

Python - Analiza danych z modułem PANDAS

www.udemy.com (<http://www.udemy.com>) (R)

LAB - S08-L002 - plot parameters

1. Zaimportuj **moduł pandas, numpy i matplotlib** i nadaj im standardowe aliasy. Uruchom polecenie powodujące umieszczanie wykresów bezpośrednio w jupyter notebook
2. Uruchom poniższy fragment kodu, aby przygotować dane do rysowania wykresu (wszystkie zastosowane polecenia powinny Ci już być na tym etapie znane):

```
import datetime

#Import data
nasa = pd.read_csv("nasa_facebook_statuses.csv",
                  usecols=["status_published", "num_reactions",
                           "num_comments", "num_shares", "num_likes"])

#Convert column type to date time
nasa["status_published"] = pd.to_datetime(nasa["status_published"])

#Select observations between two datetimes - May 2015
filter =(nasa['status_published'] >= '2016-5-1') & (nasa['status_published'] < '2016-6-1')
nasa = nasa[filter]

#Add column with day only
nasa['day'] = nasa.apply(lambda row: row["status_published"].day,axis=1)

#Remove column with status_published - we don't need it anymore
nasa.drop('status_published', axis='columns', inplace=True)

#Group all columns by sum
nasa_by_day = nasa.groupby(by='day').sum()

nasa_by_day.head()
```

3. Wyświetl wykres dla **nasa_by_day** w domyślnej postaci
4. Wyświetl wykres w rozmiarze 15" na 5". Kolejne wykresy też mają używać tego rozmiaru
5. Do wykresu dodaj tytuł **"NASA Facebook activity"**. Kolejne wykresy też mają używać tego rozmiaru
6. Wyświetl wykres tak aby każda seria danych została pokazana na oddzielnym wykresie. Ta zmiana nie ma obowiązywać w kolejnych wykresach.
7. Ponieważ dane prezentowane na wykresie znacznie różnią się od siebie (część jest bardzo duża a pozostałe są małe), to zmień skalę osi y na logarytmiczną. Ta zmiana nie ma obowiązywać w kolejnych wykresach
8. Wyświetlając wykres spraw, aby na osi x były wyświetlane wszystkie numery dni od 1 do 31 włącznie.
9. Wyświetlając wykres spraw, aby na osi x były wyświetlane wszystkie numery dni od 1 do 31 włącznie. Wartości te mają być wyświetlane pod kątem 45 stopni.
10. Wyświetl wykres dla pierwszych dziesięciu dni. Dodaj parametr, który spowoduje dodatkowo wyświetlenie pod wykresem tabeli z dokładnymi wartościami prezentowanymi również na wykresie.

Rozwiązania:

Poniżej znajdują się propozycje rozwiązań zadań. Prawdopodobnie istnieje wiele dobrych rozwiązań, dlatego jeżeli rozwiązujesz zadania samodzielnie, to najprawdopodobniej zrobisz to inaczej, może nawet lepiej :) Możesz pochwalić się swoimi rozwiązaniami w sekcji Q&A

```
In [1]: import pandas as pd
import numpy as np
import matplotlib as plt
import datetime

%matplotlib inline
```

```
In [2]: import datetime

#Import data
nasa = pd.read_csv("nasa_facebook_statuses.csv",
                  usecols=["status_published", "num_reactions",
                           "num_comments", "num_shares", "num_likes"])

#Convert column type to date time
nasa["status_published"] = pd.to_datetime(nasa["status_published"])

#Select observations between two datetimes - May 2015
filter = (nasa['status_published'] >= '2016-5-1') & \
         (nasa['status_published'] < '2016-6-1')
nasa = nasa[filter]

#Add column with day only
nasa['day'] = nasa.apply(lambda row: row["status_published"].day, axis=1)
#Remove column with status_published - we don't need it anymore
nasa.drop('status_published', axis='columns', inplace=True)

#Group all columns by sum
nasa_by_day = nasa.groupby(by='day').sum()

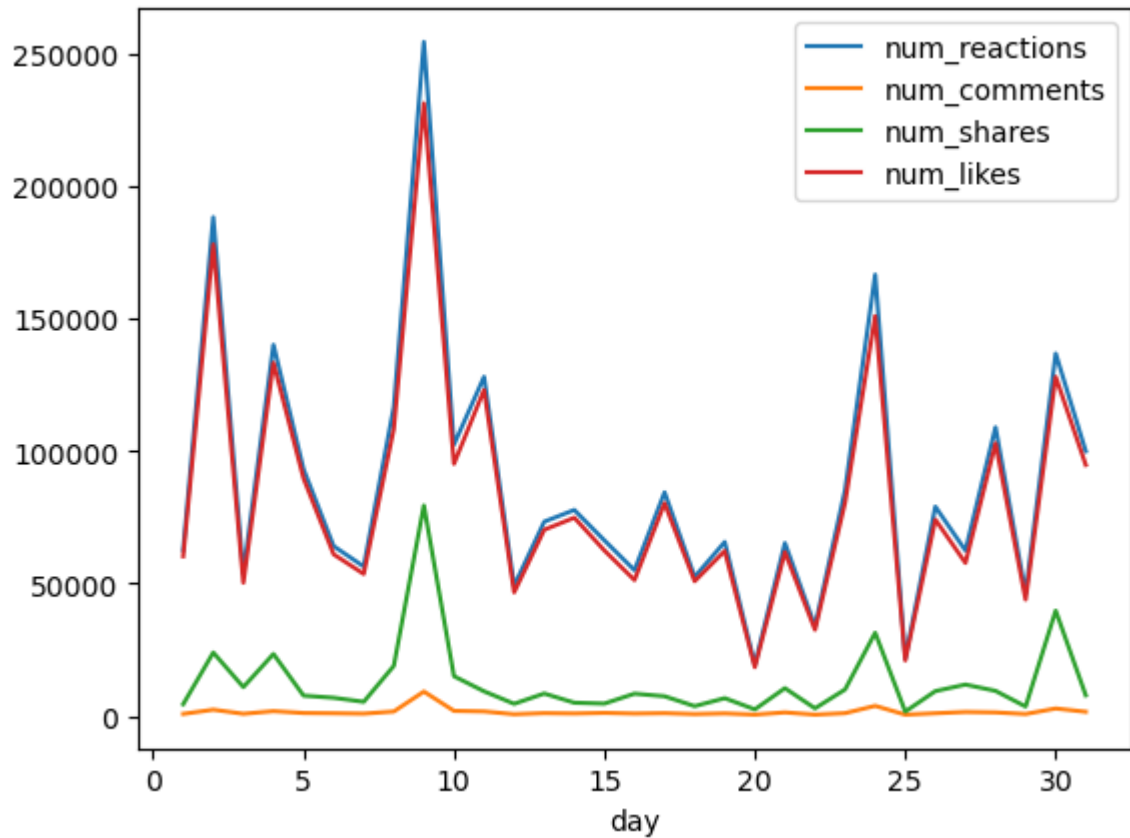
nasa_by_day.head()
```

Out[2]:

	num_reactions	num_comments	num_shares	num_likes
day				
1	62629	727	4403	60243
2	188255	2328	23897	177989
3	53700	770	10858	50264
4	140098	1776	23360	133302
5	93146	1070	7627	89657

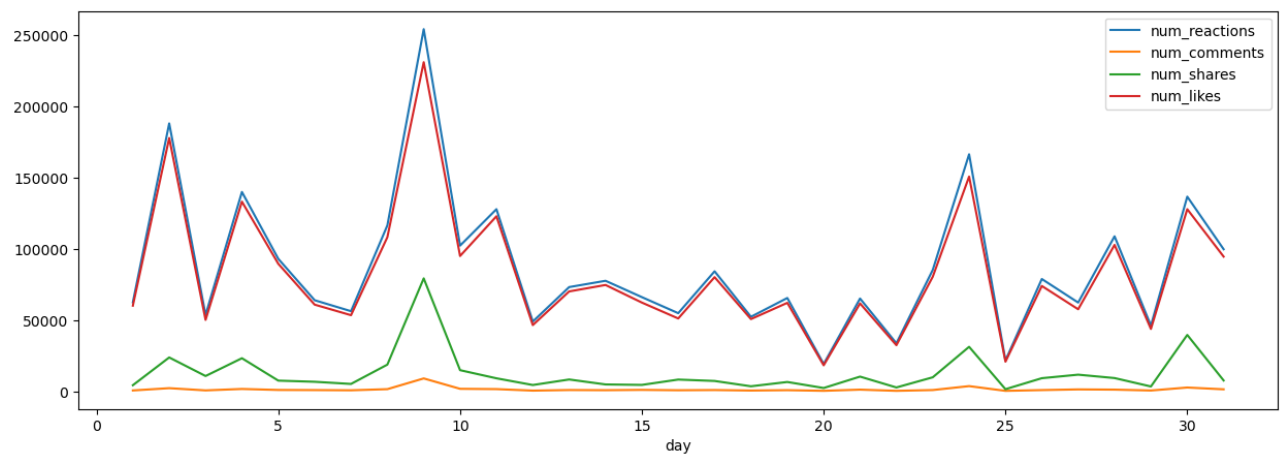
```
In [3]: nasa_by_day.plot()
```

```
Out[3]: <Axes: xlabel='day'>
```



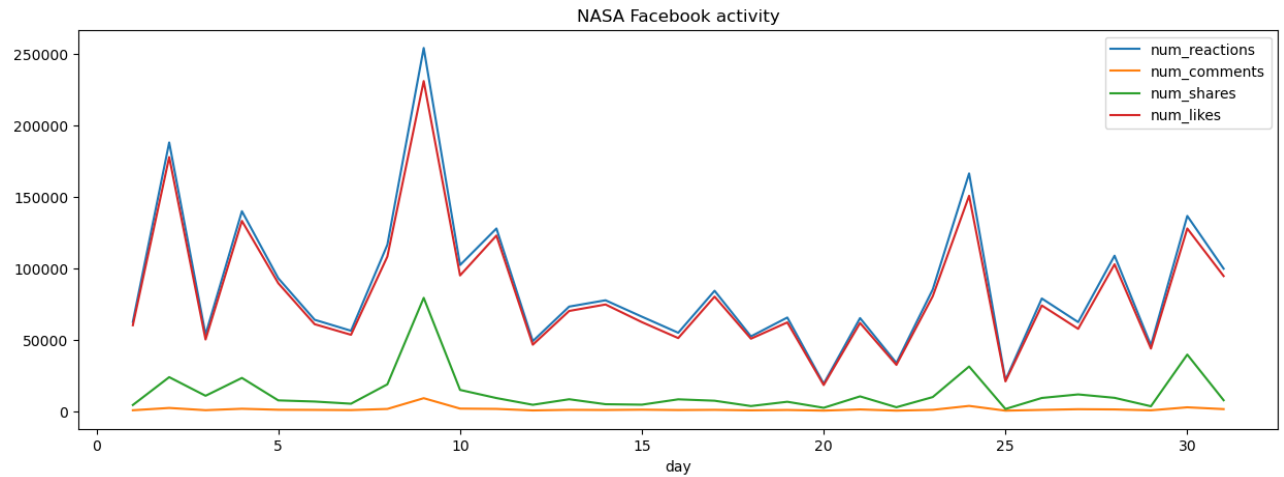
```
In [4]: nasa_by_day.plot(figsize=(15,5))
```

```
Out[4]: <Axes: xlabel='day'>
```



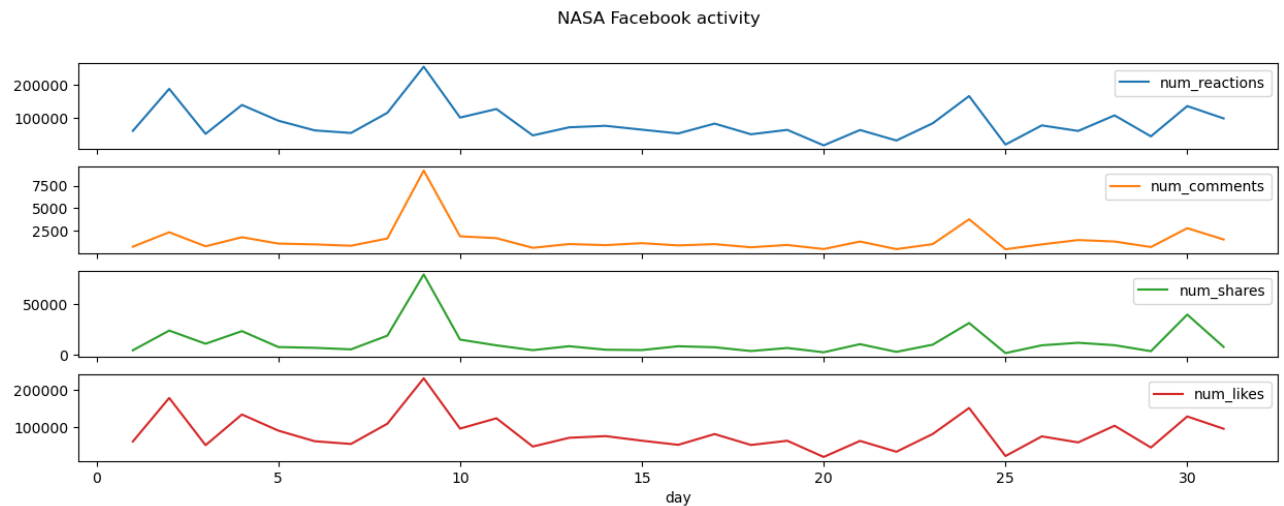
```
In [5]: nasa_by_day.plot(figsize=(15,5), title = "NASA Facebook activity")
```

```
Out[5]: <Axes: title={'center': 'NASA Facebook activity'}, xlabel='day'>
```



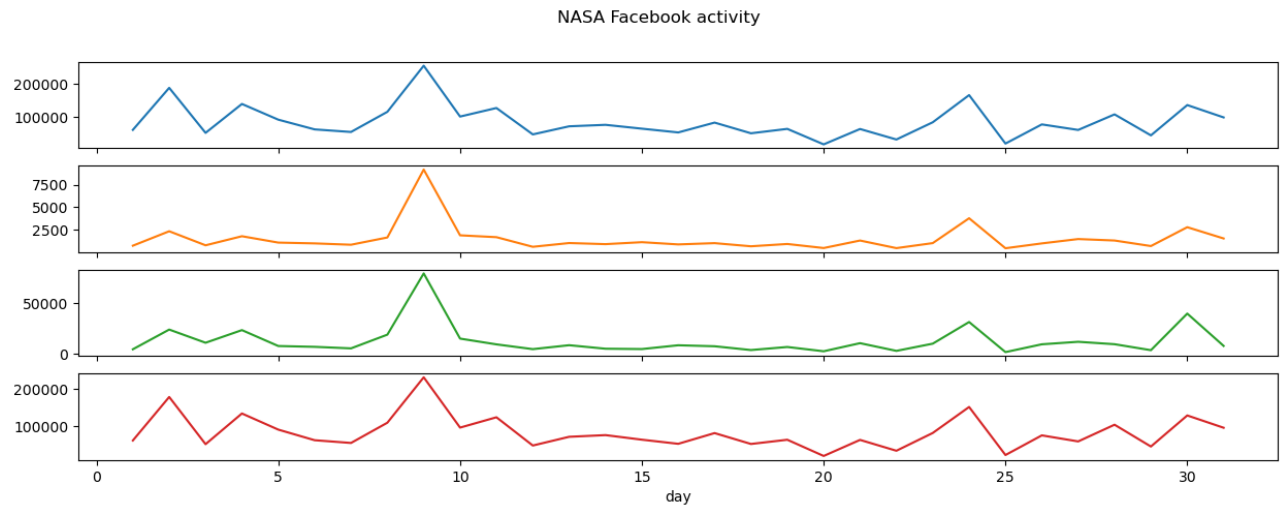
```
In [6]: nasa_by_day.plot(figsize=(15,5), title = "NASA Facebook activity",  
          subplots=True)
```

```
Out[6]: array([<Axes: xlabel='day'>, <Axes: xlabel='day'>, <Axes: xlabel='day'>,  
               <Axes: xlabel='day'>], dtype=object)
```



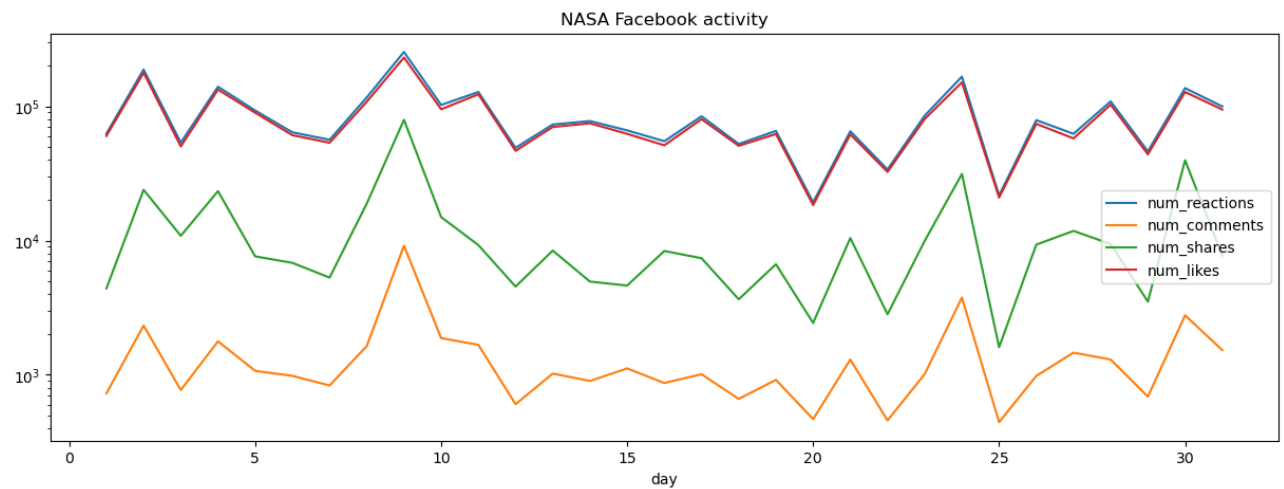
```
In [7]: nasa_by_day.plot(figsize=(15,5), title = "NASA Facebook activity",  
subplots=True, legend=False)
```

```
Out[7]: array([<Axes: xlabel='day'>, <Axes: xlabel='day'>, <Axes: xlabel='day'>,  
<Axes: xlabel='day'>], dtype=object)
```



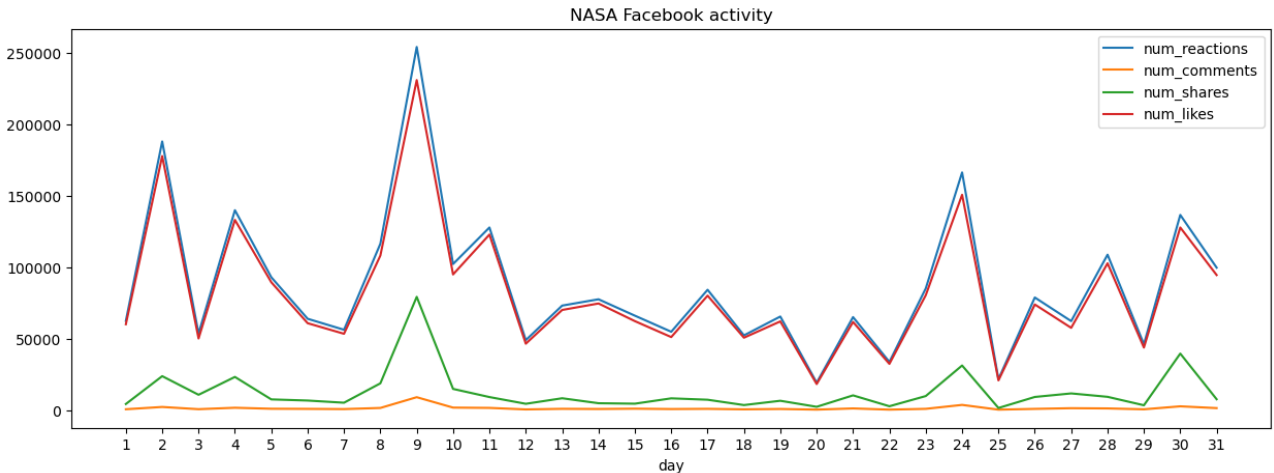
```
In [8]: nasa_by_day.plot(figsize=(15,5), title = "NASA Facebook activity",  
logy=True)
```

```
Out[8]: <Axes: title={'center': 'NASA Facebook activity'}, xlabel='day'>
```



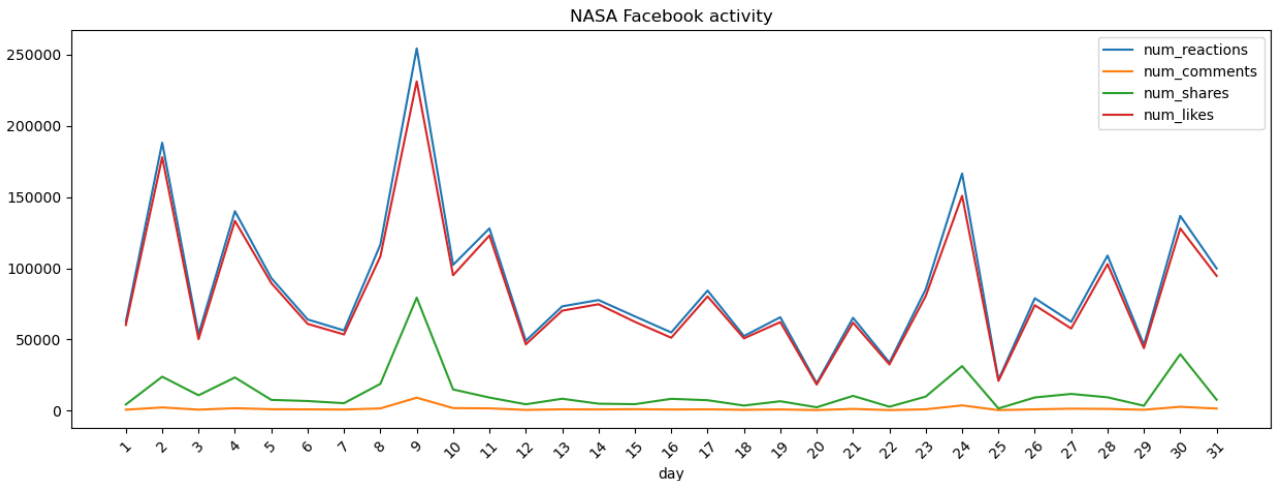
```
In [9]: nasa_by_day.plot(figsize=(15,5), title = "NASA Facebook activity",
        xticks=range(1,32))
```

Out[9]: <Axes: title={'center': 'NASA Facebook activity'}, xlabel='day'>



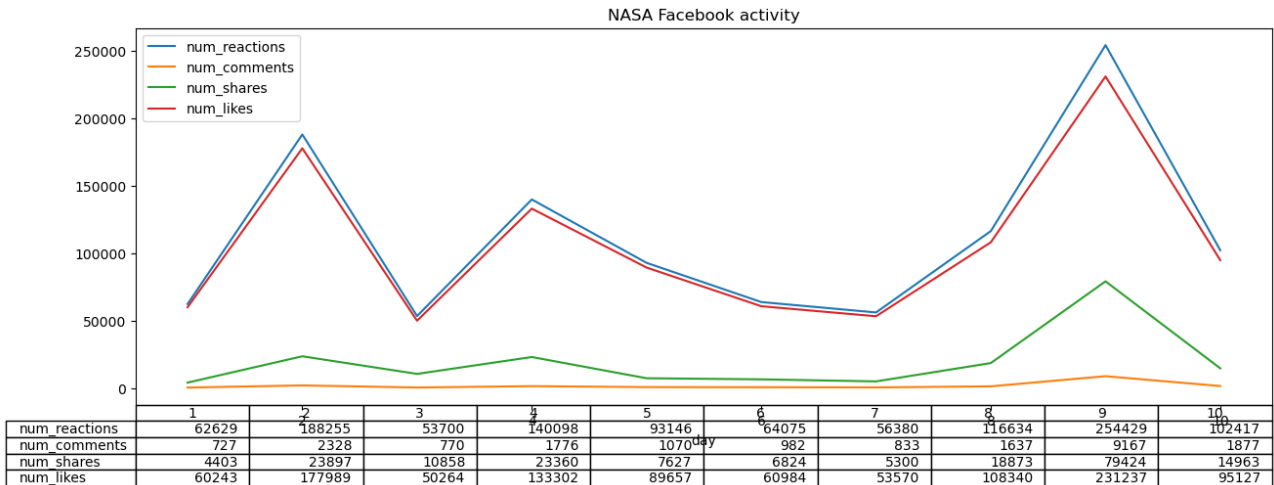
```
In [10]: nasa_by_day.plot(figsize=(15,5), title = "NASA Facebook activity",
        xticks=range(1,32), rot=45)
```

Out[10]: <Axes: title={'center': 'NASA Facebook activity'}, xlabel='day'>



```
In [11]: nasa_by_day[:10].plot(figsize=(15,5), title = "NASA Facebook activity",
        table=True)
```

Out[11]: <Axes: title={'center': 'NASA Facebook activity'}, xlabel='day'>



In []: