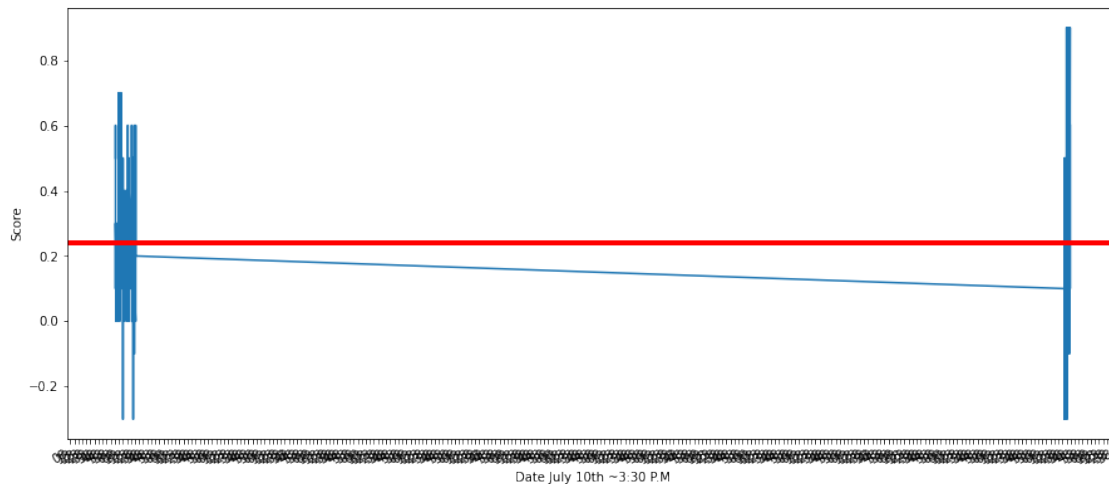
```
[13]: data = pd.concat([cdmx['datef'], cdmx['magnitude']], axis=1)
      data.set_index('datef',inplace=True)
      fig, ax = plt.subplots(figsize=(15,7))
      data.plot(ax=ax, legend=False)
      ax.axhline(y=magnitude_evg, linewidth=4, color='r')
      ax.xaxis.set_major_locator(mdates.MinuteLocator(interval=5))
      ax.xaxis.set_major_formatter(mdates.DateFormatter('%M'))
      ax.set_xlabel('Date July 10th ~3:30 P.M')
      ax.set_ylabel('Magnitude')
```

```
[13]: Text(0, 0.5, 'Magnitude')
```



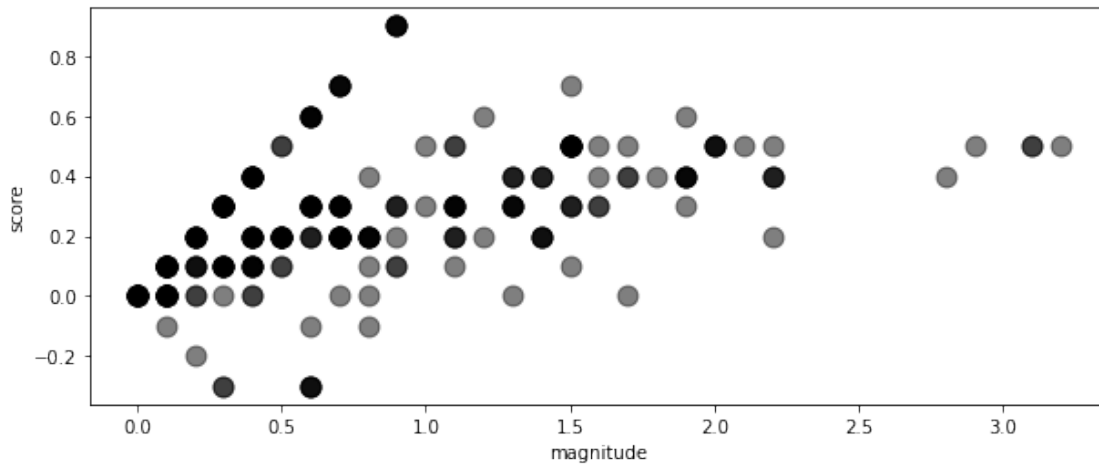
```
[14]: data2 = pd.concat([cdmx['datef'], cdmx['score']], axis=1)
data2.set_index('datef', inplace=True)
fig, ax2 = plt.subplots(figsize=(15,7))
data2.plot(ax=ax2, legend=False)
ax2.axhline(y=score_avg, linewidth=4, color='r')
ax2.xaxis.set_major_locator(mdates.MinuteLocator(interval=5))
ax2.xaxis.set_major_formatter(mdates.DateFormatter('%M'))
ax2.set_xlabel('Date July 10th ~3:30 P.M')
ax2.set_ylabel('Score')
```

```
[14]: Text(0, 0.5, 'Score')
```



High magnitude tweets are more impactful than low magnitude tweets, a weak statement doesn't say much. We will concentrate only on statements with a magnitude of 0.5 or higher.

```
[15]: fig, ax = plt.subplots()
ax.figure.set_size_inches(10,4)
ax.grid(False)
ax.scatter(cdmx.magnitude, cdmx.score, s=120, c='black', alpha=0.5)
ax.set(xlabel='magnitude', ylabel='score')
plt.show()
```



We can also see that things around zero polarity (neither very positive nor negative) are not interesting to flag

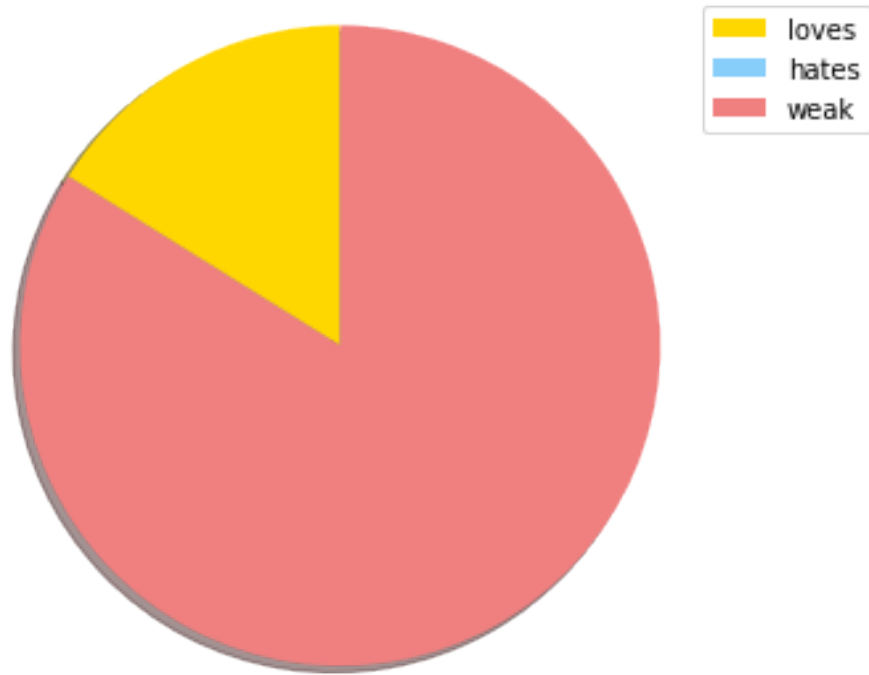
```
[16]: love=cdmx[(cdmx['magnitude'] >=0.5) & (cdmx['score'] >=0.5)]
hate=cdmx[(cdmx['magnitude'] >=0.5) & (cdmx['score'] <= -0.5)]

[17]: print(len(cdmx['magnitude']), len(love['magnitude']), len(hate['magnitude']))
print("weak", len(cdmx['magnitude']) -
      ↳(len(love['magnitude'])+len(hate['magnitude'])))
```

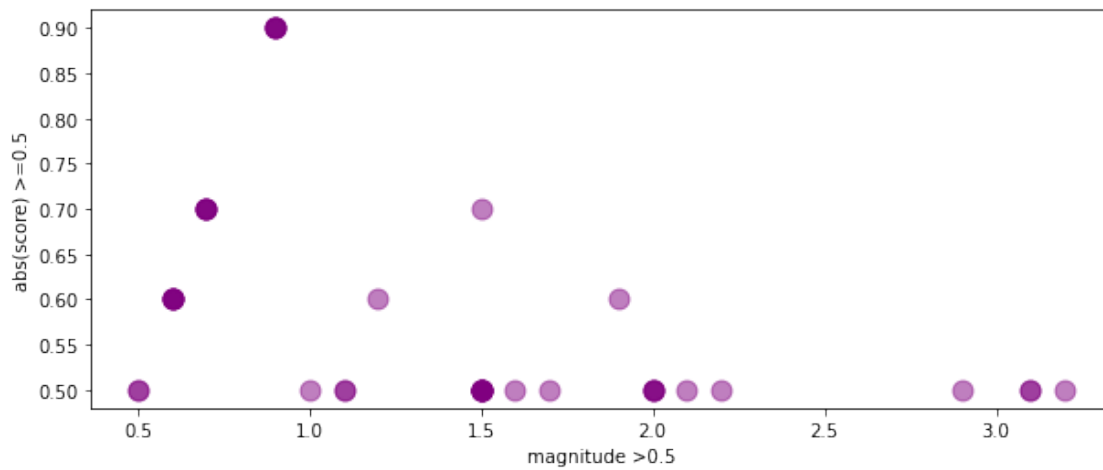
285 46 0

weak 239

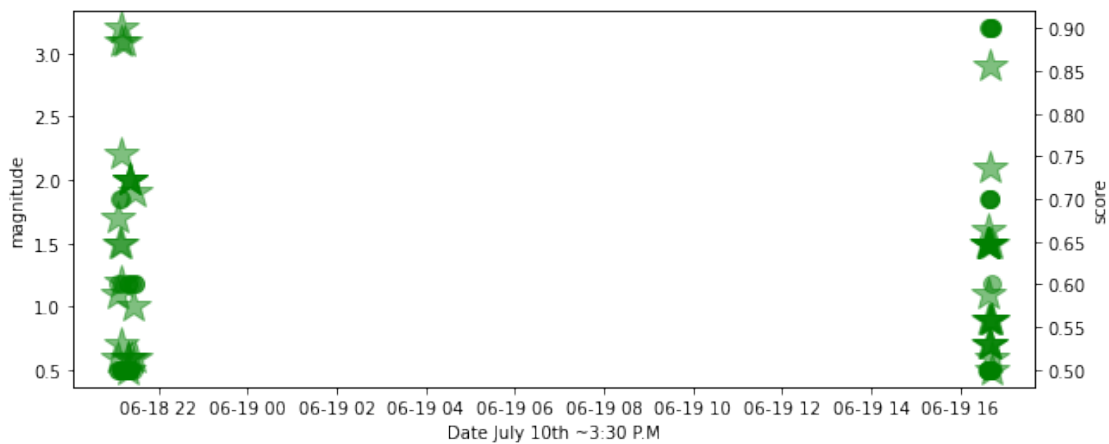
```
[19]: labels = ['loves', 'hates', 'weak']
sizes = [46, 0, 239]
colors = ['gold', 'lightskyblue', 'lightcoral']
patches, texts = plt.pie(sizes, colors=colors, shadow=True, startangle=90)
plt.legend(patches, labels, loc="best")
plt.axis('equal')
plt.tight_layout()
plt.show()
```



```
[20]: fig, ax= plt.subplots()
ax.scatter(love.magnitude, love.score, s=120, c='purple', alpha=0.5)
ax.scatter(hate.magnitude, hate.score, s=120, c='red', alpha=0.5)
ax.figure.set_size_inches(10,4)
ax.grid(False)
ax.set(xlabel='magnitude >0.5', ylabel='abs(score) >=0.5')
plt.show()
```

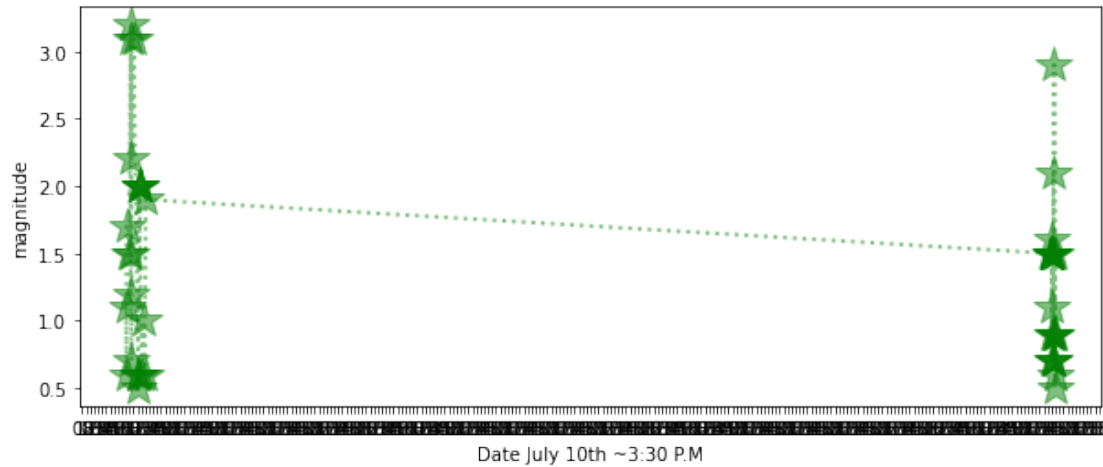


```
[21]: fig, ax1 = plt.subplots()
ax1.set_xlabel('Date July 10th ~3:30 P.M')
ax1.set_ylabel('magnitude')
ax1.grid(False)
ax1.xaxis.set_major_locator(mdates.MinuteLocator(interval=5))
ax1.xaxis.set_major_formatter(mdates.DateFormatter('%M'))
ax1.figure.set_size_inches(10,4)
ax1.plot(love['datef'], love['magnitude'], 'g*', markersize=20, alpha=0.5)
ax1.plot(hate['datef'], hate['magnitude'], 'r*', markersize=20, alpha=0.5)
ax2=ax1.twinx()
ax2.grid(False)
ax2.plot(love.datef, love.score, 'g.', markersize=20, alpha=0.5)
ax2.plot(hate.datef, hate.score, 'r.', markersize=20, alpha=0.5)
ax2.set_ylabel('score')
plt.show()
```

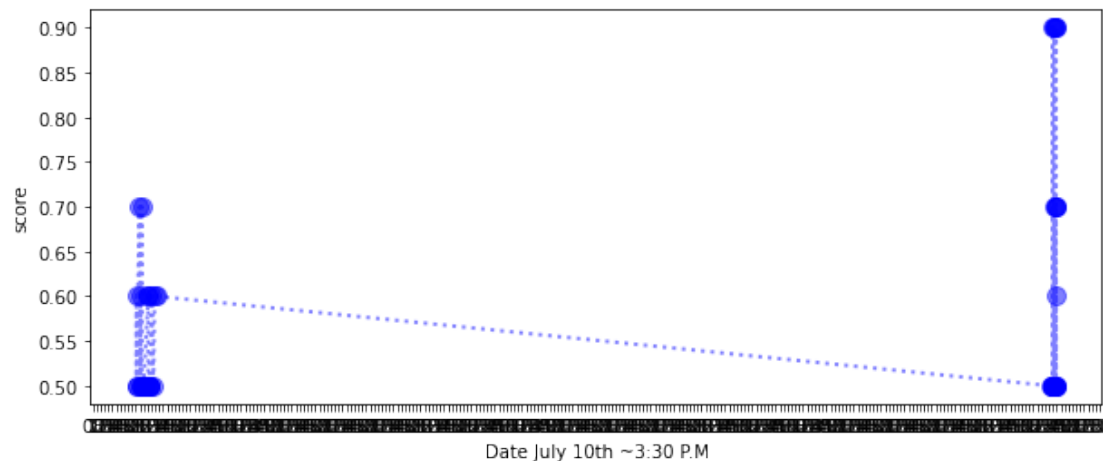


```
[22]: fig, ax = plt.subplots()
ax.set_xlabel('Date July 10th ~3:30 P.M')
ax.grid(False)
ax.xaxis.set_major_locator(mdates.MinuteLocator(interval=5))
ax.xaxis.set_major_formatter(mdates.DateFormatter('%M'))
ax.figure.set_size_inches(10,4)
ax.plot(love['datef'], love['magnitude'], 'g', linestyle=':', marker='*',
        ↳linewidth=2, markersize=20, alpha=0.5)
ax.plot(hate['datef'], hate['magnitude'], 'k', linestyle=':', marker='.',
        ↳linewidth=2, markersize=20, alpha=0.5)
ax.set_ylabel('magnitude')
plt.show()
```





```
[23]: fig, ax = plt.subplots()
ax.set_xlabel('Date July 10th ~3:30 P.M')
ax.set_ylabel('score')
ax.grid(False)
ax.xaxis.set_major_locator(mdates.MinuteLocator(interval=5))
ax.xaxis.set_major_formatter(mdates.DateFormatter('%M'))
ax.figure.set_size_inches(10,4)
ax.plot(love['datef'], love['score'], 'b', linestyle=':', marker='.', linewidth=2, markersize=20, alpha=0.5)
ax.plot(hate['datef'], hate['score'], 'r', linestyle=':', marker='.', linewidth=2, markersize=20, alpha=0.5)
plt.show()
```



[ ]: