Seminar 4 Python

*Acest **seminar** prezinta **Seminar 4 Python**.*
In acest PDF poti vizualiza cuprinsul si bibliografia (daca sunt disponibile) si aproximativ doua pagini din documentul original.
*Arhiva completa de pe site contine 10 fisiere, intr-un numar total de **10 pagini**.*
Fisierele documentului original au urmatoarele extensii: docx, xml, iml, py, csv.

Extras

Exemplu instalare pachet scikit-learn

Din https://pypi.org/project/scikit-learn/ copiem pip install scikit-learn

În Command Prompt:

C:UsersSimona OpreaAppDataLocalProgramsPythonPython36-32Scripts>pip install scikit-learn

Alte pachete Python: scipy, six, cycler, pyparsing, kiwisolver, python-dateutil, matplotlib, pytz, pandas, seaborn, numpy, sklearn, statsmodels etc.

Upgrade PIP

https://datatofish.com/upgrade-pip/

Exemplu 1. Gruparea unui set de valori în 3 clustere

```
import matplotlib.pyplot as plt

import numpy as np

from sklearn.cluster import KMeans

X = np.array([[5,3],

[10,15],

[15,12],

[24,10],

[30,45],

[85,70],

[71,80],

[60,78],

[55,52],

[80,91]])

kmeans = KMeans(n_clusters=3)
```

```
kmeans.fit(X)

print(kmeans.cluster_centers_)

print(kmeans.labels_)

f1 = plt.figure()

plt.scatter(X[:,0],X[:,1], label='True Position')

f2 = plt.figure()

plt.scatter(X[:,0], X[:,1], c=kmeans.labels_, cmap='rainbow')

f3 = plt.figure()

plt.scatter(kmeans.cluster_centers_[:,0] ,kmeans.cluster_centers_[:,1], color='black')

plt.show()
```

Clusterizare k-means în Python cu pachetul scikit-learn

Scufundarea Titanicului în 1912 a dus la 1502 victime din cei 2224 pasageri și membri ai echipajului.

Se vor utiliza două seturi de date train.csv și test.csv, ce conțin informații legate de pasageri. Diferența majoră dintre cele două seturi la nivel de coloane constă în coloana Suvived prezentă doar în setul de antrenare. Se poate considera că supraviețuirea a fost influențată de anumite atribute, cum ar fi vârsta, sexul, clasa biletului de călătorie etc. Pornind de la aceste caracteristici, se vor grupa (clusteriza) pasagerii din setul de date de test în supraviețuitori și nesupraviețuitori, comparându-se rezultatele cu setul de date pentru antrenare.

Exemplu 2. Pas 1. Import biblioteci

```
import pandas as pd

import numpy as np

from sklearn.cluster import KMeans

from sklearn.preprocessing import LabelEncoder

from sklearn.preprocessing import MinMaxScaler

import seaborn as sns

import matplotlib.pyplot as plt
```

Exemplu 2. Pas 2. Citirea fișierelor și afișarea primelor 5 înregistrări

```
pd.options.display.max_columns = 12

test = pd.read_csv('test.csv')

train = pd.read_csv('train.csv')

print('*****test*****')

print(test.head())
```

```
print('*****train*****')

print(train.head())
```

Exemplu 2. Pas 3. Calcul statistici de bază

```
print('*****test_stats*****')

print(test.describe())

print('*****train_stats*****')

print(train.describe())
```

Anumiți algoritmi machine learning, inclusiv k-means, nu permit valori lipsă. Astfel, vor fi identificate valorile lipsă.

Exemplu 2. Pas 4. Vizualizare denumire coloane din setul train și indentificare valori lipsă

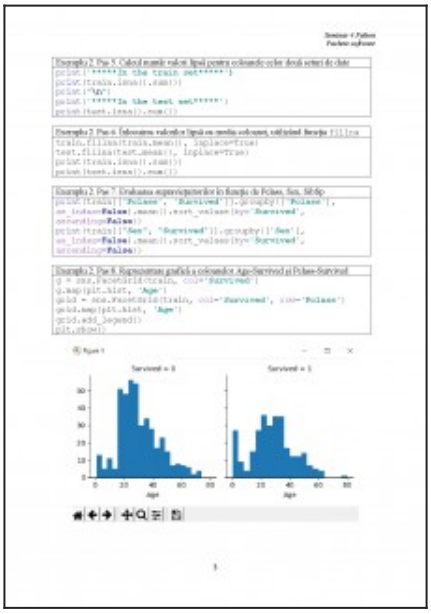
```
print(train.columns.values)

print('*****train_ valori_lipsă *****')

print(train.isna())

print('*****test_valori_lipsă*****')

print(test.isna())
```

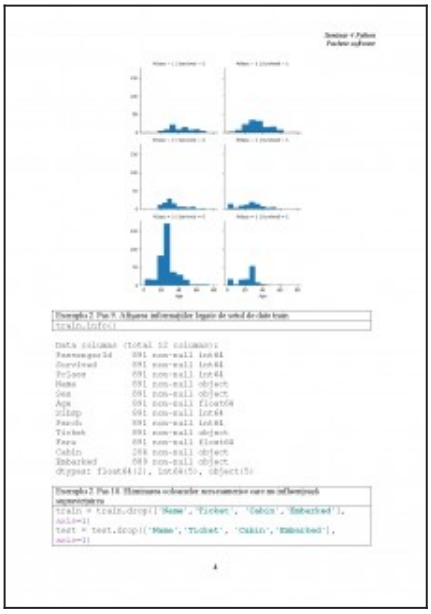
. . . . . .
. . . . . .
. . . . . .

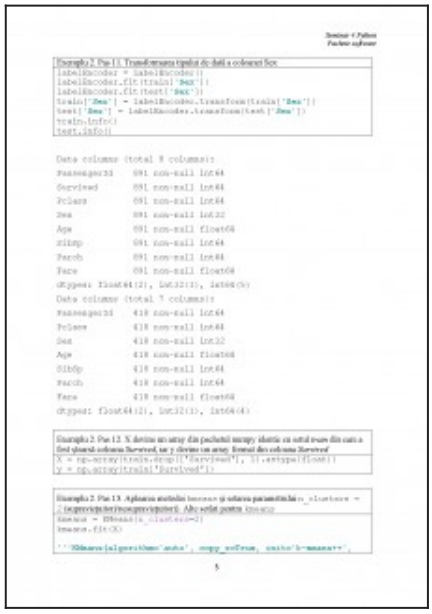
Bibliografie

1 https://stackabuse.com/k-means-clustering-with-scikit-learn/ 2 https://www.datacamp.com/community/tutorials/k-means-clustering-python 3 Wes McKinney, 2nd Edition of Python for Data Analysis DATA WRANGLING WITH PANDAS, NUMPY, AND IPYTHON, O'Reilley 4 https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc 5 https://www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.RegressionResults.html

Imagini din documentul complet:







Mai multe detalii se gasesc in [pagina documentului](#) din [Biblioteca.RegieLive.ro](#)