Istraživanje podataka

Vežbe 3

5. März 2021

Outline

Outline

Programski jezik Python

- Za programiranje u pj Python može se koristiti integrisano razvojno okruženje PyChram
- ekstenzija programa *.py*
- Pri istraživanju podataka često se koriste Jupiter notebook dokument koji može da sadrži kombinaciju koda napisanog u pj Python, rezultat koda, tekst i slike. Ekstenzija dokumenta je *.ipynb*.

NumPy

Biblioteka *NumPy* osnovna biblioteka za naučno izračunavanje u programskom jeziku Python.

- **ndarray** (alias array) klasa za nizove
  import numpy as np
  b = np.array([6, 7, 8])
- dimenzije se nazivaju osama (eng. axes)

pandas

Biblioteka *pandas* obezbeđuje strukture podataka za rad sa relacionim ili označenim podacima.
Osnovne strukture podataka su:

- **Series** za rad sa 1D podacima
- **DataFrame** za rad sa 2D podacima

Outline

Klasa Series

Neki atributi:

- **index** - oznake vrednosti
- **values** - vrednosti

Neke metode:

- **keys** - oznake vrednosti
- **value_counts** - broj pojavljivanja za svaku vrednost

Operacije $(+, -, *, /)$ se vrše nad elementima sa istim vrednostima indeksa

Series - primeri

```
import pandas as pd
import numpy as np
```

```
s1 = pd.Series([1,2,3,5])
```

```
0    1
1    2
2    3
3    5
dtype: int64
```

```
s1[2]
```

```
3
```

```
s1.get(1)
```

```
2
```

```
s1.get(8, np.nan)
```

```
nan
```

Series - primeri

```
s1 = pd.Series([1,2,3,5], index=['a', 'b', 'c', 'd'])
```

```
a    1
b    2
c    3
d    5
dtype: int64
```

```
s1['b']
```

```
2
```

Series - primeri

```
s1.index
```

```
Index(['a', 'b', 'c', 'd'], dtype='object')
```

```
s1.values
```

```
array([1, 2, 3, 5], dtype=int64)
```

```
s1.value_counts()
```

```
5    1
3    1
2    1
1    1
dtype: int64
```

Klasa DataFrame

DataFrame je 2D označena struktura podataka sa kolonama koje mogu biti različitih tipova.

- **index** - oznaka reda
- **column** - oznaka kolone

pandas.DataFrame(data, index, columns)

DataFrame - primer

```
d1= {'prva': pd.Series([1,2,3, 4], index=['a', 'b', 'c', 'd']),
     'druga': pd.Series(['x', 'y', 'z'], index=['a', 'b', 'c']
                        )}
```

```
df1=pd.DataFrame(d1)
```

```
df1
```

	prva	druga
a	1	x
b	2	y
c	3	z
d	4	NaN

DataFrame

Za tipove kolona *pandas* koristi

- nizove biblioteke NumPy (float, int, bool)
- za čuvanje niski koristi dtype *object*
- ...

DataFrame

Izdvajanje redova i kolona:

- **iloc** - izdvajanje redova i kolona prema poziciji
  iloc[izbor reda, izbor kolone]
  izbor: pozicija (jedan red), [lista pozicija], opseg (donja pozicija : gornja pozicija)
- **loc** - izdvajanje redova i kolona prema oznaci/indeksu
  loc[izbor reda, izbor kolone]
  izbor: labela, [lista labela], opseg (donja labela : gornja labela)
- **df[col]** - izdvajanje kolone
- **df[uslov]** - izdvajanje redova prema uslovu

DataFrame - primeri

```
df1['prva']
```

```
d    4
c    3
b    2
a    1
Name: prva, dtype: int64
```

```
df1['prva']['c']
```

```
3
```

DataFrame - primeri

```
df2
```

	cat	num	obj
0	a	1	c
1	b	2	d
2	a	2	e
3	NaN	3	c

```
df2.iloc[0,1]   #[pozicija reda, pozicija kolone]
```

```
1
```

DataFrame - primeri

```
df2.iloc[1,1:]
```

```
num     2
obj     d
Name: 1, dtype: object
```

```
df2.iloc[[1],1:]
```

	num	obj
1	2	d

```
df2.iloc[[1,3],1:]
```

	num	obj
1	2	d
3	3	c

DataFrame - primeri

```
df1
```

	druga	prva
d	NaN	4
c	z	3
b	y	2
a	x	1

```
df1.loc['a']
```

```
druga    x
prva     1
Name: a, dtype: object
```

DataFrame - primeri

```
df1.loc['c':'a']
```

	druga	prva
c	z	3
b	y	2
a	x	1

```
df1.loc[['a', 'c'], ['prva', 'druga']]
```

	prva	druga
a	1	x
c	3	z

DataFrame - primeri

```
df1[df1['prva']>1]
```

	druga	prva
d	NaN	4
c	z	3
b	y	2

```
df1[df1['prva']>1]['druga']
```

```
d    NaN
c      z
b      y
Name: druga, dtype: object
```

```
df1[df1['prva']>1][['druga']]
```

	druga
d	NaN
c	z
b	y

DataFrame

Neke metode:

- **sort_index** - sortiranje redova/kolona prema oznakama
- **sort_values** - sortiranje redova/kolona prema vrednostima
- **describe** - prikaz izračunatih statistika nad kolonama

DataFrame

Statistike:

- **mean**
- **quantile**
- **mode**
- **nunique** - broj različitih vrednosti
- ...