

# AnalizaKretanja1

August 25, 2019

## 1 Plot grafa podataka iz .csv filea

Demo kako uz pomo pythona i *matplotlib* biblioteke za prikaz grafova

Prvo uvezemo potrebne biblioteke

```
[1]: from numpy import genfromtxt, arange, sin, pi
import matplotlib.pyplot as plt
from matplotlib import style
import numpy as np
```

Unese se ime datoteke s podacima i mapiraju se polja sukladno zapisanome.

U ovom primjeru podaci su razdvojeni s znakom ',' ali esti je sluaja kada su podaci odvojeni nekim drugim znakom te se to treba posebno naznaiti kako bi program znao granice izmeu polja.

```
[2]: filename='GPSLOG10.CSV'
#plt.style.use('ggplot')
data=genfromtxt(filename, delimiter=',',
    →names=['Sentence', 'Time', 'Validity', 'Latitue', 'NS', 'Longitude', 'EW', 'Speed',
    →'Direction', 'Date', 'NA1', 'NA2', 'Checksum', 'Temperature'])
```

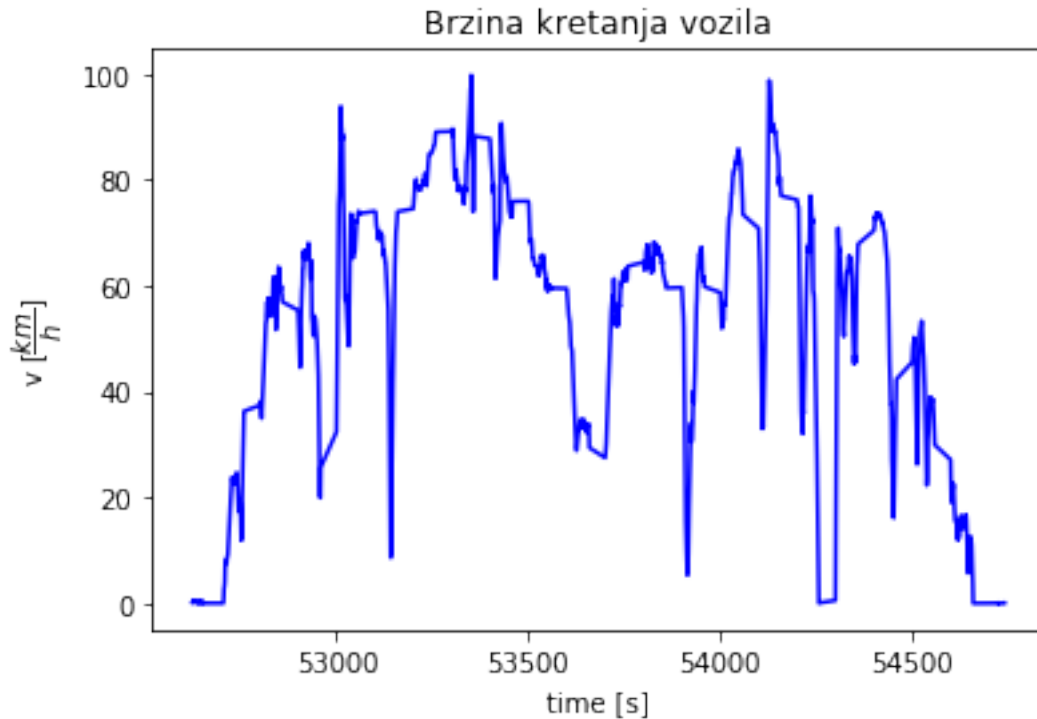
Sada smo spremni za prikazati prikupljene podatke.

Prvo moemo prikazati jednostavan s/t graf - brzinu u vremenu. Kako je brzina zapisana u vorovima, a mi je elimo prikazati u km/h potrebno izvriti konverziju. 1 nautina milja odgovara 1.852 km.

Svaki graf treba imati oznaene osi.S komandom plt.xlabel i ylabel oznaili smo osi grafa i analogno tome imenovan je i graf kako bi itatelj znao to graf predstavlja.

```
[83]: #otvori graf u novom prozoru
#%matplotlib
plt.plot(data['Time'],data['Speed']*1.852, 'b-')

plt.xlabel ('time [s]')
plt.ylabel ('v [$\dfrac{km}{h}$]')
plt.title('Brzina kretanja vozila')
plt.savefig('GrafKretanjaBrzineVozila.png',format='png', bbox_inches='tight',
    →dpi=600)
```



Naravno, pojedinačni grafovi se mogu posebno spremiti u visokoj rezoluciji i u ljenom formatu za kasniju upotrebu

```
[60]: plt.savefig('GrafKretanjaBrzineVozila.png', format='png', bbox_inches='tight',
    ↪dpi=600)
```

<Figure size 432x288 with 0 Axes>

Dodatno se mogu izračunati i pogledati razni podaci koje nas zanimaju.

Ako npr. elimo znati koja je bila maksimalna brzina kojom se vozilo kretalo to se može vidjeti na sljedećem:

```
[56]: print('Maksimalna brzina = ', np.max(data['Speed']*1.852) , 'km/h')
```

Maksimalna brzina = 99.73020000000001 km/h

Ako nas zanimaju podaci o temperaturi moguće je i koristiti i ugrađene statističke funkcije za izračunati eljene podatke

```
[104]: # prikazi graf inline
    %matplotlib inline
    plt.rcParams["figure.figsize"] = (20,10)
    plt.plot(data['Time'],data['Temperature'], 'g', linewidth=2)

    plt.xlabel ('time [s]')
```

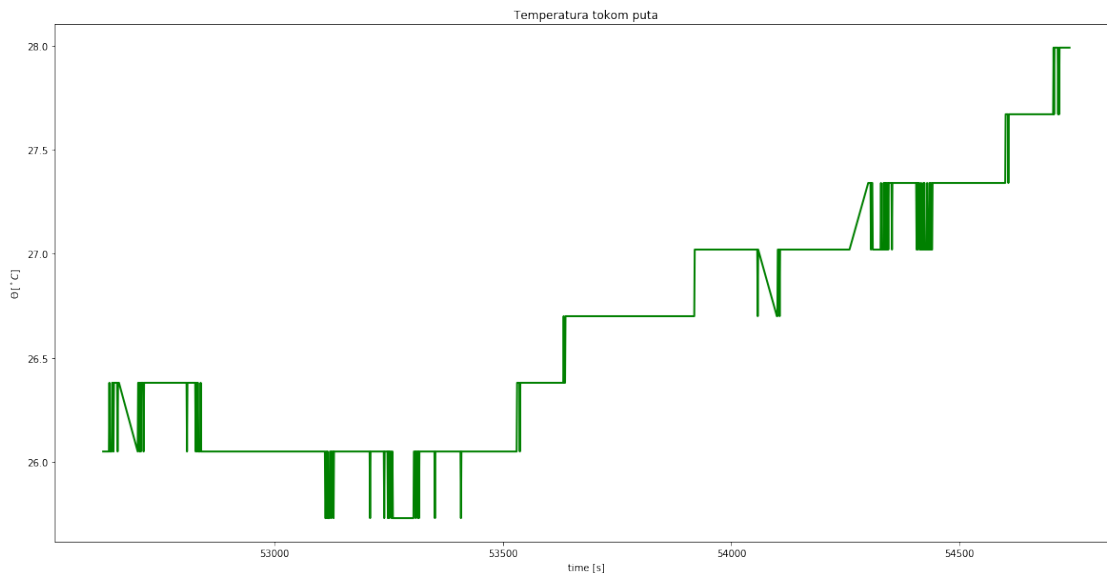
```

plt.ylabel ('$\\Theta \\, [^\\circ C]$')
plt.title('Temperatura tokom puta')

print ('Minimalna temperatura: ', np.min(data['Temperature']), 'řC')
print ('Maximalna temperatura: ', np.max(data['Temperature']), 'řC')
print ('Razlika temperature: ', np.max(data['Temperature']) - np.
    →min(data['Temperature']), 'řC')
print ('Prosjena temperatura: ', '{:.2f}'.format(np.mean(data['Temperature'])), 'řC',
    →'řC š ',
        '{:.2f}'.format(np.std(data['Temperature'])), 'řC')

```

Minimalna temperatura: 25.73 řC  
 Maximalna temperatura: 27.99 řC  
 Razlika temperature: 2.2599999999999998 řC  
 Prosjena temperatura: 26.64 řC š 0.57 řC



[ ]: