In [2]:

```
import pandas as pd
import numpy as np
import re
import seaborn as sns
import matplotlib.pyplot as plt
import math
```

In [3]:

```
df = pd.read_csv('Downloads/elevate_analytics_case_data.csv')
```

```
C:\Users\Sarvesh Shah\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3049:
DtypeWarning: Columns (6) have mixed types. Specify dtype option on import or set
low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

In [77]:

```
df.head()
```

Out[77]:

	brand_id	retailer_id	brand_relationship_id	brand_relationship_created_at	brand_relationship_confirmed_at	confirmation_reason	nor
0	20940	NaN	262526	2019-06-16 23:59:30	NaN	NaN	
1	20940	NaN	262525	2019-06-16 23:58:26	NaN	NaN	
2	19207	NaN	262524	2019-06-16 23:58:08	NaN	NaN	
3	20940	NaN	262523	2019-06-16 23:47:47	NaN	NaN	
4	20940	NaN	262522	2019-06-16 23:46:54	NaN	NaN	

5 rows × 21 columns

In [78]:

```
list(df)
```

Out[78]:

```
['brand_id',
 'retailer_id',
 'brand_relationship_id',
 'brand_relationship_created_at',
 'brand_relationship_confirmed_at',
 'confirmation_reason',
 'normalized_referer',
 'outgoing_email_id',
 'email_sent_at',
 'brand_relationship_order_id',
 'retailer_signed_up_at',
 'retailer_placed_first_order_at',
 'retailer_placed_first_confirmed_order_at',
 'power_retailer_converted_at',
 'retailer_gmv',
 'retailer_business_type',
 'brand_stockist_count',
 'brand_first_active_at',
 'brand_adopted_elevate_at',
 'power_maker_converted_at',
 'account_owner']
```

In [79]:

```
df.describe()
```

Out[79]:

	brand_id	retailer_id	brand_relationship_id	outgoing_email_id	brand_relationship_order_id	retailer_gmv	brand_stocl
count	218538.000000	32925.000000	218538.000000	5.915800e+04	420.000000	32925.000000	1220
mean	11762.567311	104335.204465	127717.487860	2.766829e+15	173303.869048	670.518482	4
std	8053.428540	29486.703843	77628.404937	5.316915e+18	75094.969090	2467.437360	7
min	5.000000	414.000000	1.000000	-9.223306e+18	26044.000000	0.000000	
25%	6913.000000	84371.000000	58978.250000	-4.662382e+18	104336.750000	0.000000	
50%	9935.000000	110802.000000	124190.500000	5.349338e+16	181488.500000	0.000000	2
75%	14074.000000	127710.000000	197126.750000	4.457374e+18	233532.250000	369.250000	5
max	48480.000000	148515.000000	262526.000000	9.222708e+18	303608.000000	126287.810000	100

In [80]:

```
df.shape
```

Out[80]:

```
(218538, 21)
```

In [81]:

```
df.isna().sum()
```

Out[81]:

```
brand_id                                    0
retailer_id                            185613
brand_relationship_id                       0
brand_relationship_created_at               0
brand_relationship_confirmed_at        185614
confirmation_reason                    185614
normalized_referer                     200120
outgoing_email_id                      159380
email_sent_at                          159380
brand_relationship_order_id            218118
retailer_signed_up_at                  185613
retailer_placed_first_order_at         202168
retailer_placed_first_confirmed_order_at    204534
power_retailer_converted_at            215167
retailer_gmv                           185613
retailer_business_type                 185689
brand_stockist_count                    96454
brand_first_active_at                      11
brand_adopted_elevate_at                 1239
power_maker_converted_at                67291
account_owner                               0
dtype: int64
```

In [64]:

```
df.dtypes
```

Out[64]:

```
brand_id                                  category
retailer_id                               category
brand_relationship_id                     category
brand_relationship_created_at       datetime64[ns]
brand_relationship_confirmed_at     datetime64[ns]
confirmation_reason                         object
normalized_referer                          object
outgoing_email_id                             bool
email_sent_at                       datetime64[ns]
brand_relationship_order_id               category
retailer_signed_up_at               datetime64[ns]
retailer_placed_first_order_at      datetime64[ns]
retailer_placed_first_confirmed_order_at    datetime64[ns]
```

```
power_retailer_converted_at                datetime64[ns]
retailer_gmv                                      float64
retailer_business_type                             object
brand_stockist_count                              float64
brand_first_active_at                      datetime64[ns]
brand_adopted_elevate_at                   datetime64[ns]
power_maker_converted_at                   datetime64[ns]
account_owner                                      object
dtype: object
```

In [4]:

```
for time in
['brand_first_active_at','brand_adopted_elevate_at','power_maker_converted_at','retailer_signed_up_
at','retailer_placed_first_order_at',
             'retailer_placed_first_confirmed_order_at','power_retailer_converted_at', 'brand_relat
ionship_created_at','brand_relationship_confirmed_at','email_sent_at']:
    df['{}'.format(time)] = pd.to_datetime(df['{}'.format(time)])

for col in
['retailer_id','brand_relationship_id','brand_id','brand_relationship_order_id','normalized_referer
']:
     df['{}'.format(col)] = df['{}'.format(col)].astype('category')
```

Hypothesis 1:

Part 1

Analysis: We ran some t-tests to examine the difference in the sign-up rate between retailers who arrived at the sign-up page using Faire direct link in email (Group 1) vs. retailers who arrived at the sign-up page from a different source (Group 2).

In [5]:

```
# Generating extra columns for analysis

df['retailer_sign_up'] = df['retailer_id'].notnull()
df['clicked_email'] = df['outgoing_email_id'].notnull()
df['retailer_placed__fc_order'] = df['retailer_placed_first_confirmed_order_at'].notnull()
```

We start out by looking at the effect of email on sign up rate. We randomly sampled 40,000 samples for each of our groups.

In [186]:

```
dfclick = df[(df.clicked_email)].sample(40000)
dfnclick = df[(df.clicked_email==False)].sample(40000)

temp =
dfclick.append(dfnclick).groupby(['clicked_email','retailer_sign_up','retailer_placed__fc_order'])
[['brand_id']].count()
display(temp)

n1 = float(temp.loc[False,:,:].sum())
n2 = float(temp.loc[True,:,:].sum())

p1 = float(temp.loc[False,True,:].sum()/n1)
q1 = float(1 - p1)

p2 = float(temp.loc[True,True,:].sum()/n2)
q2 = float(1 - p2)

print('p1=',p1,'\np2=',p2,'\nq1=',q1,'\nq2=',q2,'\nn1=',n1,'\nn2=',n2)

t = (p1 - p2) / math.sqrt((p1*q1/n1)+(p2*q2/n2))
print('\nT-score:',t)

if t>=1.96 or t<=-1.96:
    print('\nP-value p < 0.05, Significant')
else:
    print('\nP-value p > 0.05, Not Significant')
```

clicked_email	retailer_sign_up	retailer_placed__fc_order	brand_id
False	False	False	36313
	True	False	2155
		True	1532
True	False	False	27708
	True	False	7004
		True	5288

```
p1= 0.092175
p2= 0.3073
q1= 0.907825
q2= 0.6927
n1= 40000.0
n2= 40000.0

T-score: -79.00875621841654

P-value p < 0.05, Significant
```

Part 2:

Analysis: We evaluated the difference in the acquisition rate by also conducting a t-test with the significance level (α) of 0.05. We randomly sampled 6,000 samples for each of our groups.

In [34]:

```
dfclick = df[(df.clicked_email)&(df.retailer_sign_up)].sample(6000)
dfnclick = df[(df.clicked_email==False)&(df.retailer_sign_up)].sample(6000)

temp =
dfclick.append(dfnclick).groupby(['clicked_email','retailer_sign_up','retailer_placed__fc_order'])
[['brand_id']].count()
display(temp)

n1 = float(temp.loc[False,True,:].sum())
n2 = float(temp.loc[True,True,:].sum())

p1 = float(temp.loc[False,True,True]/n1)
q1 = float(1 - p1)

p2 = float(temp.loc[True,True,True]/n2)
q2 = float(1 - p2)


print('p1=',p1,'\np2=',p2,'\nq1=',q1,'\nq2=',q2,'\nn1=',n1,'\nn2=',n2)

t = (p1 - p2) / math.sqrt((p1*q1/n1)+(p2*q2/n2))
print('\nT-score:',t)

if t>=1.96 or t<=-1.96:
    print('\nP-value p < 0.05, Significant')
else:
    print('\nP-value p > 0.05, Not Significant')
```

clicked_email	retailer_sign_up	retailer_placed__fc_order	brand_id
False	True	False	3477
		True	2523
True	True	False	3442
		True	2558

```
p1= 0.4205
p2= 0.42633333333333334
```

```
q1= 0.5795
q2= 0.5736666666666667
n1= 6000.0
n2= 6000.0

T-score: -0.6466510264780151

P-value p > 0.05, Not Significant
```

Supporting data: we performed a year by year analysis of the acquisition rate, which shows that the rate has decreased from 14.3% (2018) to 12.5% (2019) even though the sign-up rate is consistent, 30.18% (2018) vs. 30.97% (2019).

In [36]:

```
df['email_sent_year'] = df['email_sent_at'].dt.year
```

In [187]:

```
df_brand = df.groupby(['email_sent_year'])[['brand_relationship_id','retailer_id']].count()
df_brand['conversion_rate'] = df_brand['retailer_id']*100/df_brand['brand_relationship_id']
df_brand.sort_values(by=['brand_relationship_id'])
```

Out[187]:

email_sent_year	brand_relationship_id	retailer_id	conversion_rate
2018.0	23799	7183	30.181940
2019.0	35357	10949	30.966994

In [188]:

```
df_brand =
df.groupby(['email_sent_year']).agg({'brand_relationship_id':'count','retailer_placed__fc_order':'
sum'})
df_brand['acquisition_rate'] =
df_brand['retailer_placed__fc_order']*100/df_brand['brand_relationship_id']
df_brand.sort_values(by=['retailer_placed__fc_order'])
```

Out[188]:

email_sent_year	brand_relationship_id	retailer_placed__fc_order	acquisition_rate
2018.0	23799	3403.0	14.29892
2019.0	35357	4426.0	12.51803

Hypothesis 2: Decay in brand's number of referrals over years

In [150]:

```
# Pre processing
df = df[df['brand_first_active_at'].notnull()]

df['brand_joined_year'] = df['brand_first_active_at'].dt.year
df['brand_adopted_elevate_at_year'] = df['brand_adopted_elevate_at'].dt.year
df['email_sent_at_years'] = df['email_sent_at'].dt.year
```

Brand effort in inviting retailers by email

Brand effort metric:
No. of emails sent for the given year/no. of brands adopted faire elevate for the same year = % email sent

In [185]:

```
temp = df.groupby(['brand_adopted_elevate_at_year','brand_joined_year']).agg({'brand_id':'count','
email_sent_at':'count'})
temp['perc_email_sent'] = temp['email_sent_at']*100./temp['brand_id']

temp[['perc_email_sent']]
```

Out[185]:

		perc_email_sent
brand_adopted_elevate_at_year	**brand_joined_year**	
2018.0	**2017**	35.658350
	2018	22.679501
2019.0	**2017**	47.361013
	2018	33.593787
	2019	30.555808