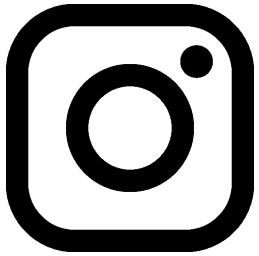
The Power of Instagram !



Data Cleaning

Hashtags	Caption
beautiful, summer, fashion, love, cute, food, ...	We were born to be REAL, not to be perfect. ...
teen, model, brunette, selfie, hollister, snap...	Gotta run, but first, let me take selfie. Me...
alexandani, cute, llbean, beanboots, anthropol...	dress to impress\U0001f457
alexandani, cute, llbean, beanboots, anthropol...	dress to impress\U0001f457
abercrombieandfitch, shopping, love, hollister...	#love #shopping #shoppen #hollister #abercro...

Hashtags	Caption
28	41
22	30
29	39
29	53
30	70

Data Cleaning

Anger	Contempt	Disgust	Fear	Happiness	Neutral	Sadness	Surprise
0.083862	0.015089	0.000583	0.000026	0.000017	0.890586	0.009657	0.000181
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.064922	0.005743	0.008687	0.011182	0.002374	0.220634	0.002961	0.683498

Emotion	Emotion_value
Neutral	0.890586
NaN	0.000000
NaN	0.000000
NaN	0.000000
Surprise	0.683498

Emotion_Anger	Emotion_Happiness	Emotion_Neutral	Emotion_Sadness	Emotion_Surprise
0.0	0.0	0.89	0.0	0.00
0.0	0.0	0.00	0.0	0.00
0.0	0.0	0.00	0.0	0.00
0.0	0.0	0.00	0.0	0.00
0.0	0.0	0.00	0.0	0.68

Data Cleaning

	UserId
0	1171579752855683619_212070047
1	1171594777274371222_176762322
2	1171407552643586413_581125501
3	1171407552643586413_581125501
4	1171508187966229230_2797323089

	UserId	PostId
0	212070047	1171579752855683619
1	1465589112	1171599907010659566
2	1477200908	1171567950101886268
3	1477200908	1171514703597155774
4	298780473	1171194809692940825

Web Scraping

ntrée [40]:
```
from selenium import webdriver
from bs4 import BeautifulSoup

driver= webdriver.Chrome(r'C:\Program Files (x86)\Google\Chrome\Application\chromedriver.exe')

followers=[]
for i in Brands:
    j=Brands[i]
    url='https://www.instagram.com/'+str(j)
    driver.get(url)
    soup = BeautifulSoup(driver.page_source,"lxml")
    for item in soup.select('a.-nal3>span.g47SY'):
        followers.append(item)
followers
```

Out[40]:
```
[<span class="g47SY">2 575</span>,
 <span class="g47SY" title="4 429 473">4,4m</span>,
 <span class="g47SY">240</span>,
 <span class="g47SY">1 017</span>,
 <span class="g47SY" title="2 790 522">2,7m</span>,
 <span class="g47SY">0</span>,
 <span class="g47SY">3 637</span>,
 <span class="g47SY" title="8 998 663">8,9m</span>,
```

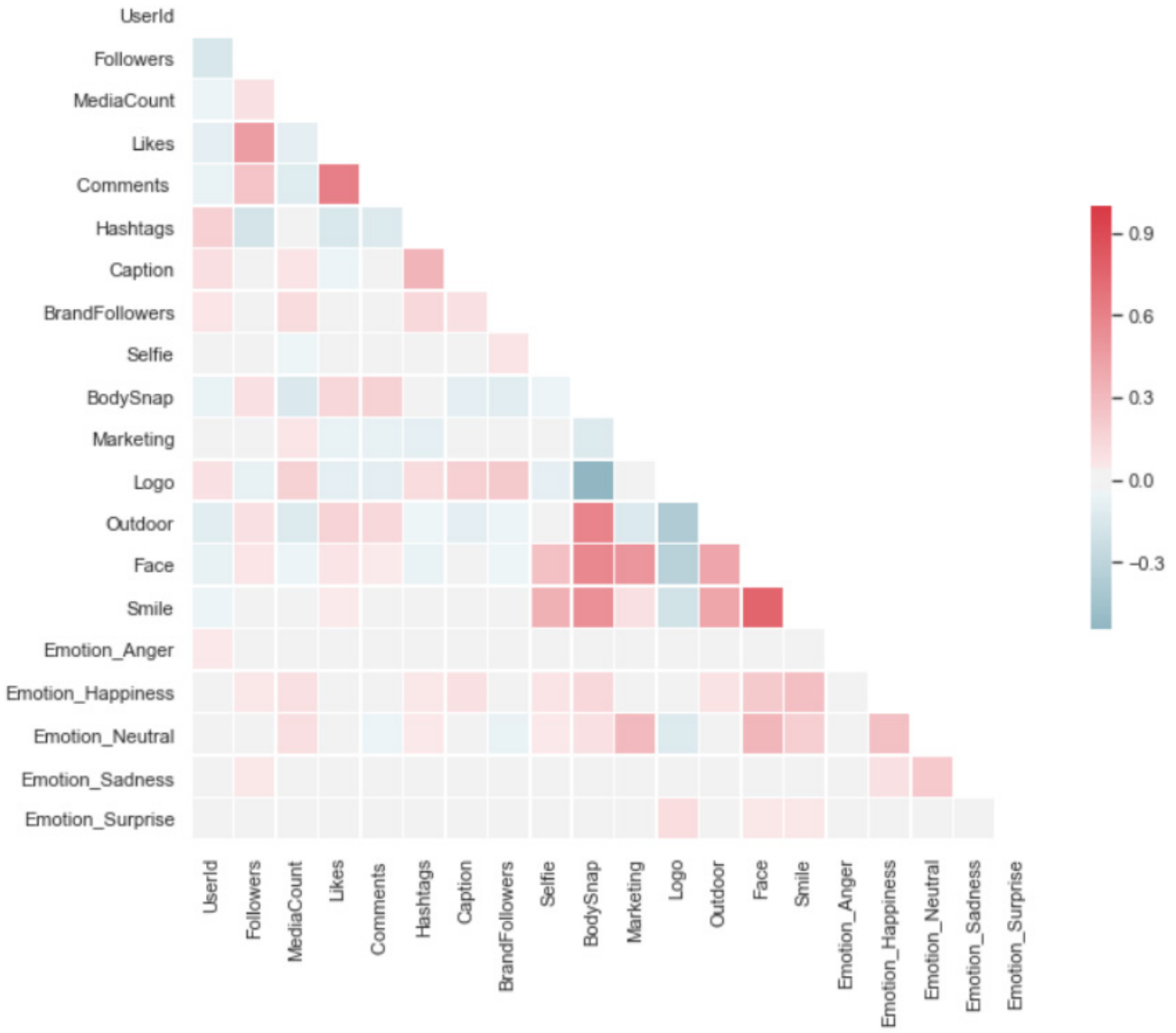
BrandName	BrandCategory	BrandFollowers
Brioni	Small couture	256 072
Topshop	High street	9 985 978
Ermenegildozegna	Small couture	598 489
Isabelmarant	Designer	1 677 342
Tiffany	Mega couture	11 113 807

Aggregation by User Id

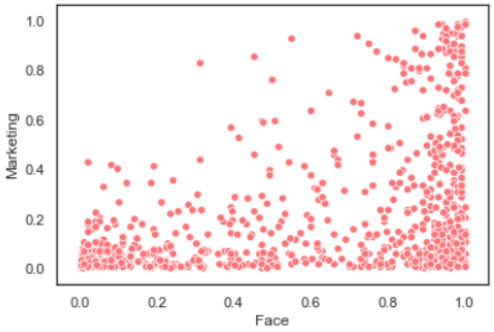
```
data2=df2.groupby('UserId').agg({"Likes":"mean",'Comments ':'mean','Followers':'mean','MediaCount':'mean',
                                 'Hashtags':'mean','Caption':"mean",'BrandName':mode_agg,'BrandFollowers':'median',
                                 'Selfie':'mean','BodySnap':'mean','Marketing':'mean','Logo':'mean','Outdoor':'mean',
                                 'Face':'mean','Smile':'mean','Outdoor':'mean', 'Emotion_Anger':'sum','Emotion_Happiness':'sum',
                                 'Emotion_Neutral':'sum','Emotion_Sadness':'sum','Emotion_Surprise':'sum'})
```

	UserId	Followers	MediaCount	Likes	Comments	Hashtags	Caption	BrandName	BrandFollowers	Selfie	...	Marketing	Logo	Outdoor	Face	Smile	Emotion_Anger	Emotion_Happiness	Emotion_Neutral	Emotion_Sadness	Emotion_Surprise
0	20922	27341.0	2392.0	302.0	18.0	24.0	63.0	Urbanoutfitters	8977868.0	0.00	...	0.02	0.46	0.02	0.01	0.01	0.0	0.0	0.00	0.0	0.0
1	90221	18165.0	25706.0	32.0	0.0	27.0	31.0	Ermenegildozegna	598489.0	0.00	...	0.51	0.11	0.13	0.83	0.05	0.0	0.0	0.97	0.0	0.0
2	98546	36269.0	2483.0	495.0	3.0	5.0	24.0	Acnestudios	2790522.0	0.09	...	0.01	0.11	0.03	0.24	0.07	0.0	0.0	0.00	0.0	0.0
3	102135	21691.0	1531.0	364.0	2.0	6.0	45.0	Burberry	16108823.0	0.45	...	0.18	0.72	0.04	0.95	0.52	0.0	1.0	0.00	0.0	0.0
4	359370	36113.0	2066.0	450.0	52.0	5.0	31.0	Vince	327184.0	0.00	...	0.19	0.09	0.45	0.79	0.09	0.0	0.0	0.00	0.0	0.0

Correlations



```
sns.scatterplot(data=data2,x=data2.Face[data2.Face!=0],y=data2.Marketing[data2.Marketing!=0],color='#ff7276');
```




```
data2.Face[data2.Face!=0].corr(data2.Marketing[data2.Marketing!=0])
```

```
0.4530398943921814
```

```
df.Face[df.Face!=0].corr(df.Marketing[df.Marketing!=0])
```

```
0.47898456001518364
```

Correlation Between Marketing & Face

```
data2.Emotion_Happiness.corr(data2.Smile)
```

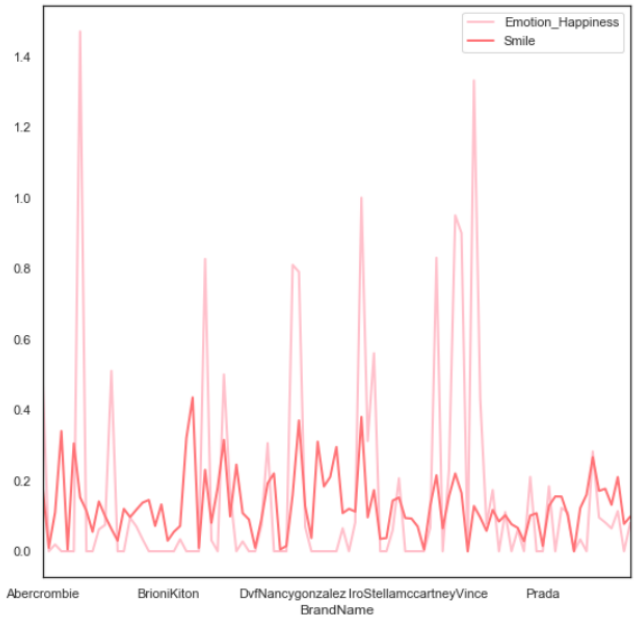
```
0.28235655149996763
```

```
df2.Emotion_Happiness.corr(df2.Smile)
```

```
0.4429865914975567
```

```
data2.groupby('BrandName')[['Emotion_Happiness','Smile']].mean().plot(figsize=[9,9],color=['pink','#ff7276'],linewidth=2)
```

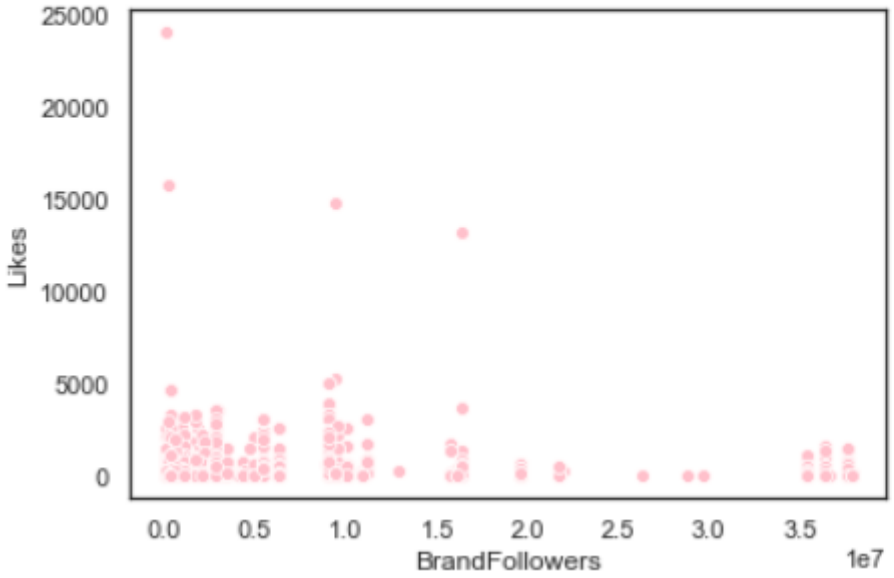
```
<matplotlib.axes._subplots.AxesSubplot at 0x2242e2a6438>
```



No Correlation Between Smile & Happiness

```
sns.scatterplot(x='BrandFollowers',y='Likes',data=data2,color='pink')
```

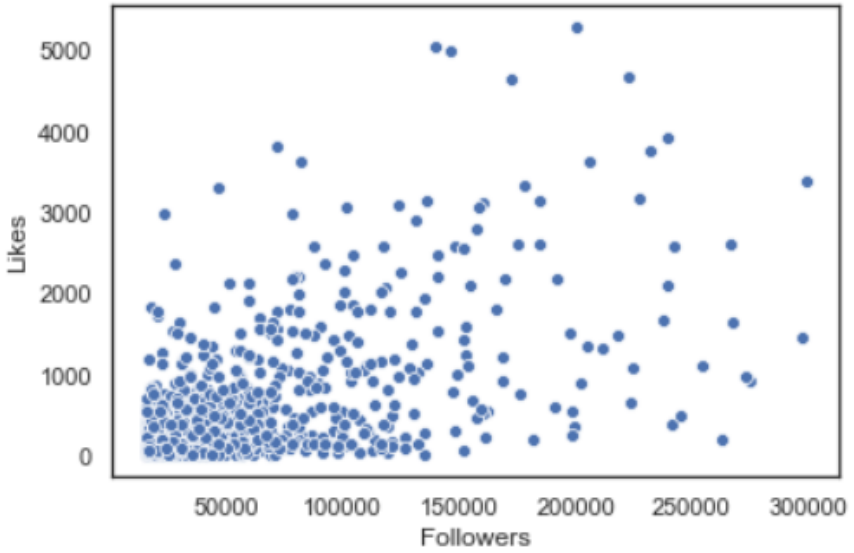
```
<matplotlib.axes._subplots.AxesSubplot at 0x22446b39390>
```



Correlation Between
Brand Followers & Likes

Correlation Between
Followers & Likes

```
sns.scatterplot(data=data2,x='Followers',y='Likes');
```




```
data2.Likes.corr(data2.Followers)
```

```
0.5898973328690647
```

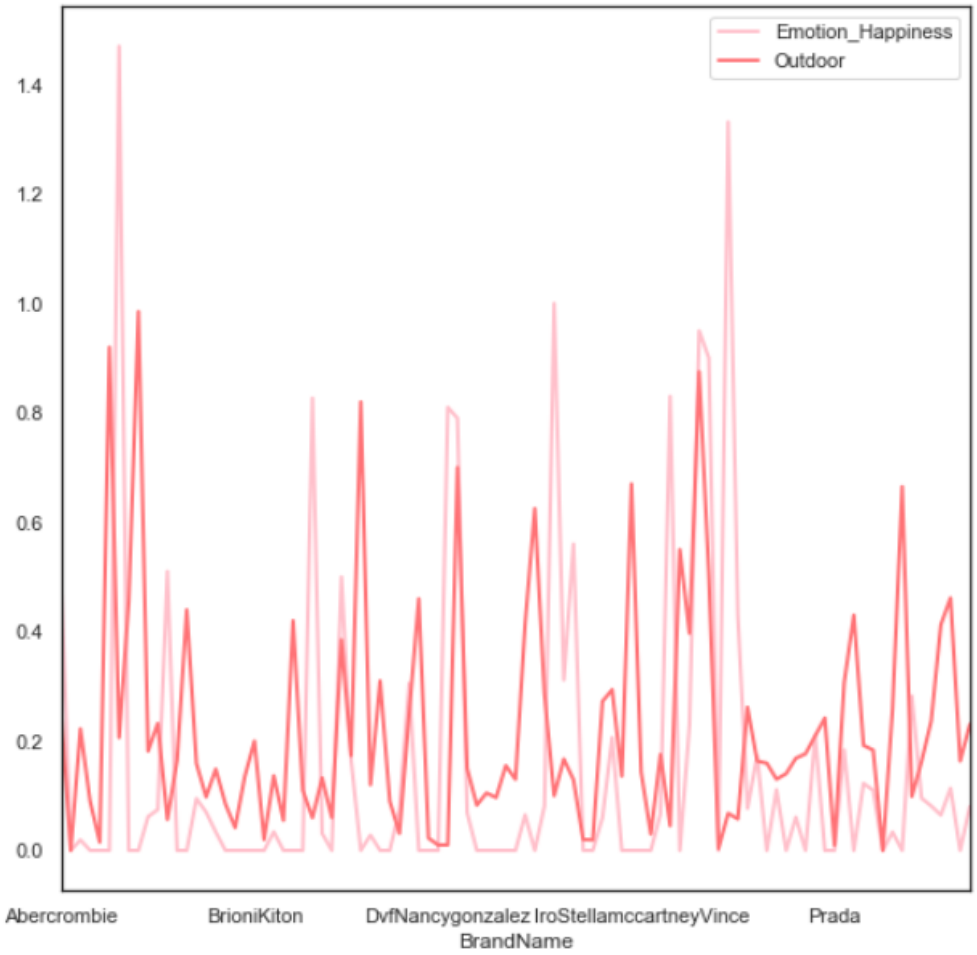
```
df.Likes.corr(df.Followers)
```

```
0.867658801811564
```

No Correlation Between Happiness & Outdoor

```
data2.groupby('BrandName')[['Emotion_Happiness','Outdoor']].mean().plot(figsize=[9,9],color=['pink','#ff7276'],linewidth=2)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x2243629dfd0>
```




```
df2.Emotion_Happiness.corr(df2.Outdoor)
```

```
0.14333448478119973
```

No Correlation Between
Brand Followers & Happiness Level in Publications