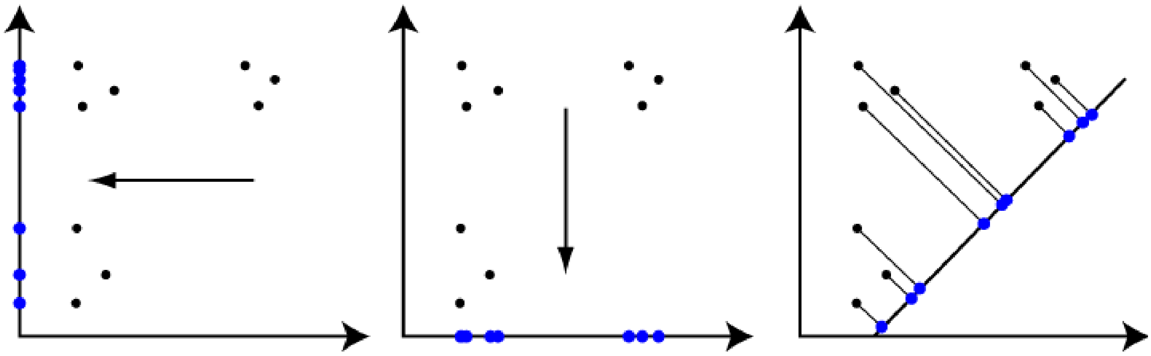
เมื่อเรามีตัวแปรมากกว่า 2 ตัว เราสามารถใช้ PCA (Principle Component Analysis) ในการลดจำนวนตัวแปรลงได้

Projection



https://wendynavarrete.com/principal-component-analysis-with-numpy/

load data

In [ ]:
```
import pandas as pd
```

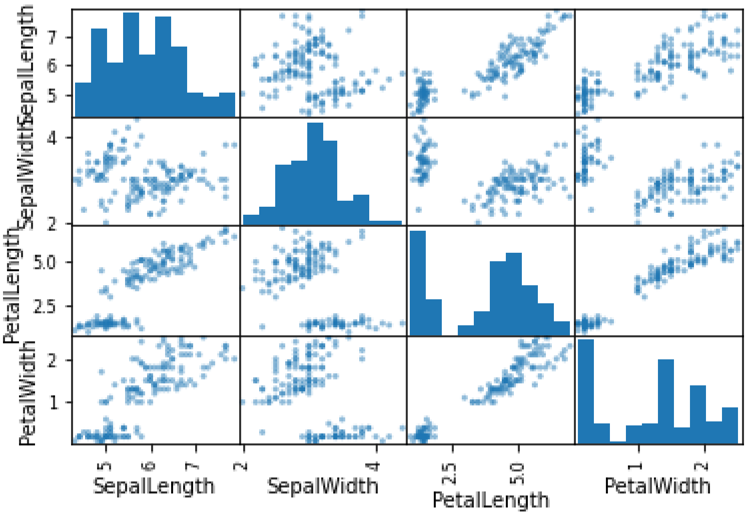
In [ ]:
```
example_df = pd.read_csv('https://raw.github.com/pandas-dev/pandas/master/pandas/te
example_df
```

Out[ ]:

	SepalLength	SepalWidth	PetalLength	PetalWidth	Name
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

In [ ]:
```
pd.plotting.scatter_matrix(example_df);
```



PCA

sklearn -> scikit-learn เป็น package ที่รวบรวม function การทำ Data Science - Machine Learning - Data Mining เอาไว้ให้เราใช้แบบไม่ต้องเขียนเอง

Import

In [ ]:
```
from sklearn.decomposition import PCA
```

Define

In [ ]:
```
pca = PCA()
```

Fit - Transform

In [ ]:
```
example_df.iloc[:,:-1].shape
```

Out[ ]:
```
(150, 4)
```

In [ ]:
```
new_pca = pca.fit_transform(example_df.iloc[:,:-1])  ## record - แถว  , dimension -
```

In [ ]:
```
new_pca.shape
```

Out[ ]:
```
(150, 4)
```

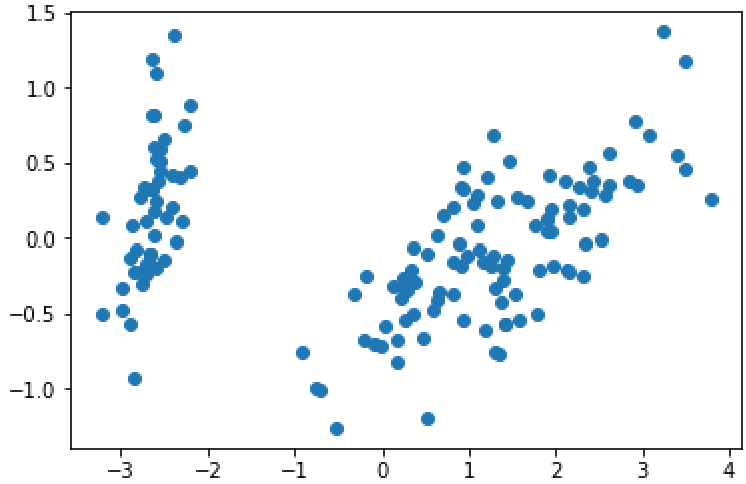
In [ ]:
```
pca.explained_variance_ratio_
```

Out[ ]:
```
array([0.92461621, 0.05301557, 0.01718514, 0.00518309])
```

In [ ]:
```
from matplotlib import pyplot as plt
```

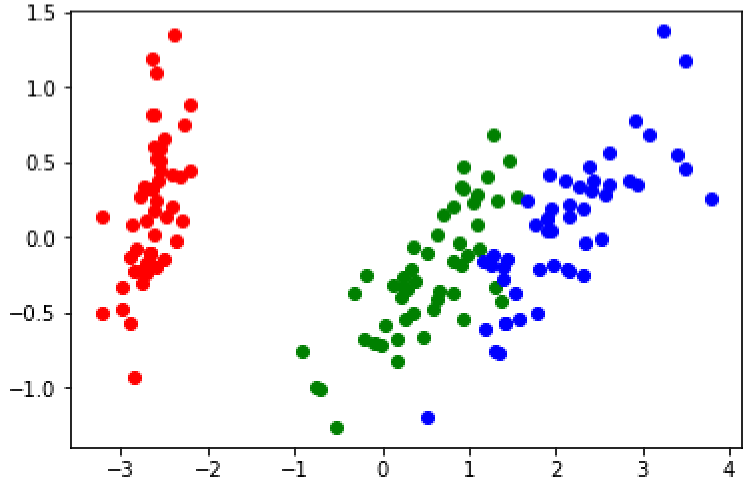
In [ ]:
```
plt.scatter(new_pca[:,0],new_pca[:,1])
```

Out[ ]:
```
<matplotlib.collections.PathCollection at 0x7f671c75fd90>
```



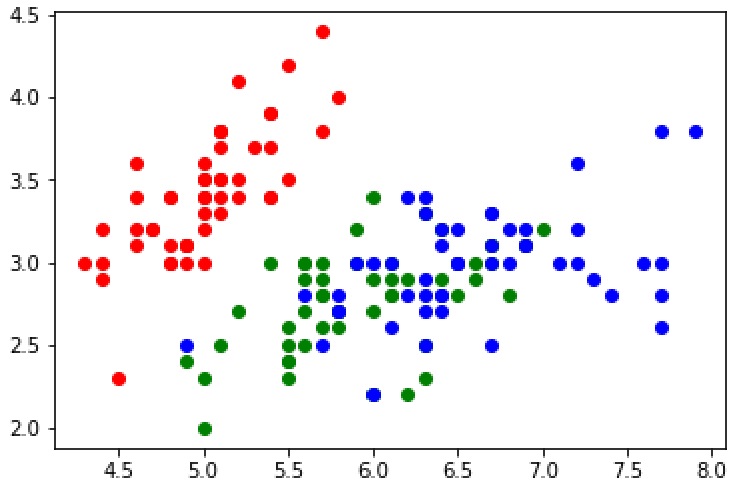
```
In [ ]: plt.plot(new_pca[:50,0],new_pca[:50,1],'ro')
        plt.plot(new_pca[50:100,0],new_pca[50:100,1],'go')
        plt.plot(new_pca[100:,0],new_pca[100:,1],'bo')
```

```
Out[ ]: [<matplotlib.lines.Line2D at 0x7f671c370350>]
```



```
In [ ]: plt.plot(example_df.iloc[:50,0],example_df.iloc[:50,1],'ro')
        plt.plot(example_df.iloc[50:100,0],example_df.iloc[50:100,1],'go')
        plt.plot(example_df.iloc[100:,0],example_df.iloc[100:,1],'bo')
```

```
Out[ ]: [<matplotlib.lines.Line2D at 0x7f671c18ab50>]
```



```
In [ ]:
```

สอน 1 เมษา 2564

In [ ]:
```
import pandas as pd
import os
from datetime import datetime as dt
from datetime import time
```

In [ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call driv
e.mount("/content/drive", force_remount=True).
```

In [ ]:
```
path = '/content/drive/My Drive/dataviz_2021_data'
```

In [ ]:
```
data = pd.read_csv(os.path.join(path,'search_request.csv'))
data.head()
```

```
/usr/local/lib/python3.7/dist-packages/IPython/core/interactiveshell.py:2718: Dtyp
eWarning: Columns (7,8,9) have mixed types.Specify dtype option on import or set l
ow_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

Out[ ]:

	Unnamed: 0	search_id	search_timestamp	user_agent	q	user_id	s
0	0	683de889-f923-494e-9d46-44a3d67b7259	2018-06-14 12:34:35.449	Wongnai/8.17.3 rv:8.17.3.3921 (iPhone5,4; iOS;...	NaN	NaN	5lqjjikta19d296mo7
1	1	4a811230-ffa4-4631-a4c8-5d0394137d02	2018-06-14 17:11:19.469	Mozilla/5.0 (iPhone; CPU iPhone OS 11_4 like M...	NaN	NaN	1r3iotmp0o9slom9
2	2	7ad6ee8e-438e-4bea-9183-74dcef9e358e	2018-06-14 13:22:31.736	Mozilla/5.0 (Linux; Android 7.0; SAMSUNG SM-J7...	NaN	NaN	5ci1eo4v5u9dha4ppg
3	3	0c17a5f5-fa89-40f4-ae94-a8659268f827	2018-06-02 12:37:27.331	Mozilla/5.0 (Linux; Android 7.1.1; SM-N950F Bu...	NaN	NaN	39n535qgje9kpojp0
4	4	6870dc3a-5602-44fc-80ed-df0a7783df9d	2018-06-02 11:19:22.404	Mozilla/5.0 (iPhone; CPU iPhone OS 11_3_1 like...	NaN	NaN	5pa03h6lj691to60

เตรียมข้อมูล

แปลงข้อมูลบอกเวลาให้เป็นตัวแปรชนิด datetime

In [ ]:
```
data['search_timestamp']  = pd.to_datetime(data['search_timestamp'],format='%Y-%m-%
```

Bar chart (กราฟแท่ง)

(กราฟผลไม้)

สร้างกราฟแท่งเปรียบเทียบปริมาณ คนเข้าใช้ web Wongnai.com เพื่อค้นหาร้านอาหาร ในแต่ละวัน

quiz 6

In [ ]:

```
data[data['search_timestamp'].dt.dayofweek == 0].shape[0]
```

Out[ ]:

```
1076297
```

In [ ]:

```
from matplotlib import pyplot as plt
```

ส่วนประกอบของกราฟแท่ง

- ตัวกราฟแท่ง (height)
- ตำแหน่งกราฟแท่ง (x)
- ชื่อแท่ง (tick_label)
- ชื่อกราฟ (plt.title)
- ชื่อแกน x (plt.xlabel)
- ชื่อแกน y (plt.ylabel)

In [ ]:

```
import matplotlib
matplotlib.__version__
```

Out[ ]:

```
'3.2.2'
```

การแสดงตัวอักษรภาษาไทยในกราฟ matplotlib

https://medium.com/@kanyawee.work/%E0%B9%81%E0%B8%AA%E0%B8%94%E0%B8%87%E0 matplotlib-%E0%B8%9A%E0%B8%99-google-colab-37210d9a9f31

https://colab.research.google.com/drive/1sTdTZx_Cm51mc8OL_QHtehWyO4725sGl#scrollTo=Ak

In [ ]:

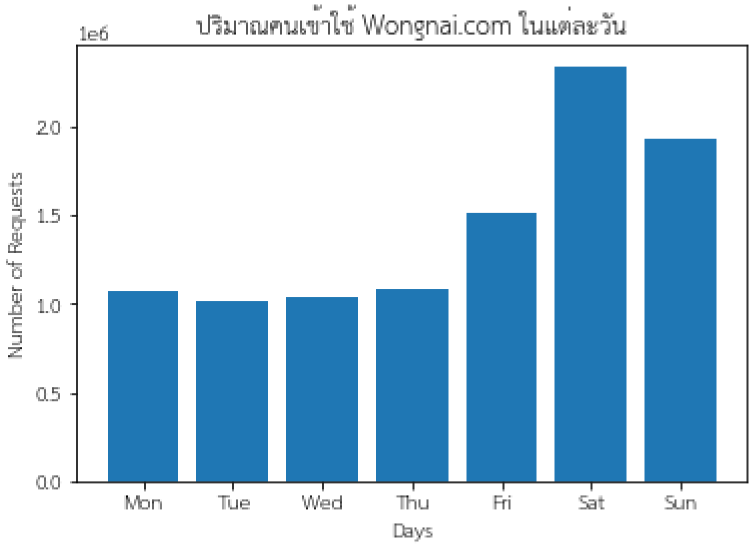
```
!wget -q https://github.com/Phonbopit/sarabun-webfont/raw/master/fonts/thsarabunnew
```

In [ ]:

```
import matplotlib as mpl
mpl.font_manager.fontManager.addfont('thsarabunnew-webfont.ttf')
mpl.rc('font', family='TH Sarabun New')
```

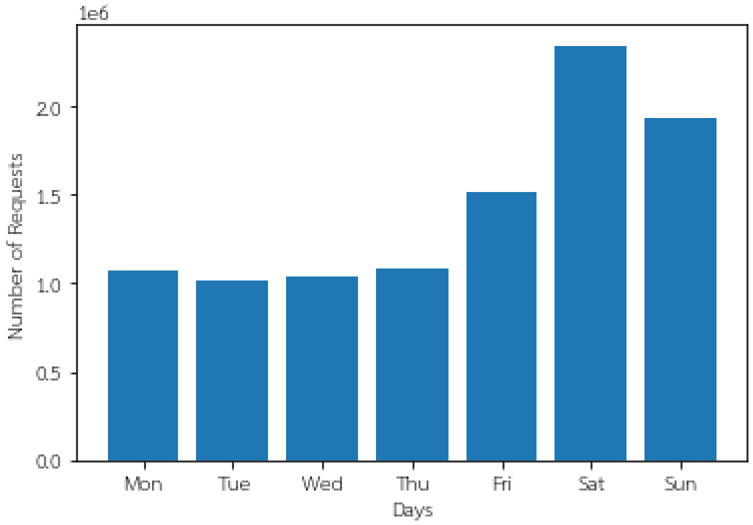
In [ ]:

```
plt.bar([1,2,3,4,5,6,7],[data[data['search_timestamp'].dt.dayofweek == 0].shape[0],
                         data[data['search_timestamp'].dt.dayofweek == 1].shape[0],
                         data[data['search_timestamp'].dt.dayofweek == 2].shape[0],
                         data[data['search_timestamp'].dt.dayofweek == 3].shape[0],
                         data[data['search_timestamp'].dt.dayofweek == 4].shape[0],
                         data[data['search_timestamp'].dt.dayofweek == 5].shape[0],
                         data[data['search_timestamp'].dt.dayofweek == 6].shape[0]
                         ],tick_label=['Mon','Tue','Wed','Thu','Fri','Sat','Sun'])
plt.xlabel('Days')
plt.ylabel('Number of Requests')
plt.title('ปริมาณคนเข้าใช้ Wongnai.com ในแต่ละวัน');
```



```
In [ ]: plt.bar([1,2,3,4,5,6,7],[data[data['search_timestamp'].dt.dayofweek == 0].shape[0],
                                 data[data['search_timestamp'].dt.dayofweek == 1].shape[0],
                                 data[data['search_timestamp'].dt.dayofweek == 2].shape[0],
                                 data[data['search_timestamp'].dt.dayofweek == 3].shape[0],
                                 data[data['search_timestamp'].dt.dayofweek == 4].shape[0],
                                 data[data['search_timestamp'].dt.dayofweek == 5].shape[0],
                                 data[data['search_timestamp'].dt.dayofweek == 6].shape[0]
                                 ],tick_label=['Mon','Tue','Wed','Thu','Fri','Sat','Sun'])
        plt.xlabel('Days')
        plt.ylabel('Number of Requests')
```

```
Out[ ]: Text(0, 0.5, 'Number of Requests')
```



Grouped bar chart

https://matplotlib.org/stable/gallery/lines_bars_and_markers/barchart.html#sphx-glr-gallery-lines-bars-and-markers-barchart-py

แสดงปริมาณคนเข้าเว๊ปในแต่ละวัน โดยเปรียบเทียบช่วงเวลา 11:00-12:00 กับ 23:00-24:00

```
In [ ]: data[(data['search_timestamp'].dt.dayofweek == 0)
        & (data['search_timestamp'].dt.time < time(hour=12))
        & (data['search_timestamp'].dt.time >= time(hour=11))].shape[0]   # monday 11:00-12:
```

```
Out[ ]: 73249
```

In [ ]:

```
b4lunch = [data[(data['search_timestamp'].dt.dayofweek == 0)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 1)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 2)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 3)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 4)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 5)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 6)&(data['search_timestam
                         ]
```

In [ ]:

```
b4lunch
```

Out[ ]:

```
[73249, 73083, 75429, 78024, 99007, 174165, 165440]
```

In [ ]:

```
data[(data['search_timestamp'].dt.dayofweek == 0)&(data['search_timestamp'].dt.time
```

Out[ ]:

```
31874
```

In [ ]:

```
b4midnight = [data[(data['search_timestamp'].dt.dayofweek == 0)&(data['search_times
           data[(data['search_timestamp'].dt.dayofweek == 1)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 2)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 3)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 4)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 5)&(data['search_timestam
           data[(data['search_timestamp'].dt.dayofweek == 6)&(data['search_timestam
                         ]
b4midnight
```

Out[ ]:

```
[31874, 32258, 31153, 35944, 53174, 58306, 35801]
```

In [ ]:

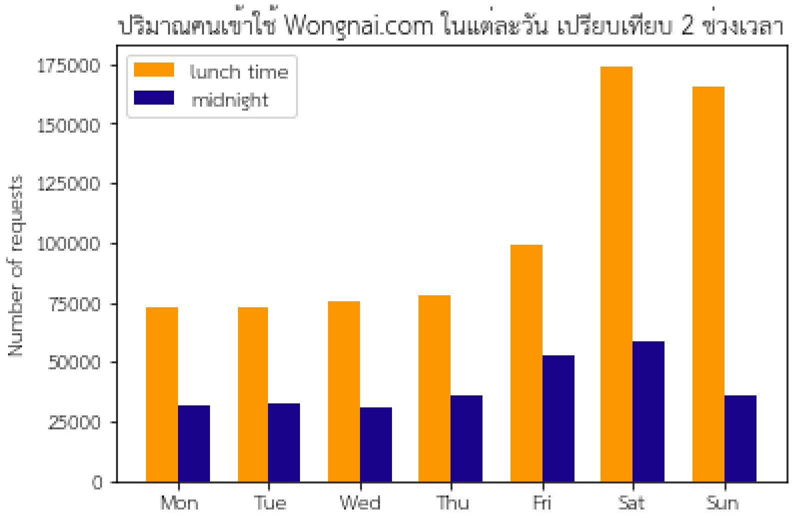
```
labels = ['Mon','Tue','Wed','Thu','Fri','Sat','Sun']
```

In [ ]:

```
import numpy as np
```

In [ ]:

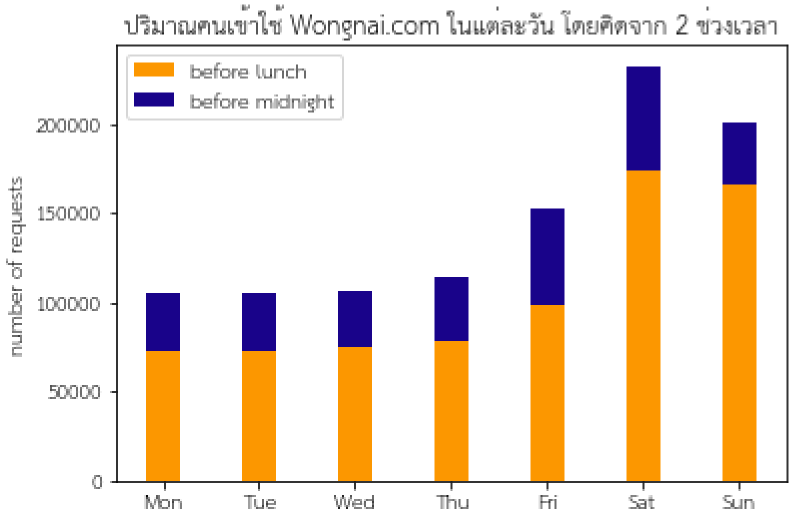
```
x = np.arange(len(labels))  # the label locations
width = 0.35  # the width of the bars

fig, ax = plt.subplots()
rects1 = ax.bar(x - width/2, b4lunch, width, label='lunch time',color = '#fc9700')
rects2 = ax.bar(x + width/2, b4midnight, width, label='midnight',color = '#19038a')

# Add some text for labels, title and custom x-axis tick labels, etc.
ax.set_ylabel('Number of requests')
ax.set_title('ปริมาณคนเข้าใช้ Wongnai.com ในแต่ละวัน เปรียบเทียบ 2 ช่วงเวลา')
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend();
```



Stacked bar chart

In [ ]:

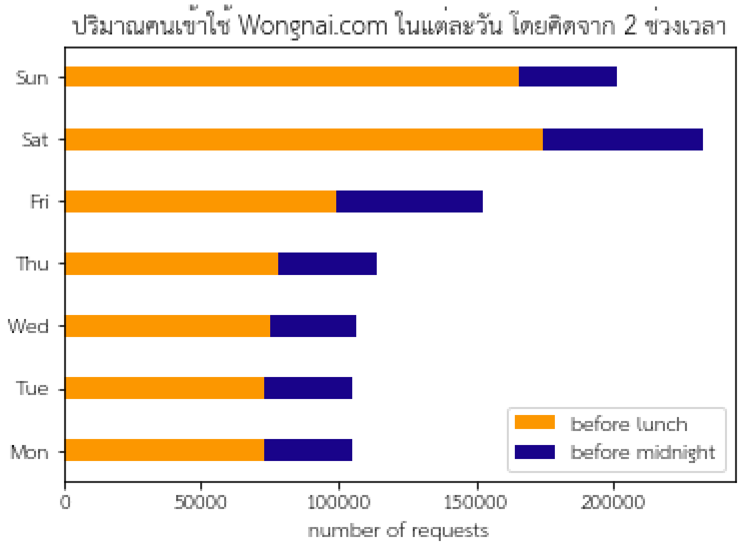
```
import matplotlib.pyplot as plt

width = 0.35       # the width of the bars: can also be len(x) sequence

fig, ax = plt.subplots()

ax.bar(labels, b4lunch, width, label='before lunch',color = '#fc9700')
ax.bar(labels, b4midnight, width, bottom=b4lunch, label='before midnight',color = '

ax.set_ylabel('number of requests')
ax.set_title('ปริมาณคนเข้าใช้ Wongnai.com ในแต่ละวัน โดยคิดจาก 2 ช่วงเวลา')
ax.legend()

plt.show()
```



In [ ]:

```
width = 0.35       # the width of the bars: can also be len(x) sequence

fig, ax = plt.subplots()

ax.barh(labels, b4lunch, width, label='before lunch',color = '#fc9700')
ax.barh(labels, b4midnight, width, left=b4lunch, label='before midnight',color = '#

ax.set_xlabel('number of requests')
ax.set_title('ปริมาณคนเข้าใช้ Wongnai.com ในแต่ละวัน โดยคิดจาก 2 ช่วงเวลา')
ax.legend()
```

```
plt.show()
```



[เช็คชื่อ] โดยให้วาด Bar chart ที่เปรียบเทียบปริมาณคนใช้งาน Wongnai.com สองช่วงเวลา โดยให้กราฟแสดงสัดส่วนของปริมาณคนใช้งานในแต่ละวันด้วย

In [ ]:

```

```

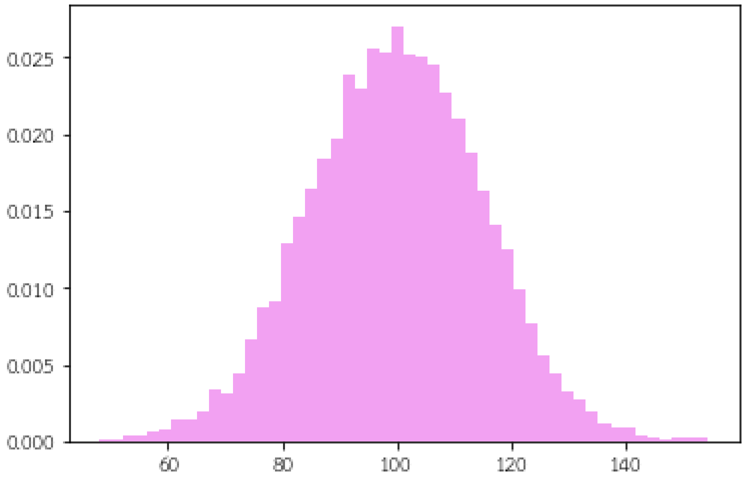
Histogram

กราฟแสดงความถี่ของข้อมูล

ตัวอย่างข้อมูลที่ random มาจาก normal distribution ที่มี mean = 100 และ stdev = 15

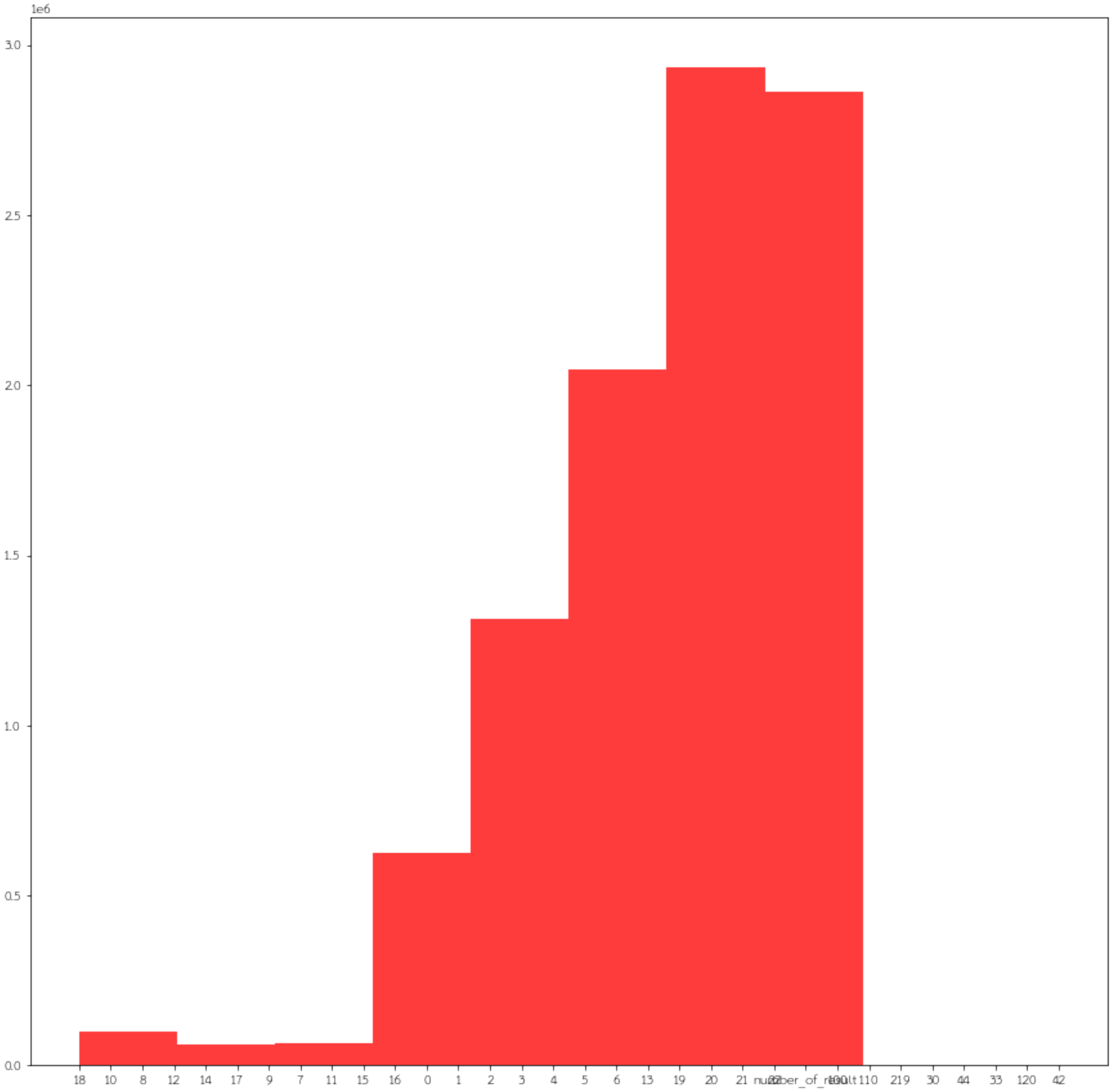
In [ ]:

```
import numpy as np
from matplotlib import pyplot as plt

np.random.seed(2021)

mu, sigma = 100, 15
X = mu + sigma * np.random.randn(10000)

plt.hist(X, 50, density = True, facecolor = 'violet', alpha = 0.75);
```



ตัวอย่างข้อมูล wongnai.com

```
In [ ]: import matplotlib
        matplotlib.rcParams['figure.figsize']=[15,15]
        output = plt.hist(list(data['number_of_result']),10,facecolor = 'red' ,alpha = 0.75
```



แก้ไข แกน x ที่เรียงข้อมูลผิด

```
In [ ]: data.dtypes
```

Out[ ]:

```
Unnamed: 0                     int64
search_id                     object
search_timestamp     datetime64[ns]
user_agent                    object
q                             object
user_id                      float64
session_id                    object
number_of_result              object
lat                           object
long                          object
dtype: object
```

เรียกดู data type ของ ตัวแปร

In [ ]:

```
type(data['number_of_result'][0])
```

Out[ ]:

```
int
```

ตรวจสอบ data type ของตัวแปร

In [ ]:

```
type(data['number_of_result'][0]) == int
```

Out[ ]:

```
True
```

ตรวจสอบดูทุกๆค่าใน column 'number_of_result'

In [ ]:

```
# for x in data['number_of_result']:
#     if type(x) != int:
#         print(f'{x} -> {type(x)}')
```

In [ ]:

```
new_type = data['number_of_result'].astype('int32')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-34-29dba17f7bb1> in <module>()
----> 1 new_type = data['number_of_result'].astype('int32')

/usr/local/lib/python3.7/dist-packages/pandas/core/generic.py in astype(self, dtyp
e, copy, errors)
   5546         else:
   5547             # else, only a single dtype is given
-> 5548             new_data = self._mgr.astype(dtype=dtype, copy=copy, errors=err
ors,)
   5549             return self._constructor(new_data).__finalize__(self, method
="astype")
   5550

/usr/local/lib/python3.7/dist-packages/pandas/core/internals/managers.py in astype
(self, dtype, copy, errors)
    602         self, dtype, copy: bool = False, errors: str = "raise"
    603     ) -> "BlockManager":
--> 604         return self.apply("astype", dtype=dtype, copy=copy, errors=errors)
    605
    606     def convert(

/usr/local/lib/python3.7/dist-packages/pandas/core/internals/managers.py in apply
(self, f, align_keys, **kwargs)
    407                 applied = b.apply(f, **kwargs)
    408             else:
--> 409                 applied = getattr(b, f)(**kwargs)
    410             result_blocks = _extend_blocks(applied, result_blocks)
    411

/usr/local/lib/python3.7/dist-packages/pandas/core/internals/blocks.py in astype(s
elf, dtype, copy, errors)
    593             vals1d = values.ravel()
    594             try:
--> 595                 values = astype_nansafe(vals1d, dtype, copy=True)
    596             except (ValueError, TypeError):
    597                 # e.g. astype_nansafe can fail on object-dtype of strings

/usr/local/lib/python3.7/dist-packages/pandas/core/dtypes/cast.py in astype_nansaf
e(arr, dtype, copy, skipna)
    972         # work around NumPy brokenness, #1987
    973         if np.issubdtype(dtype.type, np.integer):
--> 974             return lib.astype_intsafe(arr.ravel(), dtype).reshape(arr.shap
e)
    975
    976         # if we have a datetime/timedelta array of objects

pandas/_libs/lib.pyx in pandas._libs.lib.astype_intsafe()

ValueError: invalid literal for int() with base 10: 'number_of_result'
```

ลบ record ที่มีค่า ใน column 'number_of_result' เป็น number of result

In [ ]:

```
data[data['number_of_result']=='number_of_result']
```

Out[ ]:

	Unnamed: 0	search_id	search_timestamp	user_agent	q	user_id	session_id	nur
1000016	1000032	search_id	NaT	user_agent	original_q	228667.0	session_id	nu

In [ ]:

```
data = data.drop(1000016)
```

In [ ]:

```
data[data['number_of_result']=='number_of_result']
```

Out[ ]:

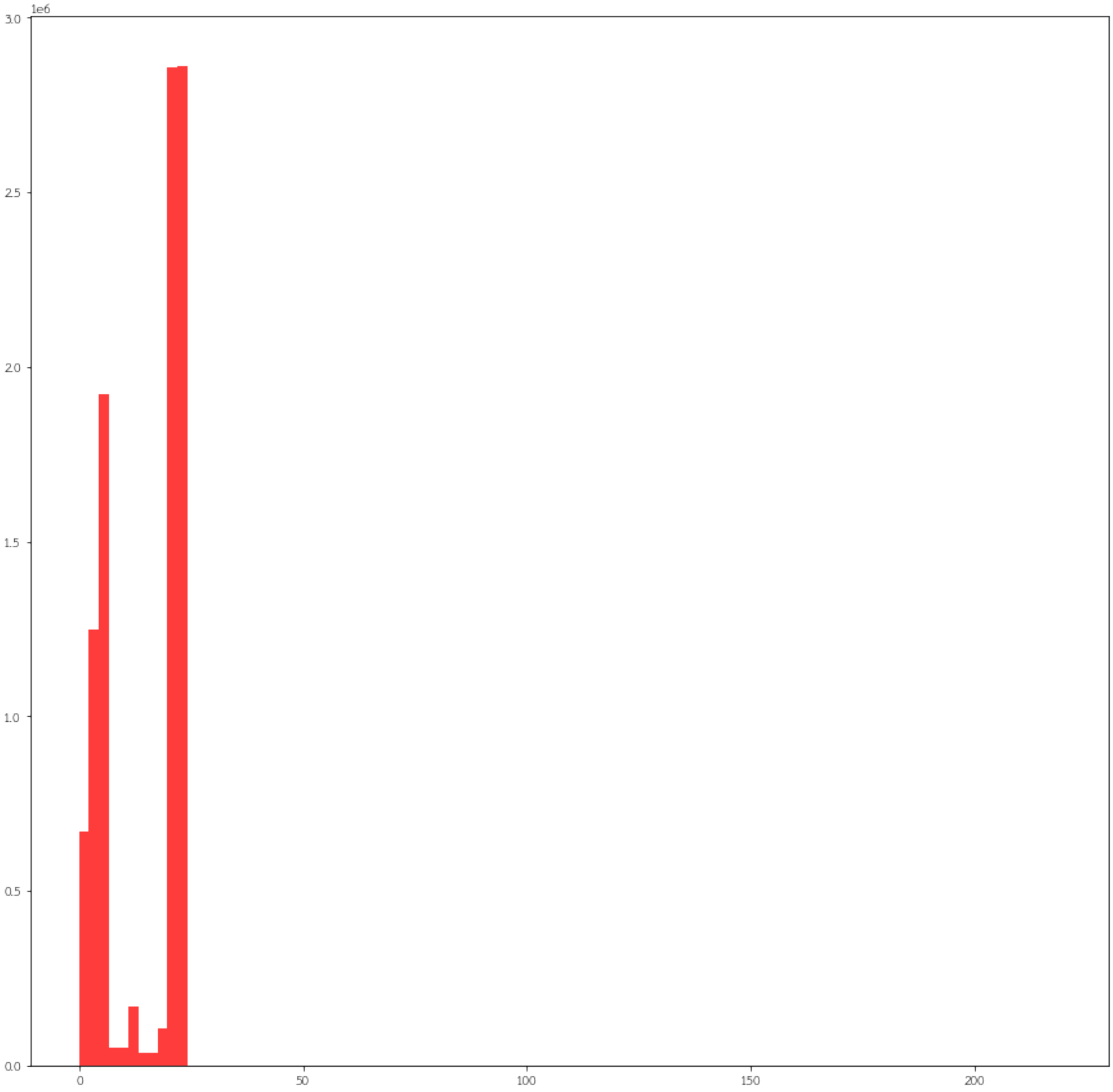
Unnamed: 0	search_id	search_timestamp	user_agent	q	user_id	session_id	number_of_result	la

In [ ]:

```
new_type = data['number_of_result'].astype('int32')
```

In [ ]:

```
output = plt.hist(new_type,100,facecolor = 'red' ,alpha = 0.75)
```



ลบ outlier

In [ ]:

```
new_type_nooutlier = new_type[new_type < 25]
```

In [ ]:

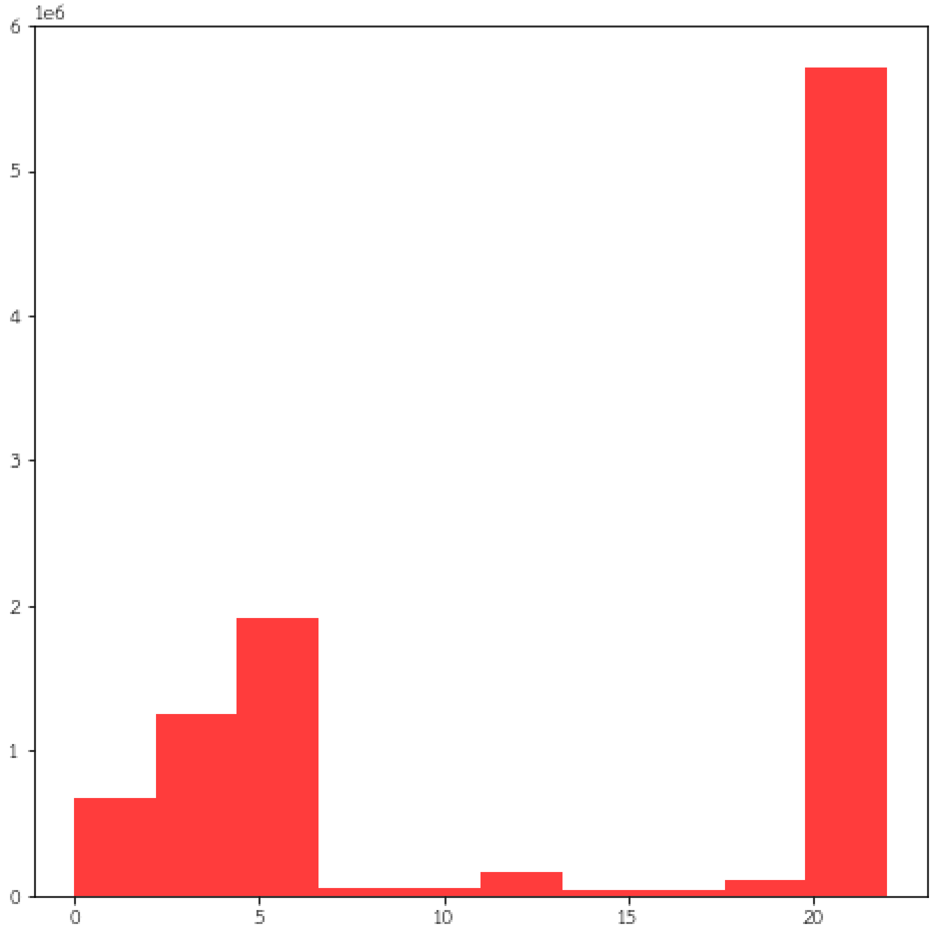
```
new_type.shape[0] - new_type_nooutlier.shape[0]
```

Out[ ]:

```
14
```

In [ ]:

```
matplotlib.rcParams['figure.figsize']=[8,8]
output = plt.hist(new_type_nooutlier,10,facecolor = 'red' ,alpha = 0.75)
```



Quiz 7 เปรียบเทียบความถี่ของแท่งที่มีค่ามากที่สุด กับ แท่งอื่นๆรวมกัน

```
In [ ]: output
```

```
Out[ ]: (array([ 670293., 1247269., 1921441.,   51703.,   50609.,  167502.,
           36883.,   35914.,  105490., 5717238.]),
 array([ 0. ,  2.2,  4.4,  6.6,  8.8, 11. , 13.2, 15.4, 17.6, 19.8, 22. ]),
 <a list of 10 Patch objects>)
```

```
In [ ]: output[0]
```

```
Out[ ]: array([ 670293., 1247269., 1921441.,   51703.,   50609.,  167502.,
          36883.,   35914.,  105490., 5717238.])
```

```
In [ ]: output[0][-1]
```

```
Out[ ]: 5717238.0
```

```
In [ ]: sum(output[0][:-1])
```

```
Out[ ]: 4287104.0
```

Tree map

```
In [ ]: !pip install squarify
```

```
Requirement already satisfied: squarify in /usr/local/lib/python3.7/dist-packages
(0.4.3)
```

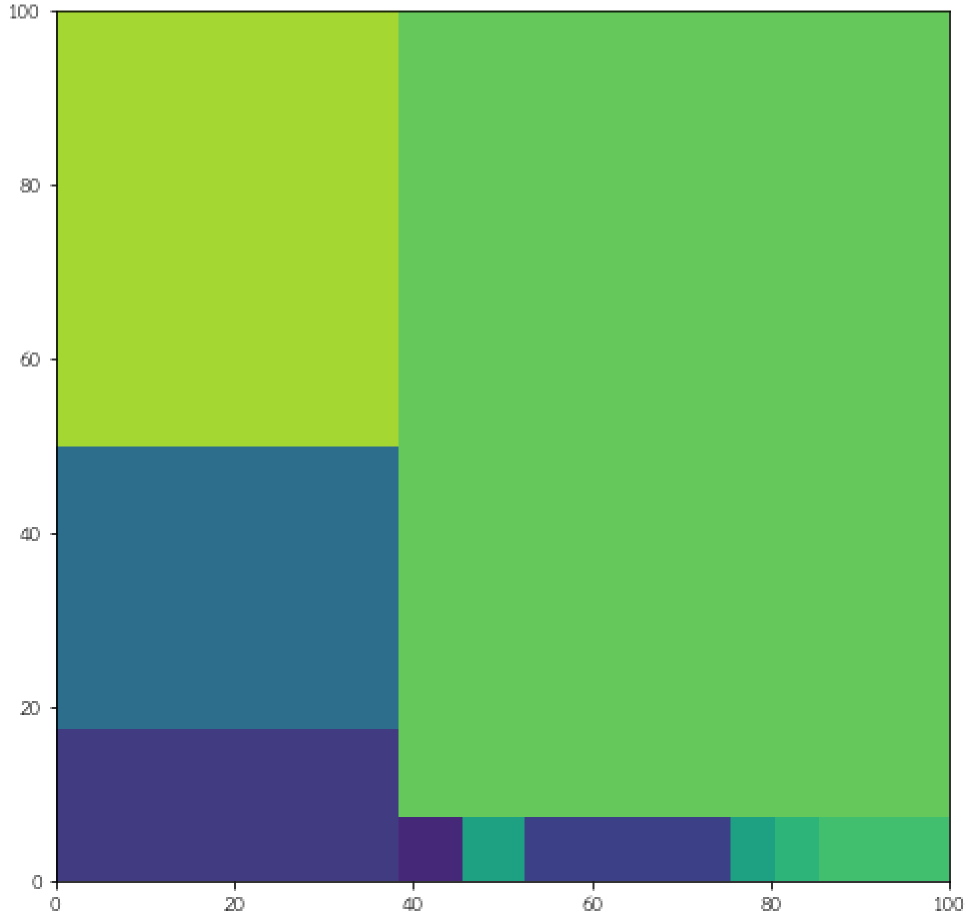
```
In [ ]: import numpy as np
        import matplotlib.pyplot as plt
import squarify
```

In [ ]:

```
squarify.plot(output[0])
```

Out[ ]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f287b486950>
```

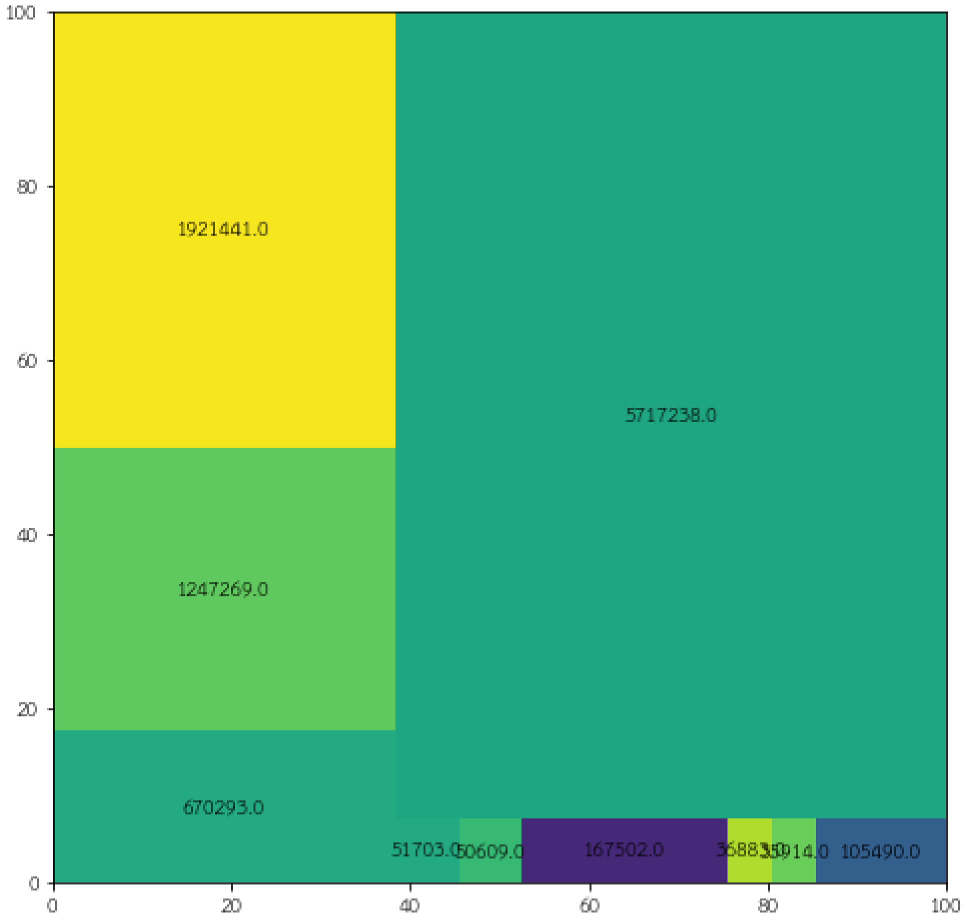


In [ ]:

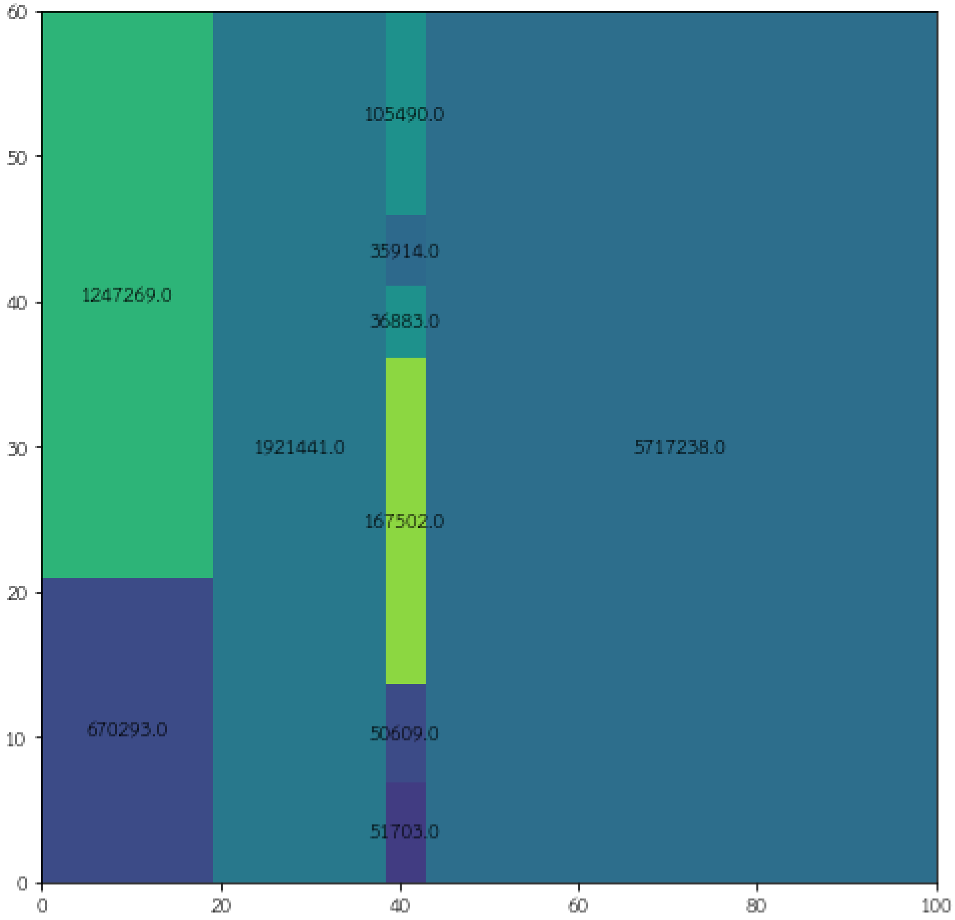
```
squarify.plot(output[0],value=output[0])
```

Out[ ]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f286d23f0d0>
```



```
In [ ]: squarify.plot(output[0],value=output[0],norm_y=60)

Out[ ]: <matplotlib.axes._subplots.AxesSubplot at 0x7f28680ac210>
```


60
50
40
30
20
10
0
0
20
40
60
80
100
105490.0
35914.0
36883.0
1247269.0
1921441.0
5717238.0
167502.0
670293.0
50609.0
51703.0