LACMAPlots

October 6, 2019

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
```

[2]:
```
%%time
data = pd.read_csv('USC_Ticketing_Membership_LACMAFund_Sept2018_Sept2019.csv',␣
 ↪encoding = "ISO-8859-1")
```

```
Wall time: 7.19 s
```

[3]:
```
data.head()
```

[3]:

	order_no	customer_no	order_dt	appeal_no	Appeal	source_no	source_name	MOS	MOS.1	price_type	PriceType	ticket_no	due_amt	fee_amt	perf_no	perf_type	description	season	description.1	zone_no	description.2	recipient_no
0	1040147	2039167	9/1/2018	1	Web	1	Default Web Source	3	Web Mode of Sale	18.0	Adult	1680352.0	25.0	2.0	12799.0	3.0	General Admission	43.0	FY19TIX	529.0	LACMA 99	NaN
1	1040147	2039167	9/1/2018	1	Web	1	Default Web Source	3	Web Mode of Sale	18.0	Adult	1680353.0	25.0	2.0	12799.0	3.0	General Admission	43.0				
2	1040147	2039167	9/1/2018	1	Web	1	Default Web Source	3	Web Mode of Sale	18.0	Adult	1680354.0	25.0	2.0	12799.0	3.0	General Admission	43.0				
3	1040153	148964	9/1/2018	1	Web	1	Default Web Source	3	Web Mode of Sale	64.0	LACMA Member	1712525.0	0.0	NaN	15577.0	6.0	Films	44.0				
4	1040153	148964	9/1/2018	1	Web	1	Default Web Source	3	Web Mode of Sale	64.0	LACMA Member	1712526.0	0.0	NaN	15577.0	6.0	Films	44.0				

```
1       FY19TIX    529.0      LACMA 99           NaN
2       FY19TIX    529.0      LACMA 99           NaN
3      FY19FILM    425.0  Bing Theater           NaN
4      FY19FILM    425.0  Bing Theater           NaN
```

[4]:
```
data_members = data[data.customer_no != 0]
data_nonmembers = data[data.customer_no == 0]
```

0.0.1 Histogram of ticket sources

Members and non-members

[5]:
```
def sources_freq_frame(data, thers = 0.1):
    sources_freq = (data.source_name.value_counts() / data.shape[0] * 100).
 ↪reset_index()
    sources_freq['source_name'] = sources_freq['source_name'].apply(lambda x: x␣
 ↪if x > thers else np.nan)
    sources_freq.dropna(inplace=True)
    return sources_freq
```

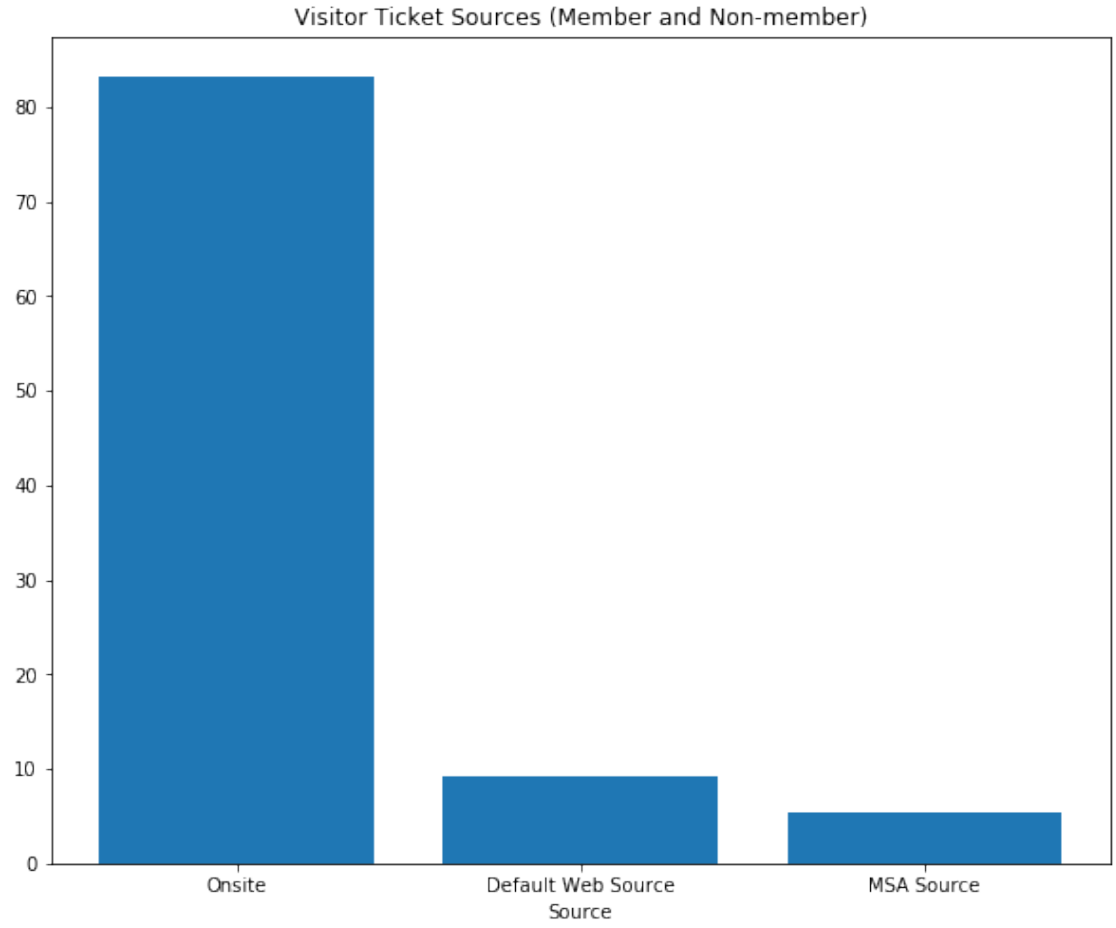
[6]:
```
sources_freq = sources_freq_frame(data, 1)
sources_freq
```

[6]:
```
                index  source_name
0             Onsite    83.239601
1  Default Web Source     9.178792
2          MSA Source     5.461703
```

[7]:
```
plt.figure(figsize=(10,8))

plt.bar(x = 'index', height = 'source_name', data = sources_freq)


plt.title('Visitor Ticket Sources (Member and Non-member)', fontweight = 14)
plt.xlabel('Source', fontweight = 12)
plt.show()
```



Member

[8]:
```
sources_freq_mem = sources_freq_frame(data_members, 1)
sources_freq_mem
```

[8]:

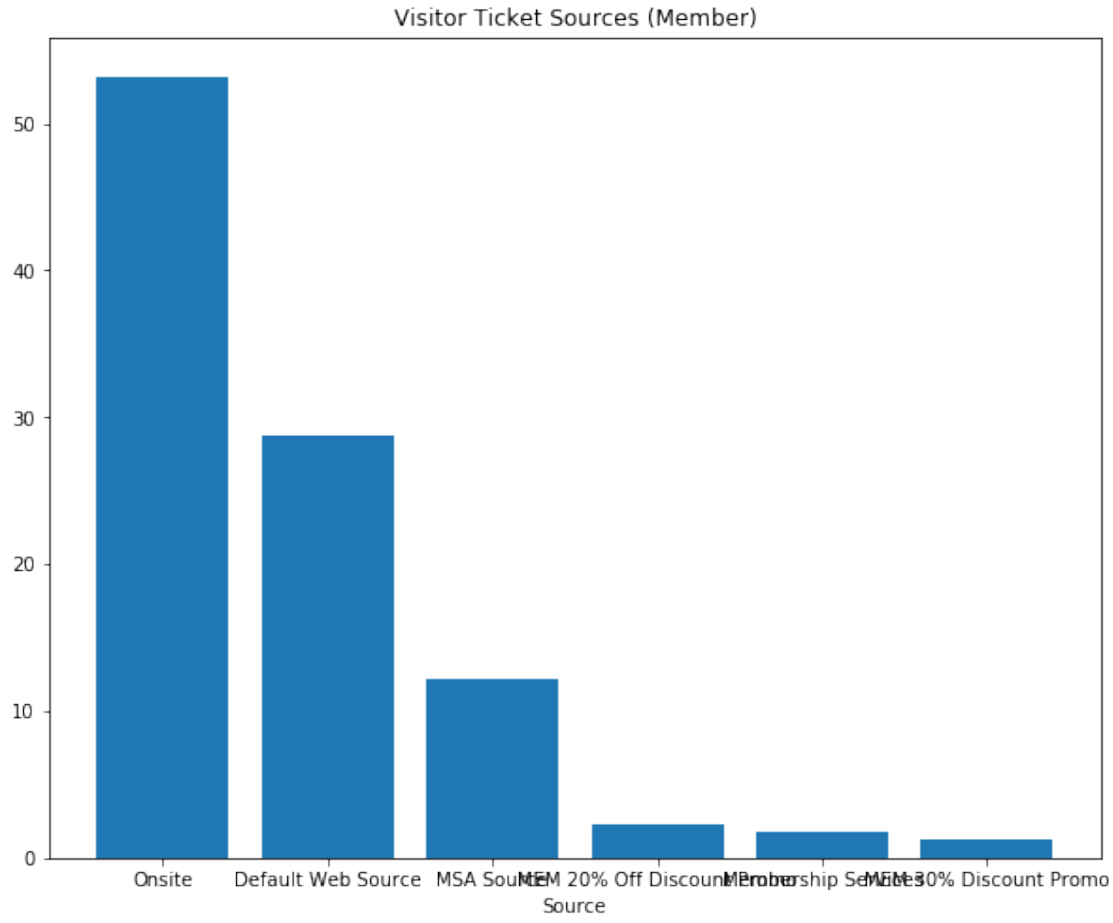
	index	source_name
0	Onsite	53.196754
1	Default Web Source	28.787732
2	MSA Source	12.173521
3	MEM 20% Off Discount Promo	2.233660
4	Membership Services	1.807071
5	MEM 30% Discount Promo	1.267662

[9]:
```
plt.figure(figsize=(10,8))

plt.bar(x = 'index', height = 'source_name', data = sources_freq_mem)


plt.title('Visitor Ticket Sources (Member)', fontweight = 14)
```

```
plt.xlabel('Source', fontweight = 12)
plt.show()
```



Non-member

[10]:
```
sources_freq_nonmem = sources_freq_frame(data_nonmembers, 1)
sources_freq_nonmem
```

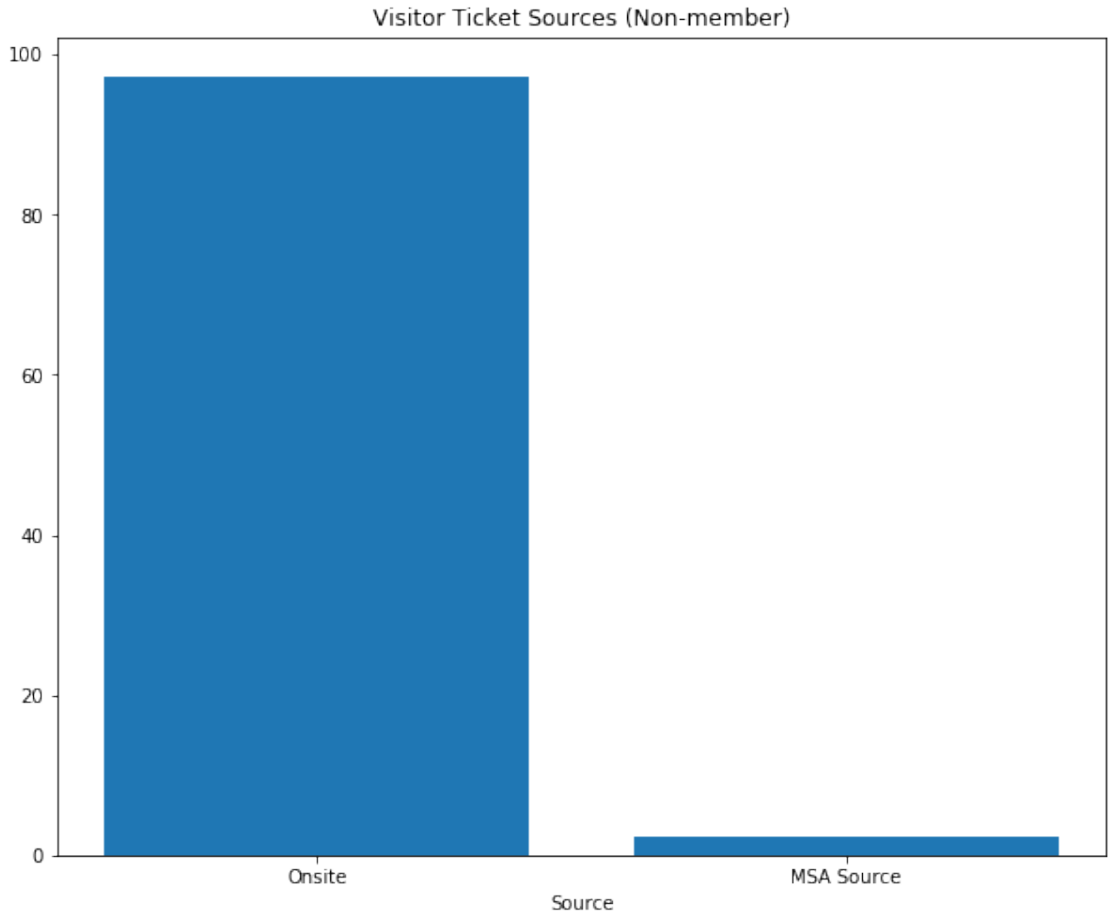
[10]:
```
         index  source_name
0      Onsite    97.301034
1  MSA Source     2.320263
```

[11]:
```
plt.figure(figsize=(10,8))

plt.bar(x = 'index', height = 'source_name', data = sources_freq_nonmem)


plt.title('Visitor Ticket Sources (Non-member)', fontweight = 14)
plt.xlabel('Source', fontweight = 12)
```

```
plt.show()
```



0.0.2 Description histogram

Members and non-members

[12]:
```
def desc_freq_frame(data, thers = 0.1):
    desc_freq = (data.description.value_counts()/ data.shape[0] * 100).
 ↪reset_index()
    desc_freq['description'] = desc_freq['description'].apply(lambda x: x if x >␣
 ↪thers else np.nan)
    desc_freq.dropna(inplace = True)
    return desc_freq
```

[13]:
```
desc_freq = desc_freq_frame(data)
desc_freq
```

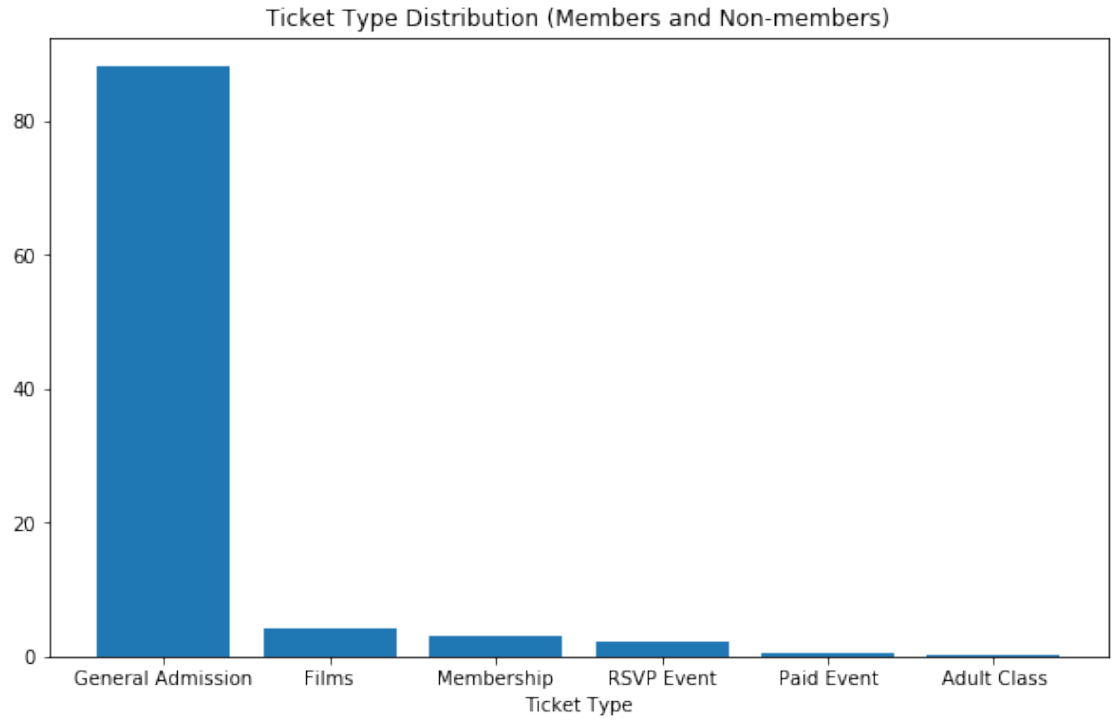
[13]:
```
                index  description
0  General Admission    88.151722
1              Films     4.247528
```

```
2          Membership     3.148518
3          RSVP Event     2.264865
4          Paid Event     0.592344
5         Adult Class     0.101271
```

[14]:
```
plt.figure(figsize=(10,6))

plt.bar(x = 'index', height = 'description', data = desc_freq)


plt.title('Ticket Type Distribution (Members and Non-members)', fontweight = 14)
plt.xlabel('Ticket Type', fontweight = 12)
plt.show()
```



0.0.3 Members

[15]:
```
desc_freq_mem = desc_freq_frame(data_members)
desc_freq_mem
```

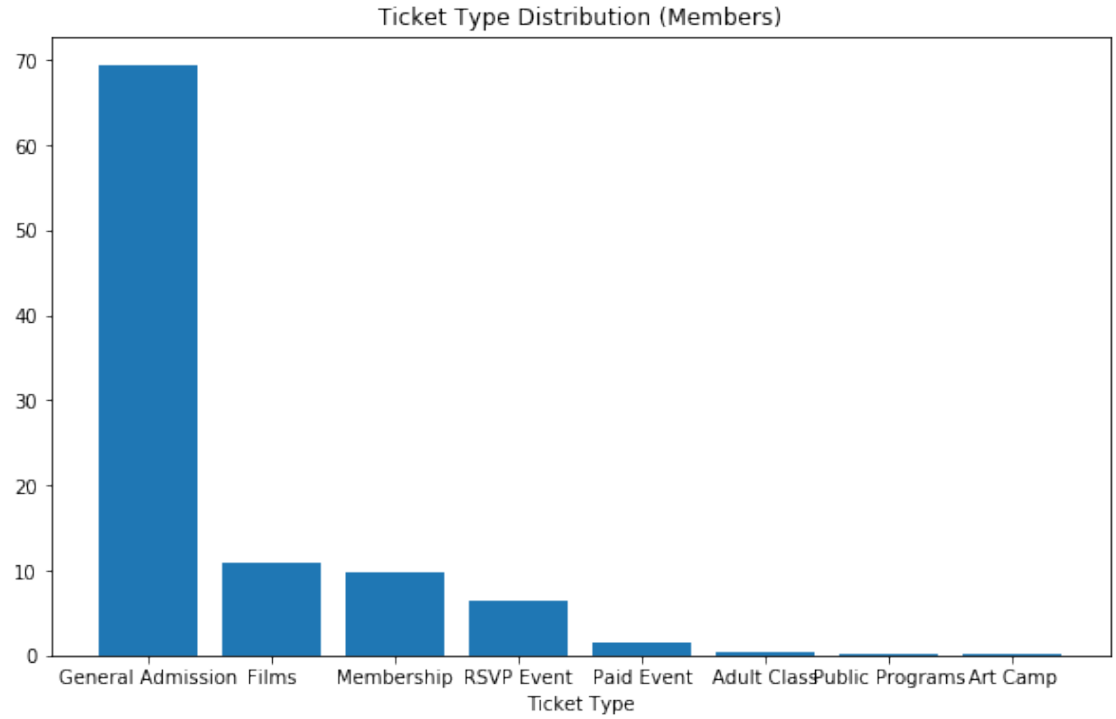
[15]:
```
                index  description
0  General Admission    69.338376
1              Films    10.902470
2         Membership     9.820745
3         RSVP Event     6.421108
```

```
4         Paid Event     1.613872
5       Adult Class      0.317642
6    Public Programs     0.148652
7          Art Camp      0.124926
```

[16]:
```
plt.figure(figsize=(10,6))

plt.bar(x = 'index', height = 'description', data = desc_freq_mem)


plt.title('Ticket Type Distribution (Members)', fontweight = 14)
plt.xlabel('Ticket Type', fontweight = 12)
plt.show()
```



Non-Members

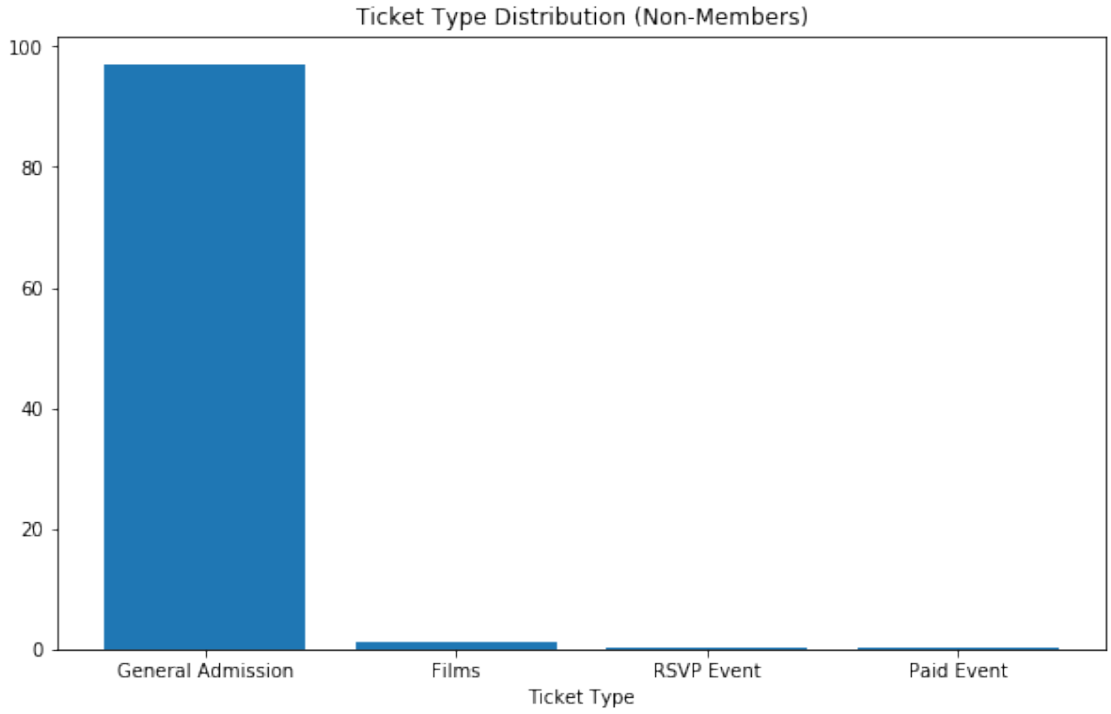
[17]:
```
desc_freq_nonmem = desc_freq_frame(data_nonmembers)
desc_freq_nonmem
```

[17]:
```
                index  description
0  General Admission    96.957232
1              Films     1.132709
2         RSVP Event     0.319552
3         Paid Event     0.114223
```

[18]:
```
plt.figure(figsize=(10,6))

plt.bar(x = 'index', height = 'description', data = desc_freq_nonmem)


plt.title('Ticket Type Distribution (Non-Members)', fontweight = 14)
plt.xlabel('Ticket Type', fontweight = 12)
plt.show()
```



0.0.4 Price Type Distribution

[19]:
```
pricetype_mask = ['LACMA Member', 'LACMA Member Guest', 'Adult', 'NexGenLA Adult␣
↪Guest', 'NexGenLA Youth Member', 'Senior (65+)',
   'Student (with ID)', 'Teen (13-17)', 'Youth (0-12)']
```

[20]:
```
def pricetype_freq_frame(data):
    pricetype_freq = (data.PriceType.value_counts() / data.shape[0] * 100).
↪loc[pricetype_mask].reset_index()
    pricetype_freq.dropna(inplace=True)
    pricetype_freq.sort_values('PriceType', ascending=False, inplace=True)
    return pricetype_freq
```

[21]:
```
pricetype_freq = pricetype_freq_frame(data)
pricetype_freq
```

```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:2:
FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-
reindex-listlike
```

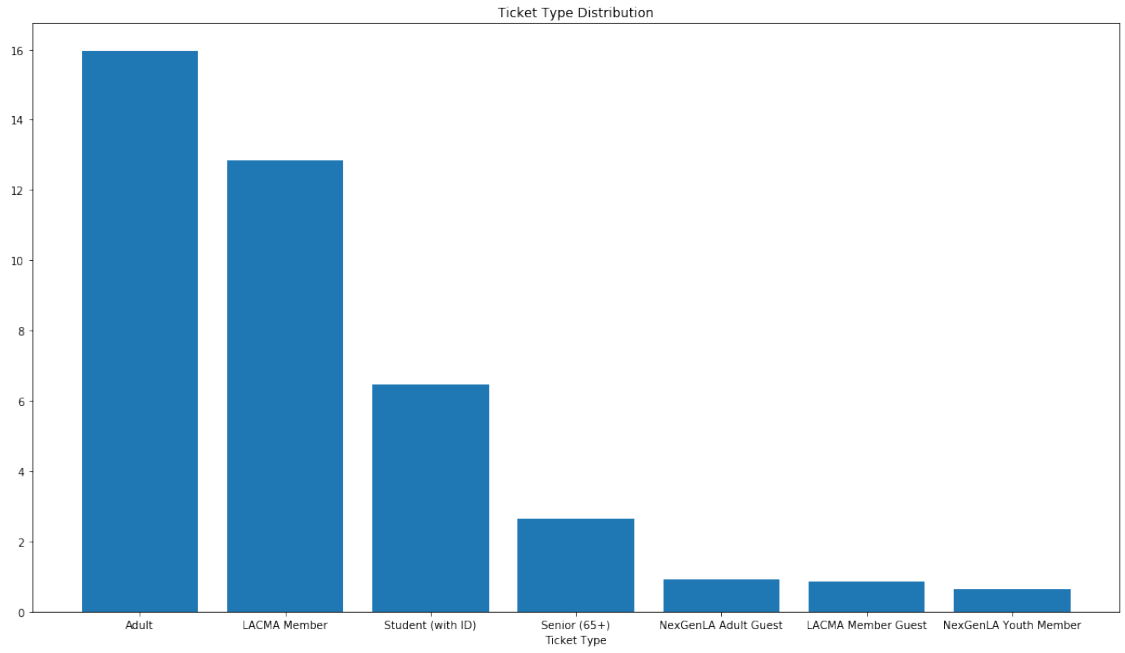
[21]:

	index	PriceType
2	Adult	15.953625
0	LACMA Member	12.856514
6	Student (with ID)	6.453729
5	Senior (65+)	2.633671
3	NexGenLA Adult Guest	0.922866
1	LACMA Member Guest	0.859417
4	NexGenLA Youth Member	0.638349

[22]:
```
plt.figure(figsize=(18,10))

plt.bar(x = 'index', height = 'PriceType', data = pricetype_freq)


plt.title('Ticket Type Distribution', fontweight = 14)
plt.xlabel('Ticket Type', fontweight = 12)
plt.show()
```



Online tickets

[23]:
```
data_web = data[data.source_name == 'Default Web Source']
```

[24]:
```
pricetype_freq_web = pricetype_freq_frame(data_web)
pricetype_freq_web
```

```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:2:
FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-
reindex-listlike
```

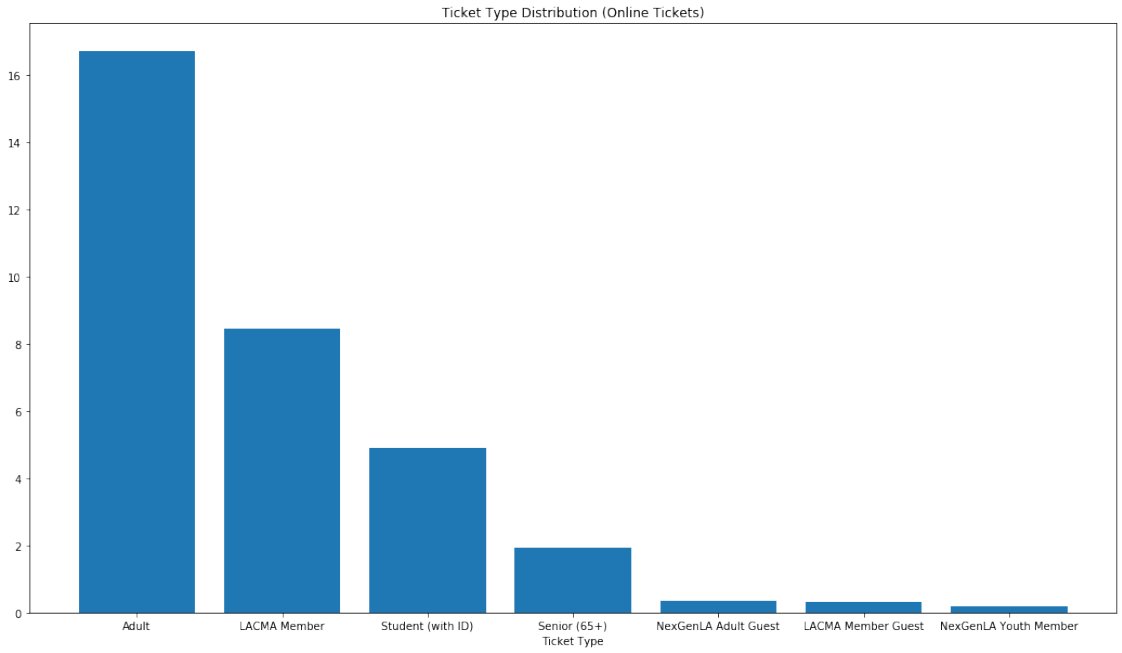
[24]:

	index	PriceType
2	Adult	16.709555
0	LACMA Member	8.456532
6	Student (with ID)	4.887566
5	Senior (65+)	1.932489
3	NexGenLA Adult Guest	0.359924
1	LACMA Member Guest	0.301058
4	NexGenLA Youth Member	0.181644

[25]:
```
plt.figure(figsize=(18,10))

plt.bar(x = 'index', height = 'PriceType', data = pricetype_freq_web)


plt.title('Ticket Type Distribution (Online Tickets)', fontweight = 14)
plt.xlabel('Ticket Type', fontweight = 12)
plt.show()
```



Onsite tickets

[26]:
```
data_onsite = data[data.source_name == 'Onsite']
```

[27]:
```
pricetype_freq_onsite = pricetype_freq_frame(data_onsite)
pricetype_freq_onsite
```

```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:2:
FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-
reindex-listlike
```

[27]:

	index	PriceType
2	Adult	16.917935
0	LACMA Member	10.694236
6	Student (with ID)	7.086278
5	Senior (65+)	2.882249
3	NexGenLA Adult Guest	0.962913
1	LACMA Member Guest	0.712541
4	NexGenLA Youth Member	0.617400

[28]:
```
plt.figure(figsize=(18,10))

plt.bar(x = 'index', height = 'PriceType', data = pricetype_freq_onsite)


plt.title('Ticket Type Distribution (Onsite Tickets)', fontweight = 14)
plt.xlabel('Ticket Type', fontweight = 12)
plt.show()
```