

Python - Analiza danych z modulem PANDAS

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

Wprowadzenie

Na lekcji zobaczysz, jak zaimportować moduły i korzystać z ich funkcji. Najważniejsza jest sama czynność importu modułów. Kolejne kroki to korzystanie z tych modułów. Jeśli masz ochotę możesz więc w tym laboratorium narysować kilka wykresów.

S02-L001-LAB Korzystanie z modułów

1. Zaimportuj moduły: **pandas**, **numpy**, **matplotlib** (tylko **pyplot**), **math** i nadaj im standardowe aliasy
2. Wywołując "magiczną funkcję" zmień ustawienia tak, aby wykresy były wyświetlane w Jupyter Notebook
3. Zdefiniuj tabelę **days** zawierającą liczby: **1,2,3,4,5**
4. Zdefiniuj tabelę **amount** zawierającą liczby: **1,5,20,450,700**
5. Narysuj wykres, który jako argumenty przyjmie tablicę **days**, a jako wartości tablicę **amount**
6. Zmień wykres tak, aby wartości były zaznaczane niebieskimi kwadratami zamiast domyślnej linii łączącej punkty
7. Korzystając z dostępnej w module **numpy** funkcji **arange** zdefiniuj tabelę **months** zawierającą numery miesięcy **od 1 do 12** i wyświetl tą tablicę
8. Zdefiniuj tabelę **income**, której wartości będą wyliczane na podstawie tabeli **months** wzorem:

```
100 + 3 * numer miesiaca
```

(powinno być 12 wartości zaczynając od: 103, 106, 109 itd.). Wyświetl tą tablicę

9. Wyświetl wykres, który jako argumenty przyjmie **months**, a wartości z tabeli **income**. Punkty danych mają się wyświetlić jako zielone kropki.
10. Zdefiniuj tablicę **cost**, której wartości będą wyliczane na podstawie tabeli **months** wzorem:

```
50 + 10 * month
```

11. Wyświetl wykres, w którym na osi X będą miesiące z tabeli **months**, prezentujący dwie serie danych: **income** i **cost**. Income powinien być prezentowany przez zielone kropki, a cost przez czerwone trójkąty
12. Narysuj wykres funkcji 2^x (dwa do potęgi x), dla x z zakresu **od -5 do 5** z dokładnością **0.1**. Opisz osie: **X - "Arguments"**, **Y - "Values"**. W tym celu:
 - Zdefiniuj tablicę **x** przyjmującą wartości -5, -4.9, -4.8....
 - Opis osie korzystając z funkcji **xlabel** i **ylabel**
 - Narysuj wykres, skorzystaj z funkcji **pow(2,x)**

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.pyplot as plt
import math as math
```

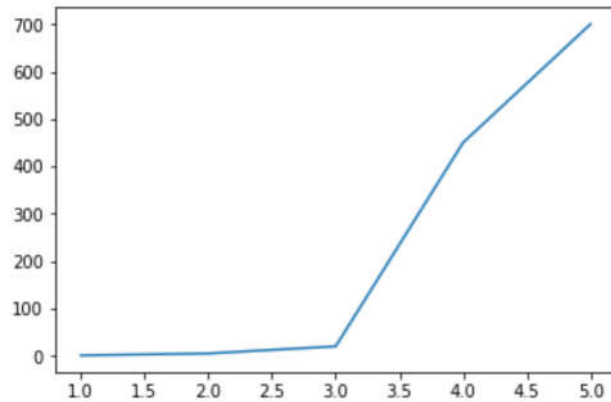
```
In [2]: %matplotlib inline
```

```
In [3]: days=[1,2,3,4,5]
```

```
In [4]: amount=[1,5,20,450,700]
```

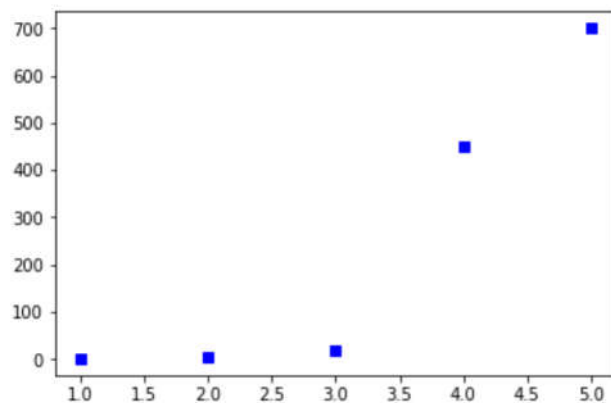
```
In [5]: plt.plot(days, amount)
```

```
Out[5]: [<matplotlib.lines.Line2D at 0x24544727cf8>]
```



```
In [6]: plt.plot(days, amount, "bs")
```

```
Out[6]: [<matplotlib.lines.Line2D at 0x245447536a0>]
```



```
In [7]: months = np.arange(1,13,1)  
months
```

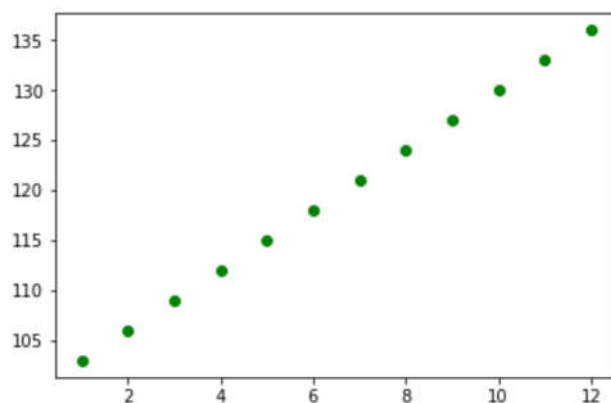
```
Out[7]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12])
```

```
In [8]: income = 100 + 3 * months  
income
```

```
Out[8]: array([103, 106, 109, 112, 115, 118, 121, 124, 127, 130, 133, 136])
```

```
In [9]: plt.plot(months, income, "go")
```

```
Out[9]: [<matplotlib.lines.Line2D at 0x24544837f60>]
```

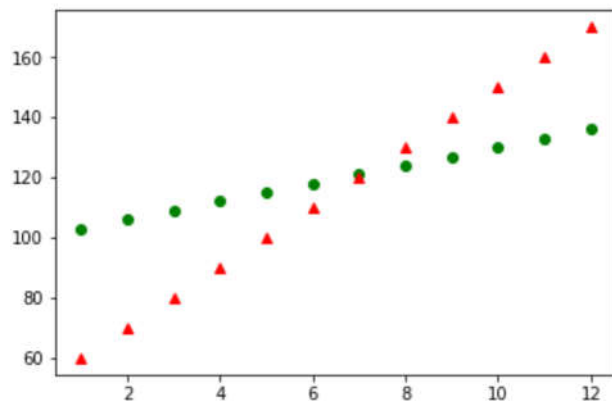


```
In [10]: cost = 50 + 10 * months
cost
```

```
Out[10]: array([ 60,  70,  80,  90, 100, 110, 120, 130, 140, 150, 160, 170])
```

```
In [11]: plt.plot(months, income, "go", months, cost, "r^" )
```

```
Out[11]: [<matplotlib.lines.Line2D at 0x24544897860>,
<matplotlib.lines.Line2D at 0x245448979e8>]
```



```
In [12]: x=np.arange(-5,5,0.1)
```

```
plt.xlabel('arguments')
plt.ylabel("values")
plt.plot(x, pow(2,x))
plt.show()
```

