Анализ датасета Facebook

Описание Facebook metrics Data Set: ¶

(ссылка: https://archive.ics.uci.edu/ml/datasets/Facebook+metrics (https://archive.ics.uci.edu/ml/datasets/Facebook+metrics))

Данные связаны с публикациями, опубликованными в течение 2014 года на странице известного косметического бренда в Facebook. Этот набор данных содержит 500 строк объектов и 19 атрибутов. Из них 7 фич, известных до публикации поста, и 12 фич для оценки показателей поста Пропущенные значения имеют вид NaN

ХАРАКТЕРИСТИКИ(пояснения):

Page total likes: число людей, лайкавших страницу
Type: тип контента Link(ссылка), Photo(фото), Status(статус), Video(видео).
Category: 1-action(акция), 2-product(товар), 3-inspiration(неявная связь с брендом)
Post Month: месяц публикации(1-Январь...12-Декабрь)
Post Weekday: неделя публикации (1-Sunday,7-Saturday)
Post Hour: час публикации(0-23)
Paid: если компания платила Facebook за рекламу (0-нет, 1-да)
Lifetime Post Total Reach: число людей, которые видели публикацию (уникальные пользователи).
Lifetime Post Total Impressions: показы - это число раз, когда отображается сообщение со страницы, независимо от того, щелкнуто оно или нет. Люди могут видеть несколько показов одного и того же сообщения. Например, кто-то может увидеть обновление страницы в Ленте Новостей один раз, а затем второй раз, если друг делится им.
Lifetime Engaged Users: число людей, щелкнувших в любом месте поста (уникальные пользователи).
Lifetime Post Consumers: число людей, щелкнувших в любом месте поста.
Lifetime Post Consumptions: число кликов в любом месте поста.
Lifetime Post Impressions by people who have liked a Page: общее число показов поста только людям, которые лайкали страницу.
Lifetime Post reach by people who like a Page: число людей, которые видели пост, потому что они лайкали эту страницу (уникальные пользователи).
Lifetime People who have liked a Page and engaged with a post: число людей, которые лайкали страницу и которые нажали в любом месте поста (уникальные пользователи).
comment: число комментариев
like: число лайков
share: число людей, поделившихся записью
Total Interactions: 'comment'+'like'+'share' - всего взаимодейтсвий

In [1]:

```
import pandas as pd
import numpy as np
from IPython.display import HTML, display
import tabulate
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")

data=pd.read_csv('C:/Users/kotsi/Downloads/dataset_Facebook.csv', sep=';')
data['Paid'] = data.Paid.astype('bool')
data.head()
```

Out[1]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lif Enc
0	139441	Photo	2	12	4	3	False	2752	5091	178
1	139441	Status	2	12	3	10	False	10460	19057	145
2	139441	Photo	3	12	3	3	False	2413	4373	177
3	139441	Photo	2	12	2	10	True	50128	87991	221
4	139441	Photo	2	12	2	3	False	7244	13594	671

In [2]:

```
data.describe()
```

Out[2]:

	Page total likes	Category	Post Month	Post Weekday	Post Hour	Lifetime Post Total Reach
count	500.000000	500.000000	500.000000	500.000000	500.000000	500.00000
mean	123194.176000	1.880000	7.038000	4.150000	7.840000	13903.36000
std	16272.813214	0.852675	3.307936	2.030701	4.368589	22740.78789
min	81370.000000	1.000000	1.000000	1.000000	1.000000	238.00000
25%	112676.000000	1.000000	4.000000	2.000000	3.000000	3315.00000
50%	129600.000000	2.000000	7.000000	4.000000	9.000000	5281.00000
75%	136393.000000	3.000000	10.000000	6.000000	11.000000	13168.00000
max	139441.000000	3.000000	12.000000	7.000000	23.000000	180480.00000

In [4]:

```
data.dtypes
```

Out[4]:

```
Page total likes                                                 i
nt64
Type                                                            ob
ject
Category                                                         i
nt64
Post Month                                                       i
nt64
Post Weekday                                                     i
nt64
Post Hour                                                        i
nt64
Paid
bool
Lifetime Post Total Reach                                        i
nt64
Lifetime Post Total Impressions                                  i
nt64
Lifetime Engaged Users                                           i
nt64
Lifetime Post Consumers                                          i
nt64
Lifetime Post Consumptions                                       i
nt64
Lifetime Post Impressions by people who have liked your Page     i
nt64
Lifetime Post reach by people who like your Page                 i
nt64
Lifetime People who have liked your Page and engaged with your post   i
nt64
comment                                                          i
nt64
like                                                            flo
at64
share                                                           flo
at64
Total Interactions                                               i
nt64
dtype: object
```

In [201]:

```
# Для этих стобцов нам бесполезно считать среднее, максимальное и минимальное
categorical_columns=data[['Type','Category','Post Month', 'Post Weekday', 'Post Hour',
'Paid']] #мода, медиана

numerical_columns=data.drop(['Category', 'Post Month', 'Post Weekday', 'Post Hour', 'Pa
id'], axis=1)

def static_counted(columns):
    measures=dict()
    measures['0_mean']=columns.mean()
    measures['1_max']=columns.max()
    measures['2_min']=columns.min()
    measures['3_median']=columns.median()
    measures=pd.DataFrame(measures).T
    return measures
```

In [202]:

```
# Мода
categorical_columns.mode()
```

Out[202]:

	Type	Category	Post Month	Post Weekday	Post Hour	Paid
0	Photo	1	10	7	3	False

In [203]:

```
static_counted(numerical_columns)
```

Out[203]:

	Lifetime Engaged Users	Lifetime People who have liked your Page and engaged with your post	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Post Impressions by people who have liked your Page	Lifetime Post Total Impressions
0_mean	920.344	609.986	798.772	1415.13	16766.4	29585.9
1_max	11452	4376	11328	19779	1107833	1110282
2_min	9	9	9	9	567	570
3_median	625.5	412	551.5	851	6255.5	9051

In [204]:

```
numerical_columns.mode().T
```

Out[204]:

	0	1	2	3	4	5	6	7	8	9	...	
Page total likes	136393	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
Type	Photo	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
Lifetime Post Total Reach	677	2232	2645	2938	3322	3358	3414	3528	3754	5280	...	Na
Lifetime Post Total Impressions	4372	6503	7004	8533	8745	12735	NaN	NaN	NaN	NaN	...	Na
Lifetime Engaged Users	537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
Lifetime Post Consumers	182	298	319	322	335	374	482	503	513	642	...	Na
Lifetime Post Consumptions	431	513	599	652	719	730	795	889	NaN	NaN	...	Na
Lifetime Post Impressions by people who have liked your Page	1210	1284	2541	2888	3675	4911	4935	5010	5732	NaN	...	Na
Lifetime Post reach by people who like your Page	690	704	1228	1640	1674	1722	2044	2124	2162	2174	...	32
Lifetime People who have liked your Page and engaged with your post	403	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
comment	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
like	98	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
share	13	14	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na
Total Interactions	0	39	75	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	Na

14 rows × 31 columns

In [205]:

```
types=list(data['Type'].unique())
types
```

Out[205]:

```
['Photo', 'Status', 'Link', 'Video']
```

In [206]:

```
un_photos=categorical_columns.drop(np.where(categorical_columns['Type']!='Photo')[0])
photos=numerical_columns.drop(np.where(numerical_columns['Type']!='Photo')[0])
un_statuses=categorical_columns.drop(np.where(categorical_columns['Type']!='Status')[0
])
statuses=numerical_columns.drop(np.where(numerical_columns['Type']!='Status')[0])
un_linkes=categorical_columns.drop(np.where(categorical_columns['Type']!='Link')[0])
linkes=numerical_columns.drop(np.where(numerical_columns['Type']!='Link')[0])
un_videos=categorical_columns.drop(np.where(categorical_columns['Type']!='Video')[0])
videos=numerical_columns.drop(np.where(numerical_columns['Type']!='Video')[0])

un_arr=[un_photos, un_statuses, un_linkes, un_videos]
arr=[photos, statuses, linkes, videos]
```

In [207]:

```
for i in un_arr:
    print(i['Type'].unique())
    display(HTML(tabulate.tabulate(i.mode(), tablefmt='html', headers=categorical_colum
ns.columns.values[1:])))
```

```
['Photo']
```

		Category	Post Month	Post Weekday	Post Hour	Paid
0	Photo	1	10	7	3	False

```
['Status']
```

		Category	Post Month	Post Weekday	Post Hour	Paid
0	Status	2	12	5	10	0
1	nan	nan	nan	7	nan	nan

```
['Link']
```

		Category	Post Month	Post Weekday	Post Hour	Paid
0	Link	1	12	6	3	False

```
['Video']
```

		Category	Post Month	Post Weekday	Post Hour	Paid
0	Video	1	11	2	11	1
1	nan	nan	nan	3	nan	nan

In [208]:

```
for i in arr:
    print(i['Type'].unique())
    display(HTML(tabulate.tabulate(static_counted(i.drop(["Type"], axis=1)), tablefmt=
'html', headers=numerical_columns.drop(["Type"], axis=1).columns.values[:])))
```

['Photo']

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	I Impr by wl lik
0_mean	122354	13137.8	28994.5	818.946	690.432	1299.03	
1_max	139441	180480	1.11028e+06	11452	11328	19779	1.107
2_min	81370	238	570	9	9	9	
3_median	128032	4675	8118.5	605.5	528.5	827	

['Status']

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	I Impr by wl lik
0_mean	132647	13078.9	24244.5	2040.22	1949.56	2838.87	
1_max	139441	31136	59964	6164	5934	9237	
2_min	104070	3930	7509	128	86	112	
3_median	135713	11096	20849	1701	1599	2201	

['Link']

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	I Impr by wl lik
0_mean	116363	18544.6	28725.5	342.818	292.682	374.091	
1_max	138895	70912	229733	1374	1106	1345	
2_min	85979	1536	3094	24	23	26	
3_median	115396	7422	9773	244	205	290	

['Video']

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	I Impr by wl lik
0_mean	135015	51205.7	102622	1707	1584.71	2600.14	
1_max	138895	139008	277100	3872	3822	7327	
2_min	126424	13544	30235	459	411	539	
3_median	137893	30624	56950	1779	1643	2331	

In [209]:

```
for i in arr:
    print(i['Type'].unique())
    display(HTML(tabulate.tabulate(i.drop(["Type"], axis=1).mode(), tablefmt='html', he
aders=numerical_columns.drop(["Type"], axis=1).columns.values)))
```

```
['Photo']
```

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Pos Impressions by people who have liked you Page
0	124940	677	4372	537	298	431	1210
1	nan	2232	6503	nan	319	513	1284
2	nan	2938	7004	nan	335	599	2541
3	nan	3322	8533	nan	374	652	2888
4	nan	3358	8745	nan	482	719	3675
5	nan	3414	12735	nan	503	730	4911
6	nan	3528	nan	nan	513	795	4935
7	nan	3754	nan	nan	642	889	5010
8	nan	5280	nan	nan	nan	nan	5732
9	nan	5290	nan	nan	nan	nan	nan
10	nan	9528	nan	nan	nan	nan	nan
11	nan	32208	nan	nan	nan	nan	nan
12	nan	nan	nan	nan	nan	nan	nan
13	nan	nan	nan	nan	nan	nan	nan
14	nan	nan	nan	nan	nan	nan	nan
15	nan	nan	nan	nan	nan	nan	nan
16	nan	nan	nan	nan	nan	nan	nan
17	nan	nan	nan	nan	nan	nan	nan
18	nan	nan	nan	nan	nan	nan	nan
19	nan	nan	nan	nan	nan	nan	nan
20	nan	nan	nan	nan	nan	nan	nan
21	nan	nan	nan	nan	nan	nan	nan
22	nan	nan	nan	nan	nan	nan	nan
23	nan	nan	nan	nan	nan	nan	nan
24	nan	nan	nan	nan	nan	nan	nan
25	nan	nan	nan	nan	nan	nan	nan
26	nan	nan	nan	nan	nan	nan	nan

```
['Status']
```

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Pos Impressions by people who have liked you Page
0	139441	3930	7509	128	86	1692	5009
1	nan	6692	13092	130	108	nan	7549
2	nan	8260	14305	135	119	nan	9562
3	nan	8284	14759	199	126	nan	9684
4	nan	8488	14847	304	286	nan	10266
5	nan	8628	15104	870	843	nan	11710
6	nan	8728	15294	908	847	nan	11970
7	nan	8896	16054	951	861	nan	12905
8	nan	9056	16362	997	937	nan	12996
9	nan	9120	16811	1006	953	nan	13246
10	nan	9124	16978	1132	1032	nan	13838
11	nan	9232	17099	1151	1073	nan	13887
12	nan	9236	17202	1191	1130	nan	14014
13	nan	9504	17776	1264	1209	nan	14450
14	nan	10060	17827	1341	1270	nan	15220
15	nan	10152	19057	1457	1361	nan	15382
16	nan	10188	19279	1473	1397	nan	15418
17	nan	10460	19556	1476	1407	nan	15850
18	nan	10472	19680	1480	1426	nan	16018
19	nan	10744	20327	1521	1441	nan	16034
20	nan	10824	20573	1530	1462	nan	16095
21	nan	10956	20691	1636	1570	nan	17272
22	nan	11096	20849	1701	1599	nan	17502
23	nan	11844	21080	1843	1724	nan	17627
24	nan	12044	22538	1967	1877	nan	17717
25	nan	12468	24198	2055	1912	nan	17748
26	nan	13152	24738	2143	1966	nan	18838

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetim Pos Impression by peopl who hav liked you Pag
27	nan	13216	24917	2240	2193	nan	1907
28	nan	13280	25269	2295	2235	nan	1959
29	nan	13872	25666	2417	2327	nan	2003
30	nan	14424	26000	2543	2438	nan	2019
31	nan	15296	27468	2573	2448	nan	2054
32	nan	15576	27513	2664	2531	nan	2100
33	nan	15816	30514	2675	2567	nan	2210
34	nan	16576	30612	2733	2570	nan	2313
35	nan	17360	31448	2750	2584	nan	2333
36	nan	17576	33058	2806	2654	nan	2358
37	nan	17912	33613	2827	2781	nan	2436
38	nan	18320	34143	3370	3244	nan	2553
39	nan	19552	34774	3572	3464	nan	2558
40	nan	20168	35904	3742	3682	nan	2688
41	nan	21256	41906	4258	4100	nan	3545
42	nan	21744	42313	4840	4754	nan	3597
43	nan	23832	42334	5352	5202	nan	3781
44	nan	31136	59964	6164	5934	nan	3784

```
['Link']
```

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Pos Impressions by people who have liked you Page
0	138353	1536	3094	66	322	26	230
1	nan	1925	3481	nan	nan	70	233
2	nan	2645	4270	nan	nan	71	290
3	nan	2933	5144	nan	nan	97	397
4	nan	3454	5808	nan	nan	99	426
5	nan	3616	6853	nan	nan	118	628
6	nan	4664	6887	nan	nan	126	662
7	nan	4938	7910	nan	nan	130	673
8	nan	5168	7927	nan	nan	170	704
9	nan	5730	8371	nan	nan	223	719
10	nan	6876	9463	nan	nan	251	896
11	nan	7968	10083	nan	nan	329	910
12	nan	9356	10885	nan	nan	337	940
13	nan	12540	13023	nan	nan	389	1207
14	nan	18480	14986	nan	nan	396	1252
15	nan	21176	19301	nan	nan	460	1385
16	nan	34192	28438	nan	nan	477	1391
17	nan	35360	42092	nan	nan	505	1576
18	nan	39600	45260	nan	nan	581	1691
19	nan	45920	54779	nan	nan	763	1784
20	nan	68992	94172	nan	nan	1267	2290
21	nan	70912	229733	nan	nan	1345	4233

```
['Video']
```

	Page total likes	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Post Impressions by people who have liked your Page
0	137893	13544	30235	459	411	539	21436
1	nan	16416	31950	517	458	667	24667
2	nan	21872	40413	1141	1068	1728	26622
3	nan	30624	56950	1779	1643	2331	30131
4	nan	36208	61262	2080	1735	2356	32033
5	nan	100768	220447	2101	1956	3253	59658
6	nan	139008	277100	3872	3822	7327	107502

самый популярный объект

Самый популярный объект в выборке и почему

1.Явный вид

Если принимать что события "comment", "like", "share" равнозначны для популярности,то можно смотреть только на столбец Total interactions

Самый популярным постом можно считать пост 244, который имеет наибольшее число взимодейтсвий

In [210]:

```
The_best=numerical_columns.drop(['Page total likes','Type','Lifetime Post Impressions b
y people who have liked your Page','Lifetime Post reach by people who like your Page',
'Lifetime People who have liked your Page and engaged with your post','comment','like',
'share'], axis=1)
pd.Series(The_best.max())
data.loc[The_best['Total Interactions']==The_best['Total Interactions'].max()]
```

Out[210]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	L Er
244	130791	Photo	2	7	3	5	True	180480	319133	80

По этим результатам мы также видим, что пост набрал такое больше число "лайков" за счет активного продвижения в рекламе

Самым популярным в плане лайков вне зависимости от частоты показов поста является пост 87.

Его мы нашли через отношение "Лайков" к числу уникальных человек, у которых пост был виден

In [211]:

```
data.loc[(The_best['Total Interactions']/The_best['Lifetime Post Total Reach'])==(The_b
est['Total Interactions']/The_best['Lifetime Post Total Reach']).max()]
```

Out[211]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Li En
87	137177	Photo	3	11	3	10	False	4012	7955	55(

2. Неявно

Если мерить популярность не по числу видимых другим пользователям метрикам(лайк, коммент, поделились), а по числу взаимодействий(кликов) с постом, то самым популярным является пост 278

In [212]:

```
data.loc[(The_best['Lifetime Post Consumptions'])==(The_best['Lifetime Post Consumption
s']).max()]
```

Out[212]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions
278	126345	Photo	1	6	2	9	True	5854	11854

По числу взаимодействий пользователя со страницей самым популярным постом является пост 422, где на одного увидевшего пост человека приходится около 3,5 кликов

In [213]:

```
data.loc[(The_best['Lifetime Post Consumptions']/The_best['Lifetime Post Total Reach'])
==(The_best['Lifetime Post Consumptions']/The_best['Lifetime Post Total Reach']).max()]
```

Out[213]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetim Post Tota Impression
422	102112	Photo	1	3	1	19	False	238	570

In [214]:

```
data=data.drop(data.loc[data['Lifetime Post Consumers']>5000].index)
data=data.dropna()
data.shape
```

Out[214]:

```
(493, 19)
```

Корреляция

In [215]:

```
print(data.corr()['Lifetime Post Consumers'].abs().sort_values(ascending=False).head(19
))
```

```
Lifetime Post Consumers                                           1.0
00000
Lifetime Engaged Users                                            0.9
54537
Lifetime People who have liked your Page and engaged with your post    0.9
17066
Lifetime Post Consumptions                                        0.6
22288
Lifetime Post reach by people who like your Page                  0.6
06367
Lifetime Post Total Reach                                         0.4
95348
share                                                             0.4
27001
Total Interactions                                                0.4
24432
like                                                              0.4
17469
comment                                                           0.3
91607
Lifetime Post Total Impressions                                   0.3
62817
Lifetime Post Impressions by people who have liked your Page      0.2
82010
Post Month                                                        0.1
82346
Page total likes                                                  0.1
63651
Post Hour                                                         0.0
75401
Paid                                                              0.0
55354
Category                                                          0.0
19279
Post Weekday                                                      0.0
06984
Name: Lifetime Post Consumers, dtype: float64
```

In [216]:

```
data.corr()
```

Out[216]:

	Page total likes	Category	Post Month	Post Weekday	Post Hour	Paid	Life R
Page total likes	1.000000	-0.089393	0.940863	-0.019351	-0.148719	0.004988	-0.07
Category	-0.089393	1.000000	-0.125625	-0.048798	-0.110165	-0.024383	-0.14
Post Month	0.940863	-0.125625	1.000000	0.002870	-0.180475	-0.017943	-0.09
Post Weekday	-0.019351	-0.048798	0.002870	1.000000	0.039976	0.011714	-0.03
Post Hour	-0.148719	-0.110165	-0.180475	0.039976	1.000000	-0.063387	0.016
Paid	0.004988	-0.024383	-0.017943	0.011714	-0.063387	1.000000	0.130
Lifetime Post Total Reach	-0.070554	-0.142337	-0.093678	-0.034280	0.016049	0.130787	1.000
Lifetime Post Total Impressions	-0.097753	-0.095664	-0.097971	-0.026777	0.017495	0.054955	0.696
Lifetime Engaged Users	-0.110595	0.021026	-0.133543	-0.019829	0.055332	0.081467	0.593
Lifetime Post Consumers	-0.163651	-0.019279	-0.182346	0.006984	0.075401	0.055354	0.495
Lifetime Post Consumptions	-0.121257	-0.154525	-0.149147	0.007010	0.123321	0.063722	0.284
Lifetime Post Impressions by people who have liked your Page	-0.095862	-0.050352	-0.094560	-0.044684	0.040058	0.000337	0.325
Lifetime Post reach by people who like your Page	-0.058425	-0.111519	-0.091696	-0.062013	0.058457	0.099824	0.749
Lifetime People who have liked your Page and engaged with your post	0.023380	0.016352	-0.010555	0.001713	0.060172	0.035316	0.415

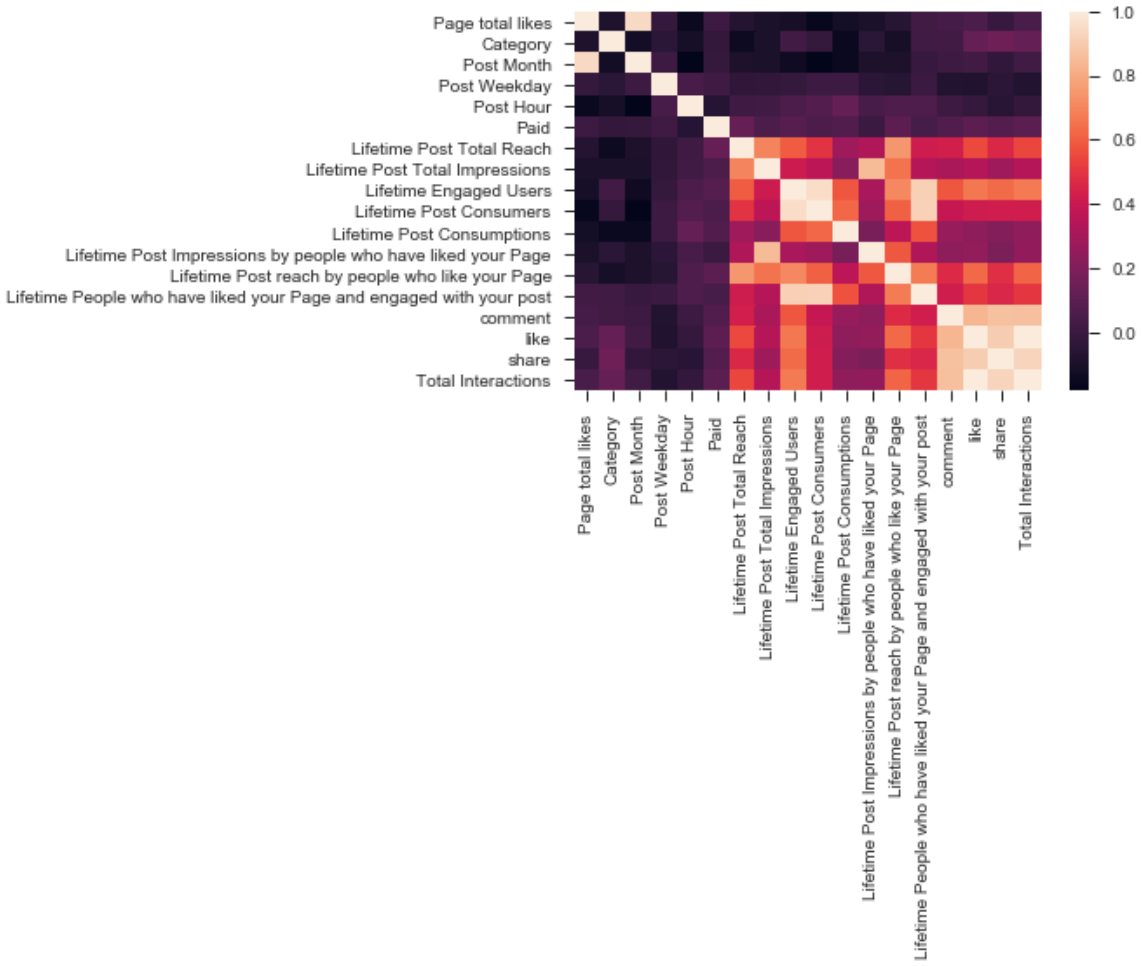
	Page total likes	Category	Post Month	Post Weekday	Post Hour	Paid	Life R
comment	0.030709	0.024747	0.003867	-0.072777	0.008160	0.065423	0.430
like	0.052932	0.124238	0.024327	-0.077848	-0.018635	0.102031	0.551
share	-0.007921	0.148989	-0.025345	-0.048092	-0.055476	0.073078	0.462
Total Interactions	0.045810	0.123657	0.018053	-0.075601	-0.021598	0.098544	0.544

In [217]:

```
sns.heatmap(data.corr())
```

Out[217]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1aabd080>
```



In [218]:

```
data=data.drop(['Page total likes','Category','Post Month','Post Weekday','Post Hour',
'Paid','Lifetime Post Total Impressions','Lifetime Post Impressions by people who have
 liked your Page','comment','like','share','Total Interactions'], axis=1)
sns.heatmap(data.corr(), annot=True, fmt='.3f')
```

Out[218]:

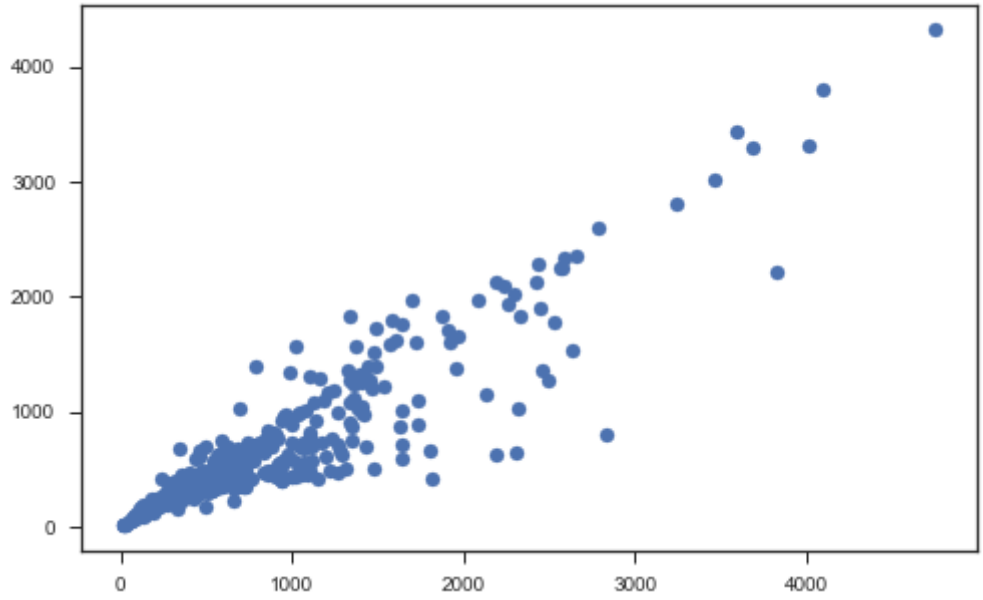
```
<matplotlib.axes._subplots.AxesSubplot at 0x359f9dd8>
```

In [219]:

```
plt.figure(figsize=(8,5))
plt.scatter(data['Lifetime Post Consumers'], data['Lifetime People who have liked your
 Page and engaged with your post'])
```

Out[219]:

```
<matplotlib.collections.PathCollection at 0x38aa2898>
```



Гистограмма

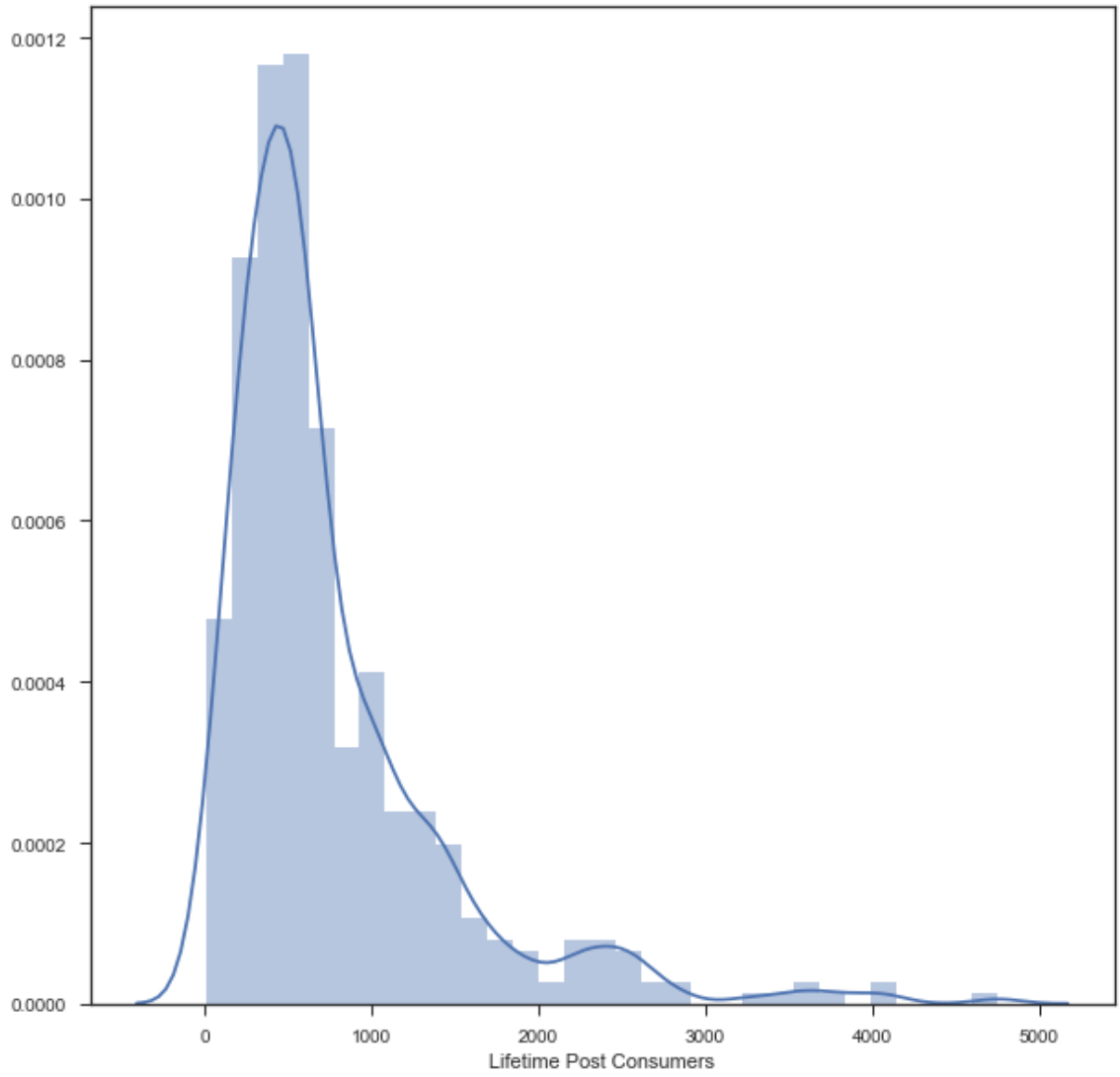
In [220]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(data['Lifetime Post Consumers'])
```

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\lib\sit
e-packages\matplotlib\axes\_axes.py:6462: UserWarning: The 'normed' kwarg
is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

Out[220]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x38aa2e80>
```



Jointplot - комбинация гистограмм и диаграмм рассеивания

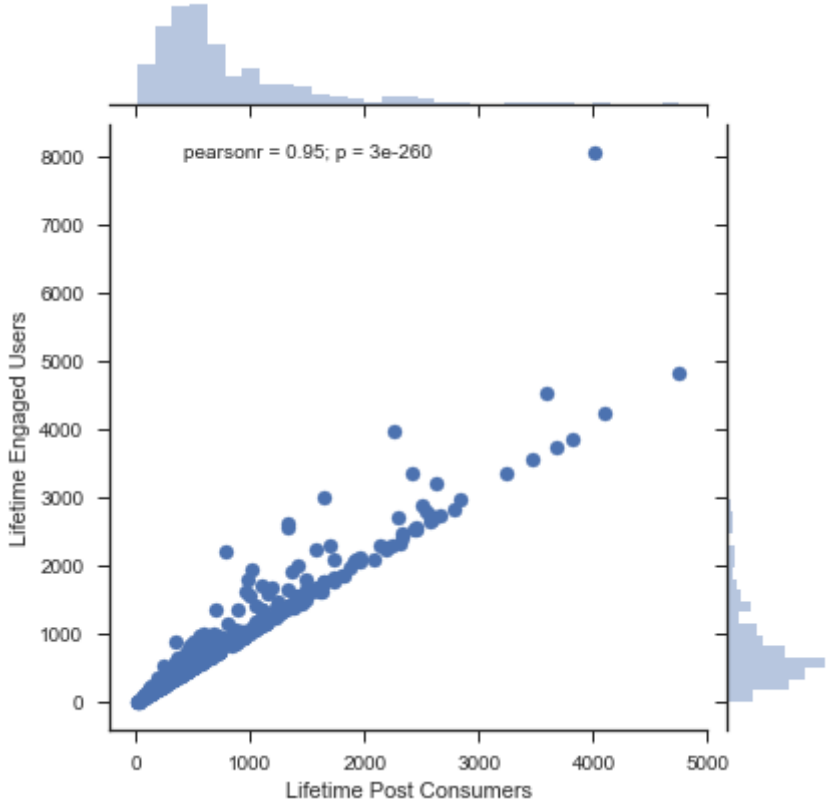
In [221]:

```
sns.jointplot(x='Lifetime Post Consumers', y='Lifetime Engaged Users', data=data)
```

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\lib\sit
e-packages\matplotlib\axes\_axes.py:6462: UserWarning: The 'normed' kwarg
is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\lib\sit
e-packages\matplotlib\axes\_axes.py:6462: UserWarning: The 'normed' kwarg
is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

Out[221]:

```
<seaborn.axisgrid.JointGrid at 0x36116908>
```



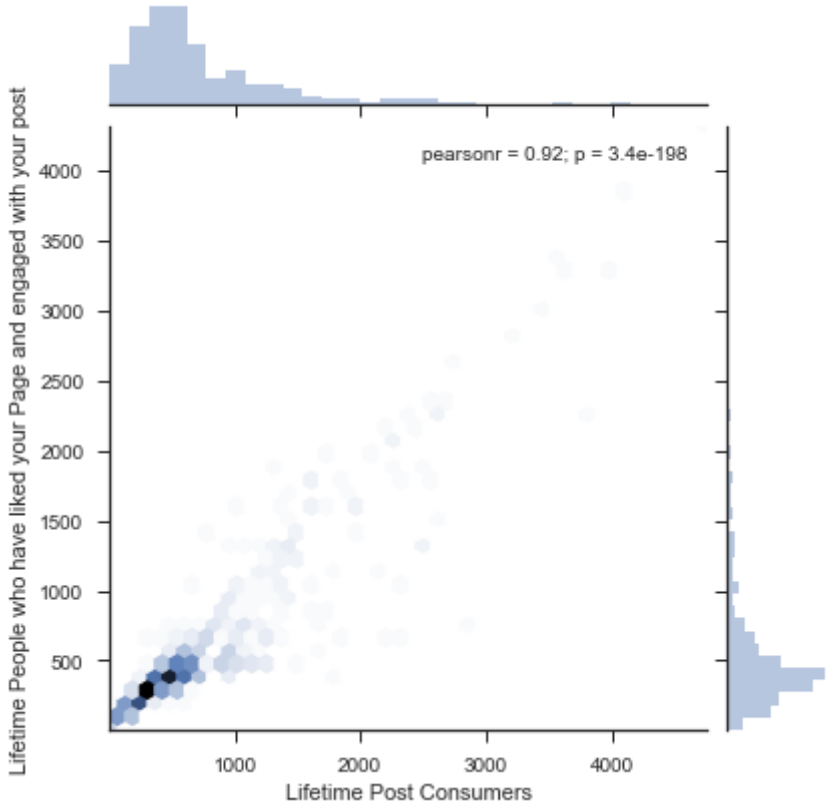
In [222]:

```
sns.jointplot(x='Lifetime Post Consumers', y='Lifetime People who have liked your Page
 and engaged with your post', data=data, kind='hex')
```

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\lib\sit
e-packages\matplotlib\axes\_axes.py:6462: UserWarning: The 'normed' kwarg
is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\lib\sit
e-packages\matplotlib\axes\_axes.py:6462: UserWarning: The 'normed' kwarg
is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

Out[222]:

```
<seaborn.axisgrid.JointGrid at 0x301777b8>
```

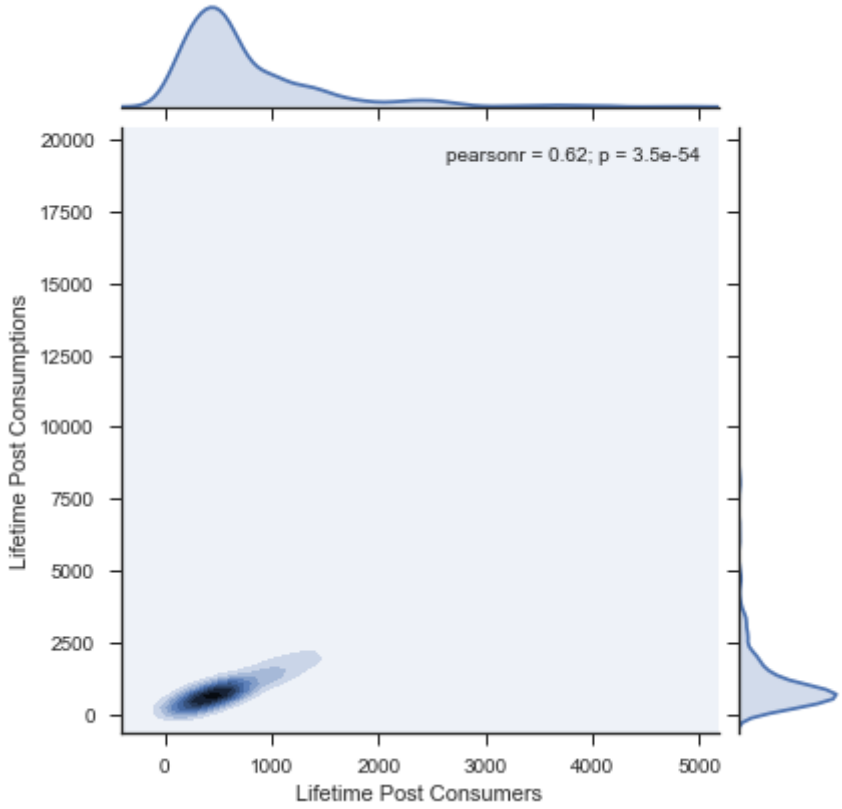


In [223]:

```
sns.jointplot(x='Lifetime Post Consumers', y='Lifetime Post Consumptions', data=data, k
ind='kde')
```

Out[223]:

```
<seaborn.axisgrid.JointGrid at 0x361207f0>
```



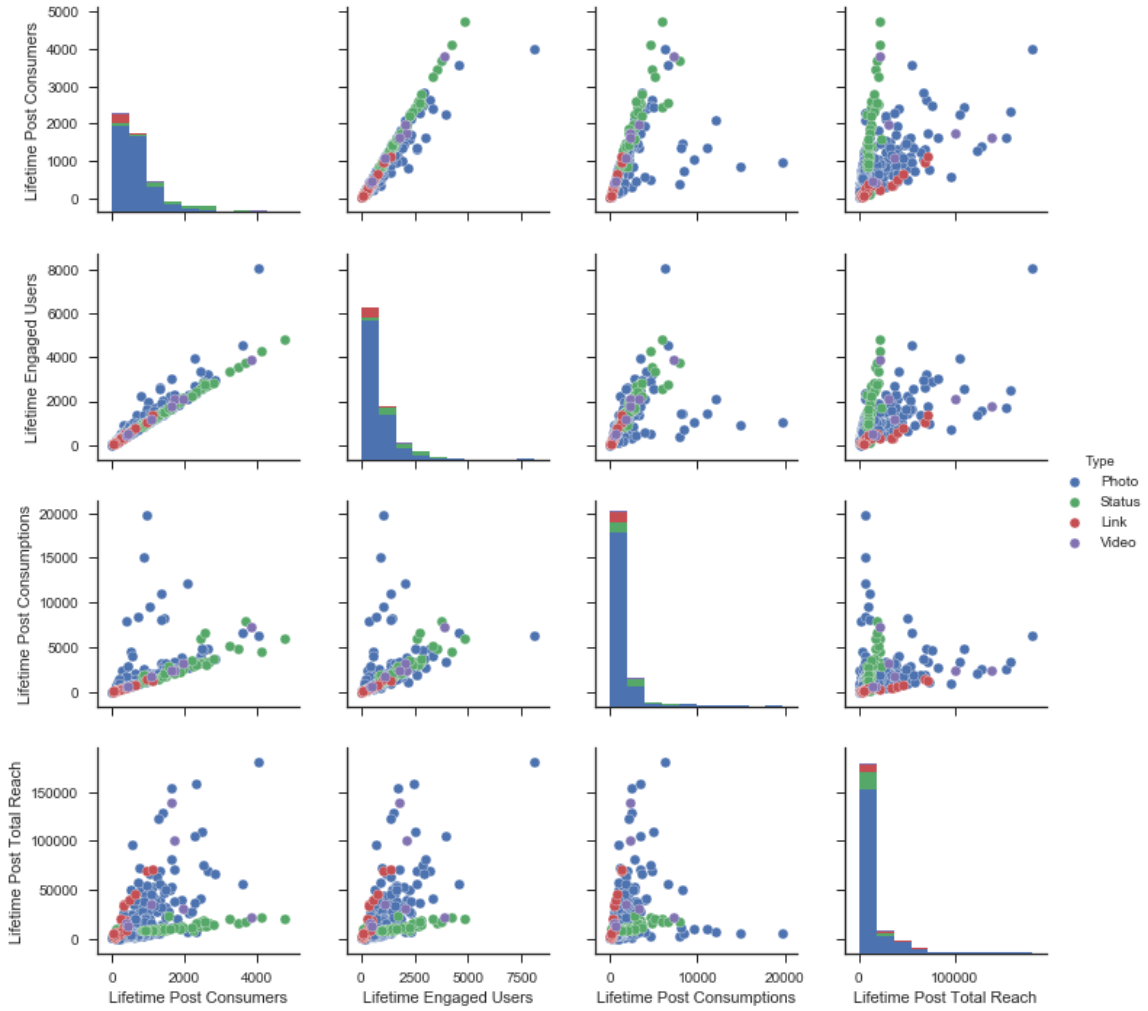
Парные диаграммы

In [224]:

```
cols = ['Lifetime Post Consumers','Lifetime Engaged Users','Lifetime Post Consumptions'
,'Lifetime Post Total Reach','Type']
sns.pairplot(data[cols], hue='Type')
```

Out[224]:

```
<seaborn.axisgrid.PairGrid at 0x1d15dba8>
```



Ящик с усами

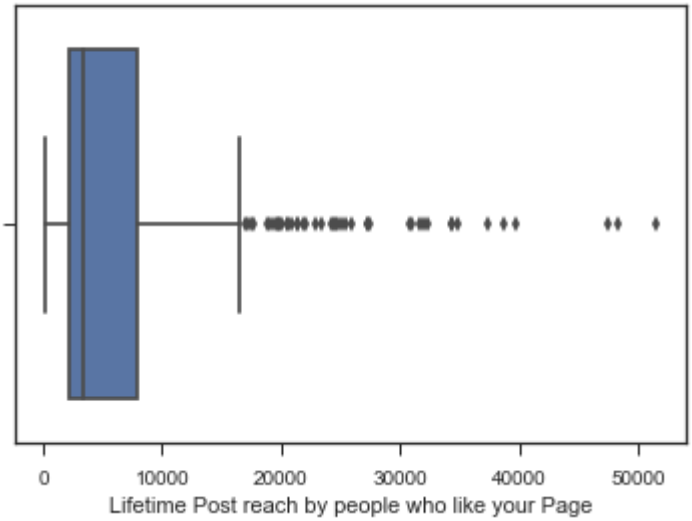
Отображает одномерное распределение вероятности

In [225]:

```
sns.boxplot(x=data['Lifetime Post reach by people who like your Page'])
```

Out[225]:

<matplotlib.axes._subplots.AxesSubplot at 0x38b18518>

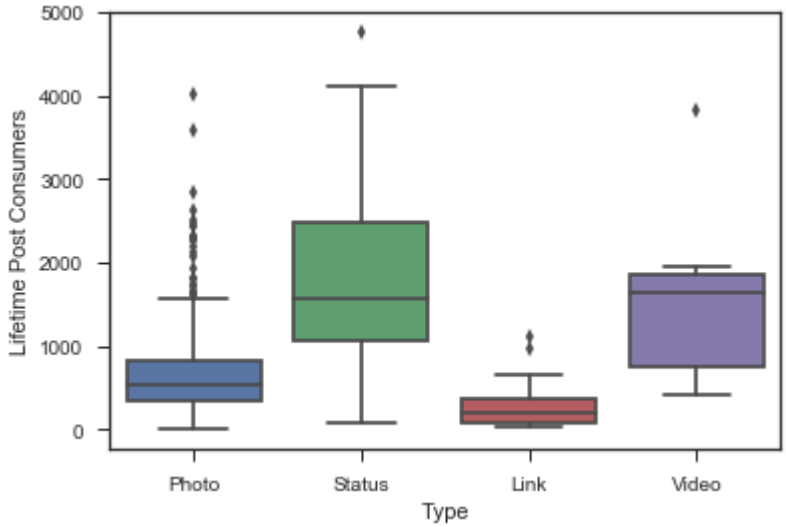


In [226]:

```
# Распределение параметра Page total likes сгруппированные по Type.
sns.boxplot(x='Type', y='Lifetime Post Consumers', data=data)
```

Out[226]:

<matplotlib.axes._subplots.AxesSubplot at 0x377a7828>

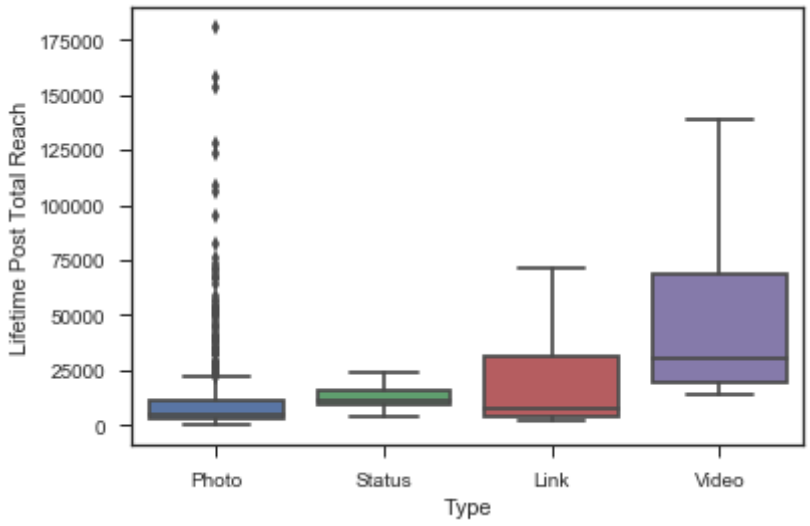


In [227]:

```
sns.boxplot(x='Type', y='Lifetime Post Total Reach', data=data)
```

Out[227]:

<matplotlib.axes._subplots.AxesSubplot at 0x37040978>



где 1-action(акция), 2-product(товар), 3-inspiration(неявная связь с брендом)

Violin plot

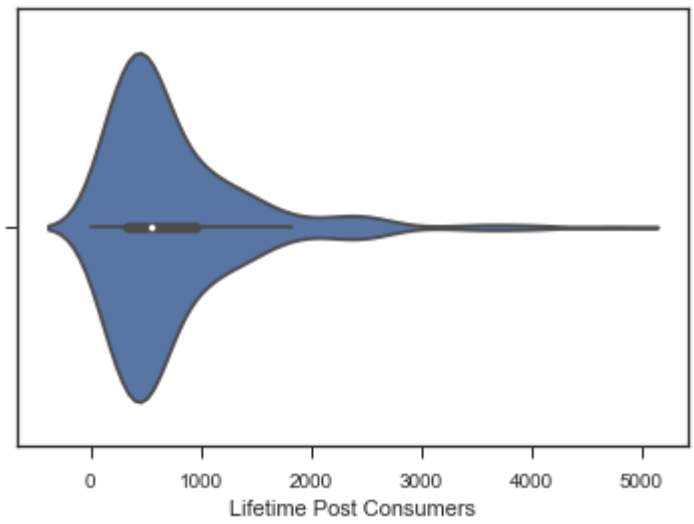
Похоже на предыдущую диаграмму, но по краям отображаются распределения плотности

In [228]:

```
sns.violinplot(x=data['Lifetime Post Consumers'])
```

Out[228]:

<matplotlib.axes._subplots.AxesSubplot at 0x385c0b70>



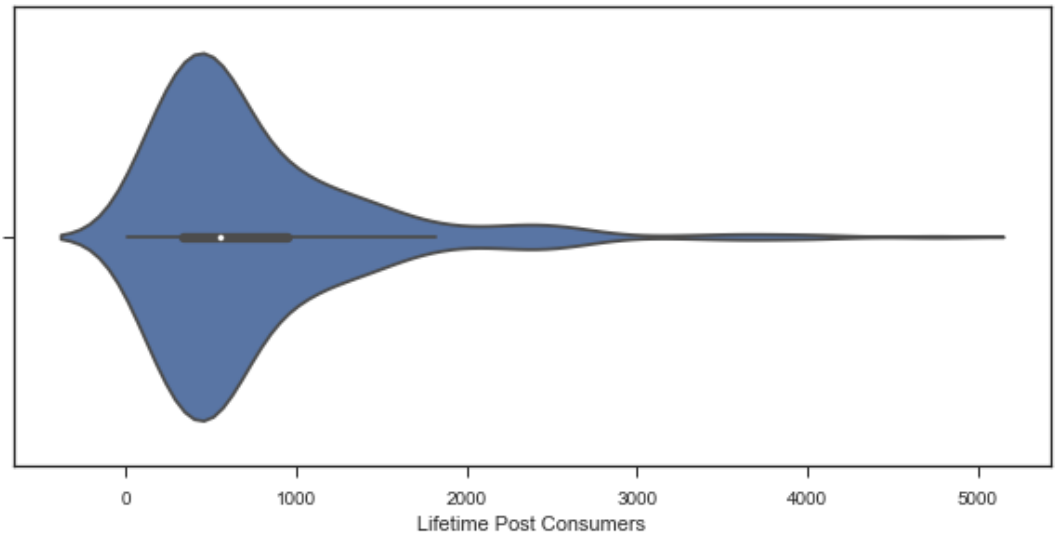
In [231]:

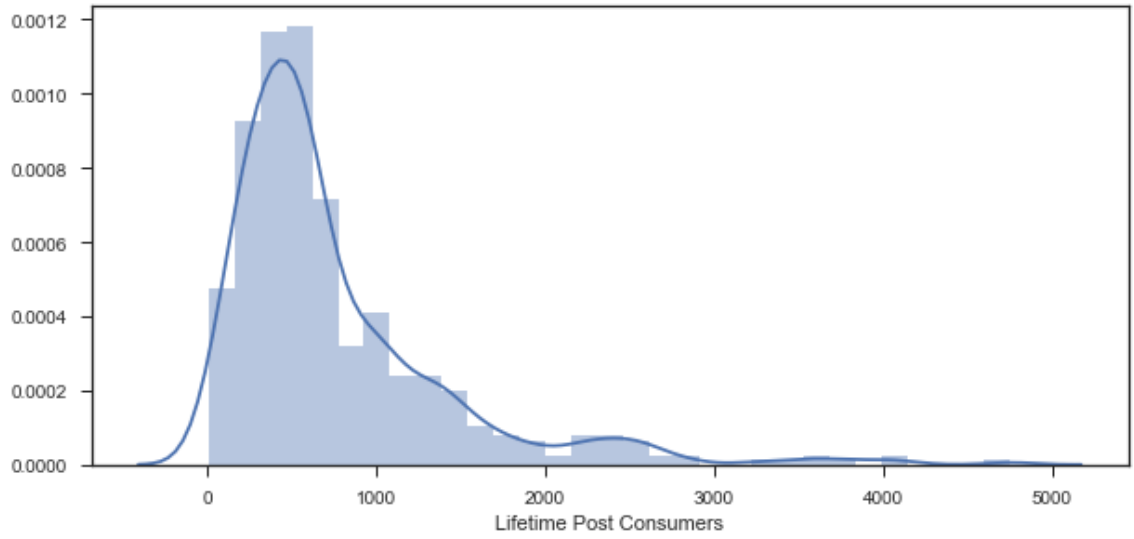
```
fig, ax = plt.subplots(2, 1, figsize=(10,10))
sns.violinplot(ax=ax[0], x=data['Lifetime Post Consumers'])
sns.distplot(data['Lifetime Post Consumers'], ax=ax[1])
```

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\lib\sit
e-packages\matplotlib\axes\_axes.py:6462: UserWarning: The 'normed' kwarg
is deprecated, and has been replaced by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

Out[231]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x21370f28>
```





In [232]:

```
# Распределение параметра Page total likes сгруппированные по Type.
sns.violinplot(x='Type', y='Lifetime Post Consumers', data=data)
```

Out[232]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d36c4e0>