Ανάλυση Δεδομένων Μάθησης (Learning Analytics)

Τζήμας Δημήτριος,
Συντονιστής Εκπ. Έργου Πληροφ. Δυτ. Μακεδονίας
detzimas@csd.auth.gr
blogs.sch.gr/dtzimas

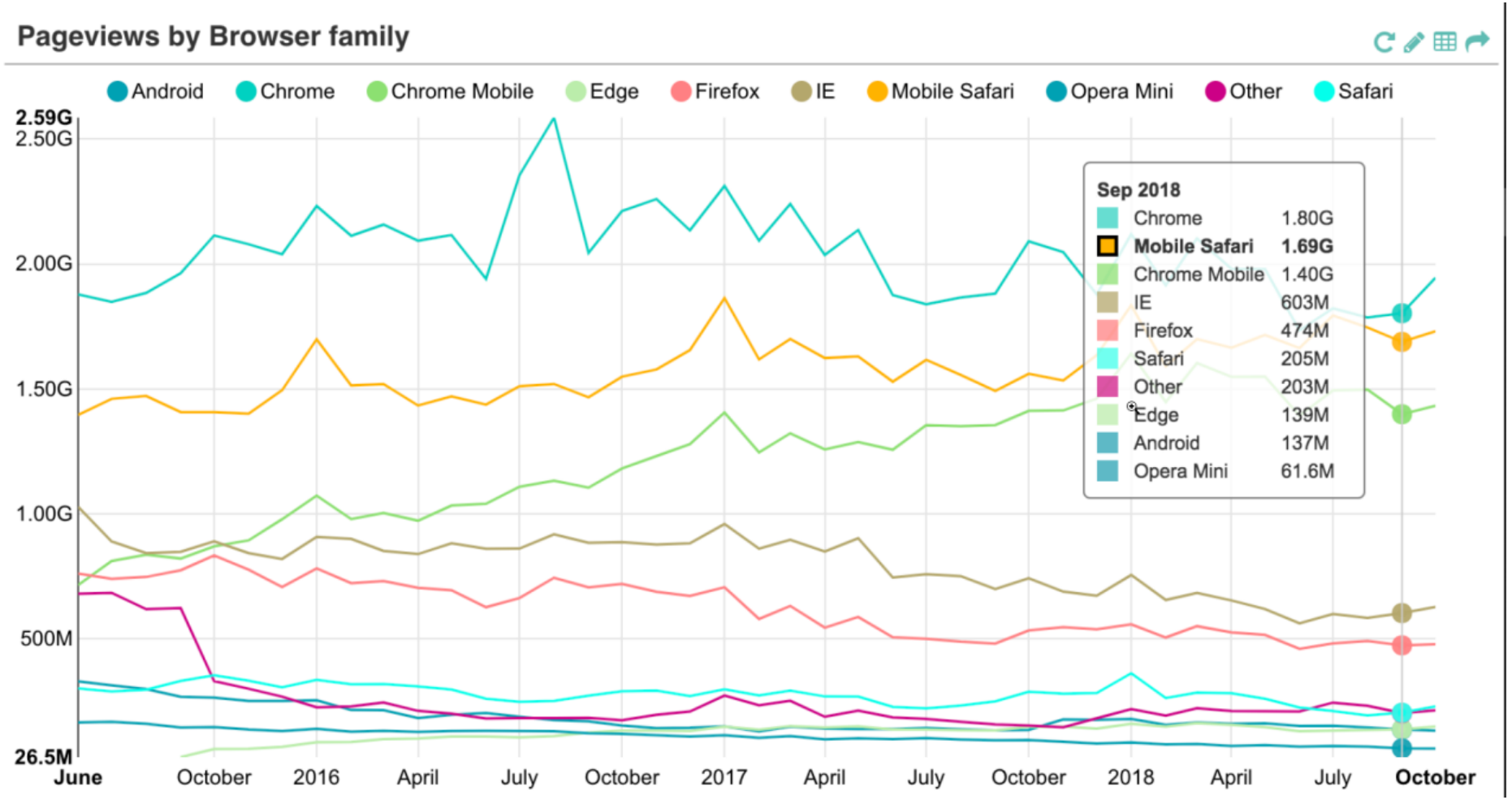
Περιεχόμενα

1. Ανάλυση (Μαθησιακών) Δεδομένων
2. Python-based Analytics

1. Τι είναι η ανάλυση δεδομένων

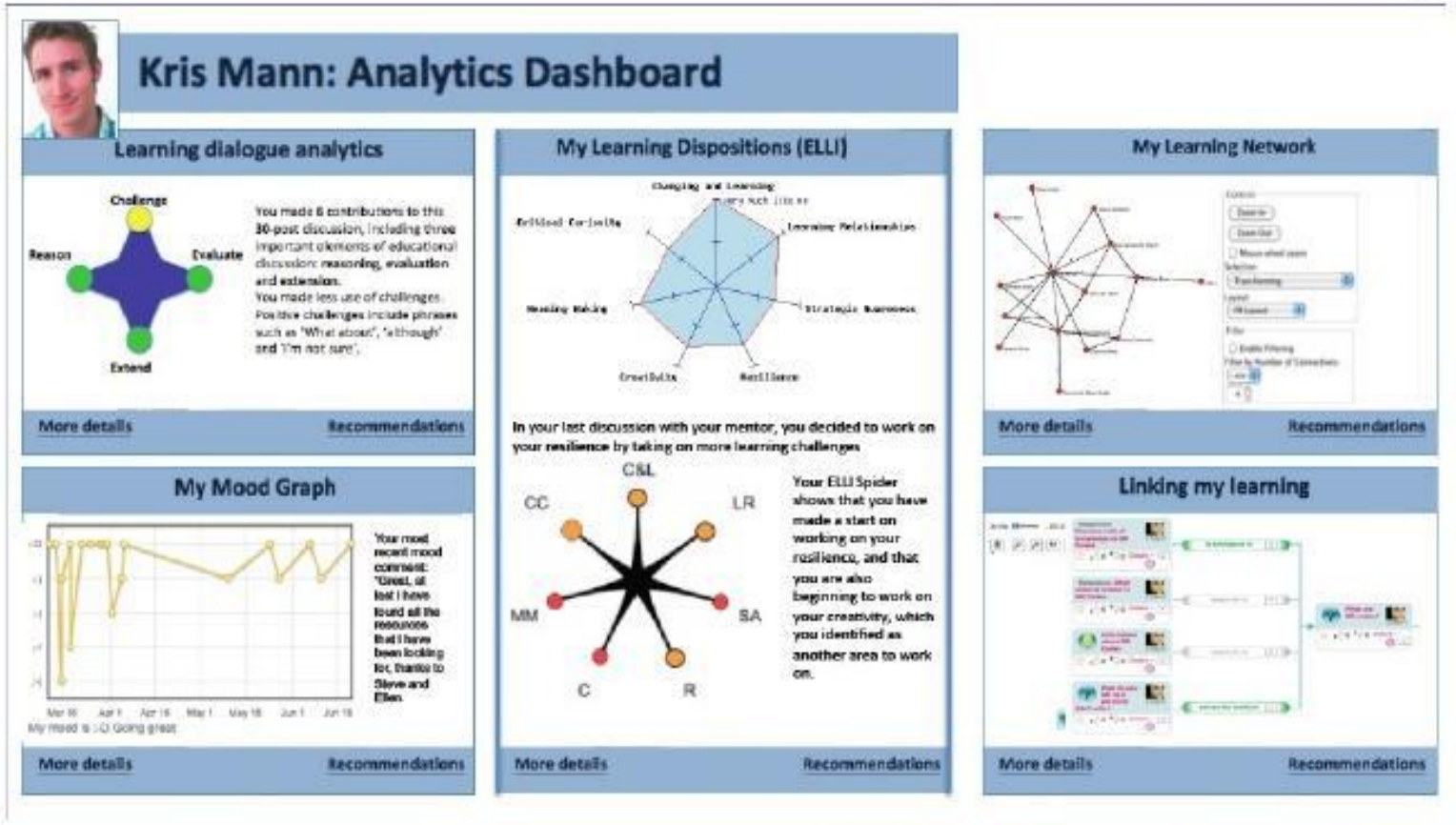
- Εξάγει σχέσεις και πρότυπα και απαντά σε ερωτήματα:

Τι συνέβη, πως και γιατί? Τι συμβαίνει τώρα? Προβλέψεις.



Τεχνικές Ανάλυσης Δεδομένων

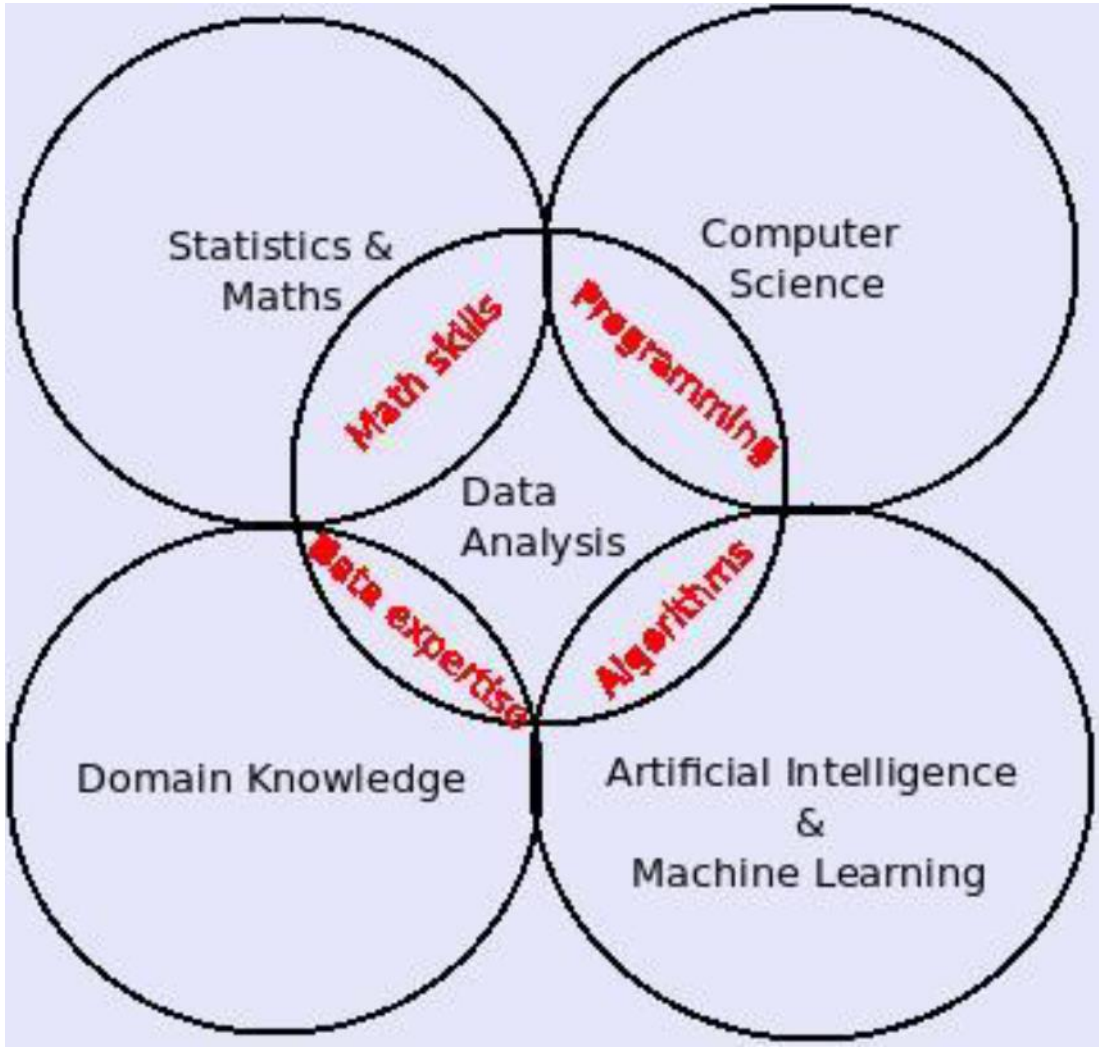
Different dashboard views



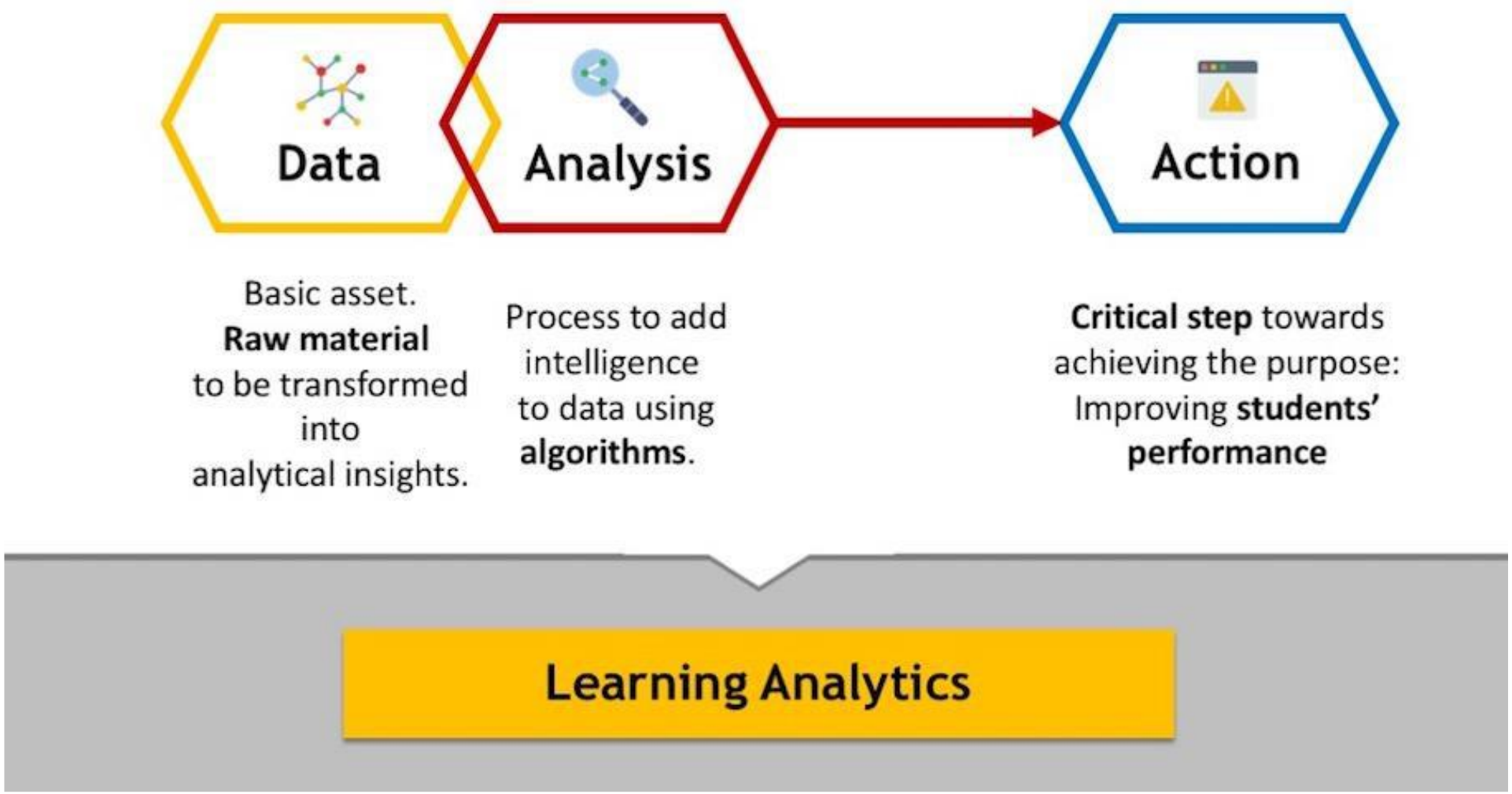
SocialLearn

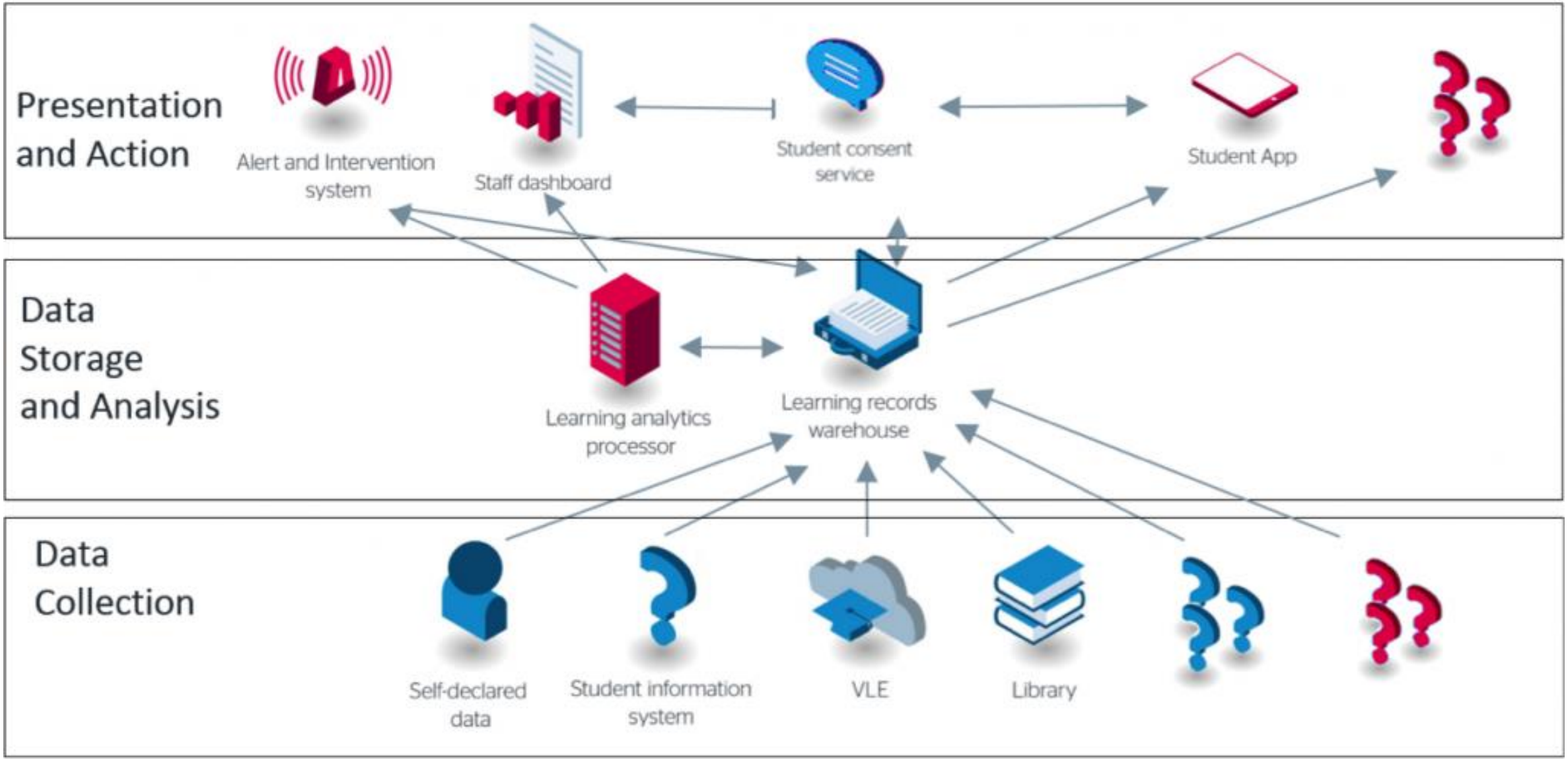
2. Ανάλυση Δεδομένων Μάθησης

- Επστημολογία (**What** are we measuring, **How**)
- Παιδαγωγική (**Why**, for **Whom**)
- Αξιολόγηση (**Where** & **When**)

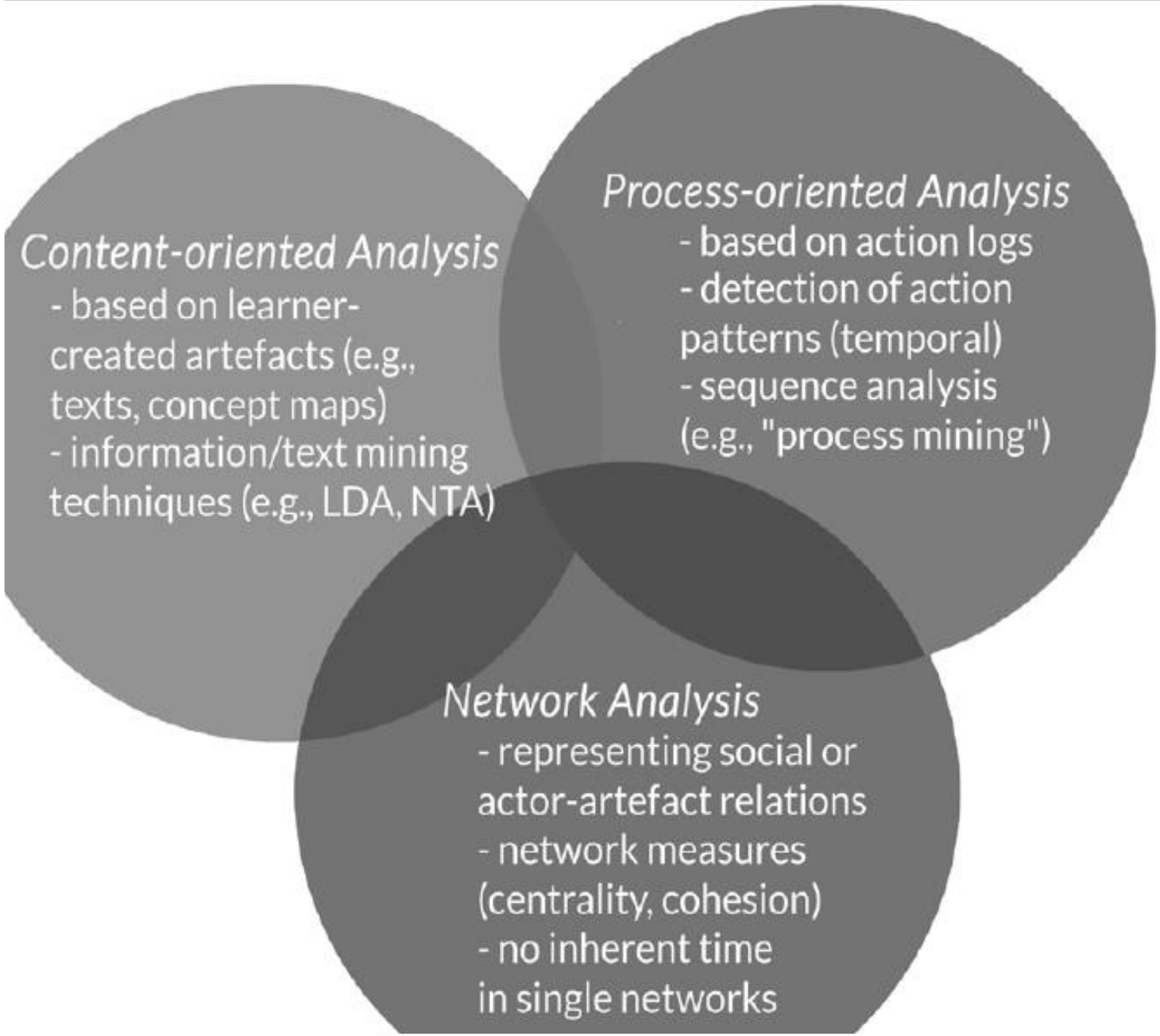


Τι είναι Ανάλυση Δεδομένων Μάθησης

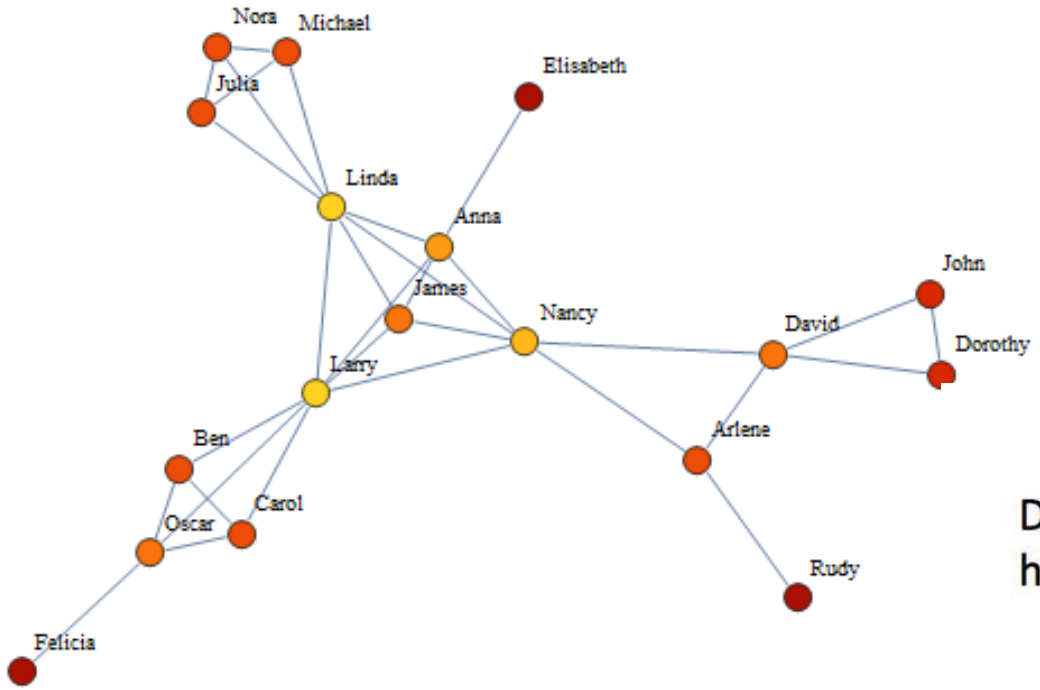


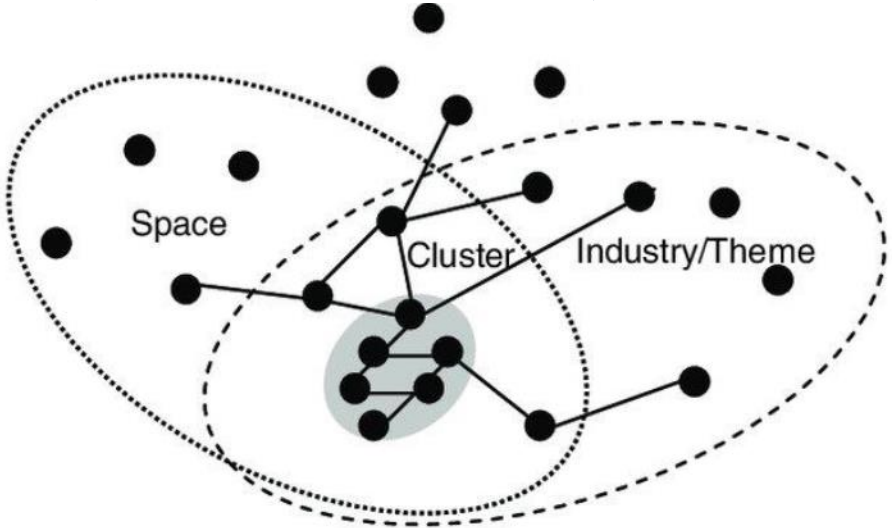

Presentation and Action
Alert and Intervention system
Staff dashboard
Student consent service
Student App
Data Storage and Analysis
Learning analytics processor
Learning records warehouse
Data Collection
Self-declared data
Student information system
VLE
Library

Υπολογιστικές μέθοδοι

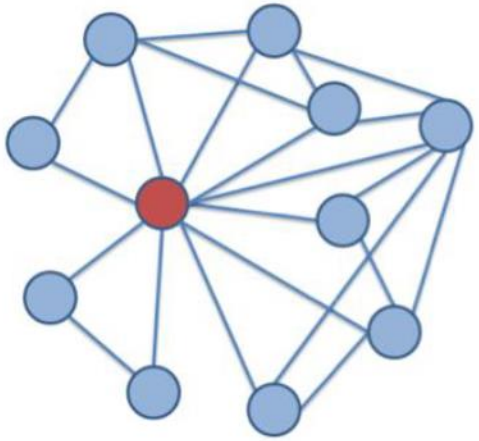


Υπολογιστικές μέθοδοι (Network Analytic methods)

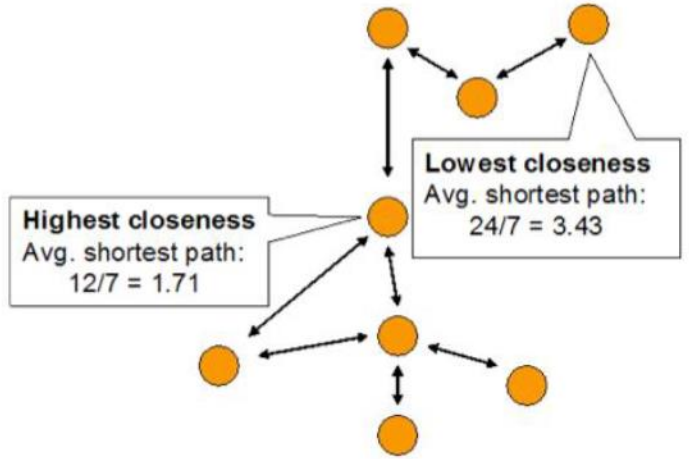




Degree centrality:
highest number of edges

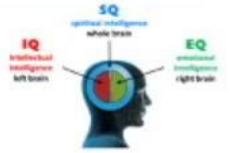


Closeness centrality:
lowest average shortest distance to all other nodes



- Constructs

Concepts	Variables	Constructs
Empirical abstraction to classify natural phenomenon	When concepts can be assigned with values, they can be manipulated as **variables**, so that their relationship can be examined	Some concepts are observable and easily distinguished from others. But other concepts are less tangible, and can be defined only by inference. **Construct** are invented names for abstract variables that cannot be measured directly, but by measuring relevant correlated behaviour or observable.

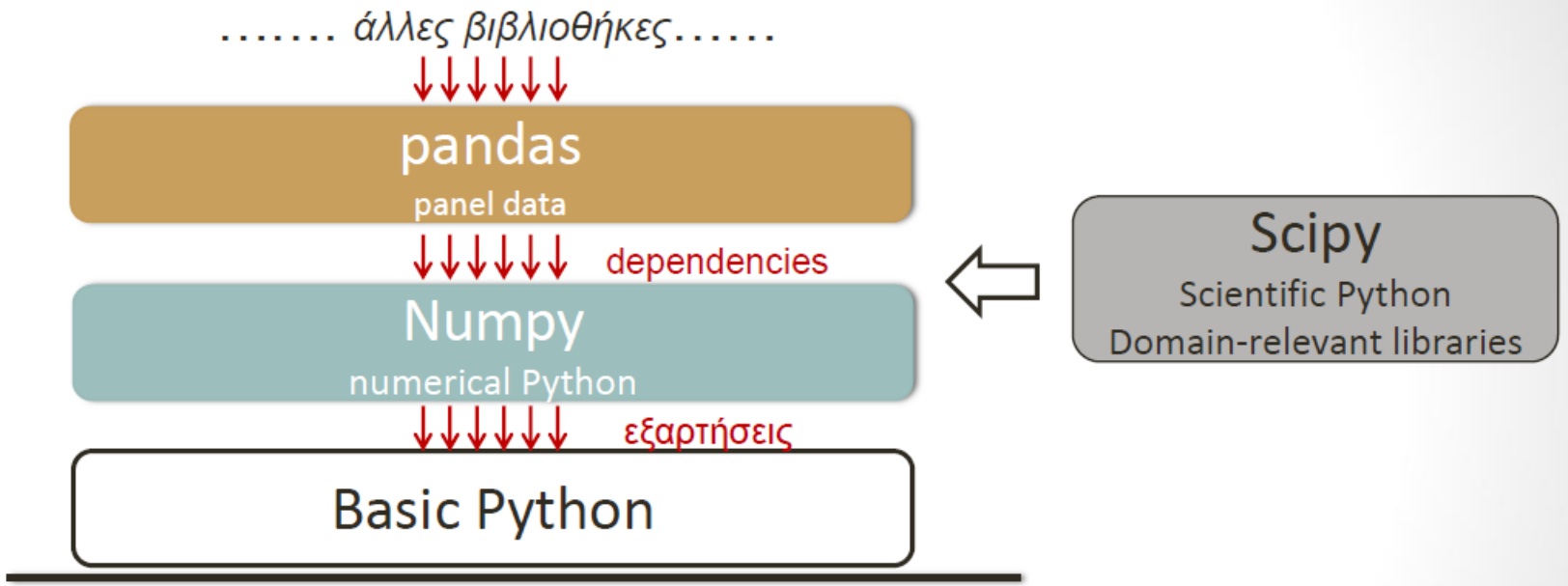


3. Python-based analytics

- Anaconda for Python 3.7 http://bit.ly/2JAPJu4
- http://pytolearn.csd.auth.gr/ (scientific programming)
- Jupyter Notebook

- Επεξηγηματική & Προγνωστική
- Python Βιβλιοθήκες numpy, scipy, pandas, matplotlib, statsmodels, scikit.

Οικοσύστημα Python



Roots: Python class hierarchy

Numpy

- ***List:*** Reduced efficiency is *not a problem when data size is relatively small*. But, it really turns to be a restriction when dealing with big data. This is also the reason for introducing a more efficient object, namely **ndarray.**

- Η βιβλιοθήκη **numpy** προσφέρει τη δομή **Πίνακας** (**ndarray**): Αποδοτική ομοιογενής δομή δεδομένων με υψηλή ταχύτητα επεξεργασίας. Η numpy προσφέρει καλύτερη διαχείριση περισσότερων τύπων.

Numpy

NumPy is the fundamental package for **scientific computing** with Python. It contains among other things:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities
- Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

- Numerical Python
- Scientific programming
- Big Data analysis
- Array oriented computing
- ***import numpy as np***

- *Διανυσματοποίηση*

Matplotlib

- **matplotlib,** (Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers, and for graphical user interface toolkits.)
- pyplot module (very similar to *Matlab*)

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
x = np.linspace(0, 2*np.pi, 100);
y = np.sin(x) plt.plot(x, y)
```

Pandas

- **Υψηλής ευελιξίας και αποδοτικότητας βιβλιοθήκη** για την οργάνωση & ανάλυση δεδομένων
- Προσφέρει **φιλικές** προς το χρήστη **δομές δεδομένων με ονόματα**, (βασισμένες στα λεξικά της Python και τους πίνακες της Numpy)
- Οργάνωση δεδομένων σε 2 βασικά αντικείμενα: **DataFrame** & **Series**
- Πλεονεκτήματα:

Α) **Εισαγωγή** δεδομένων από διάφορα αρχεία (***xls, csv, text, json***, κ.ά.)

Β) **Οργάνωση** δεδομένων σε δομές-πίνακα (table-like) με **ετικέτες**

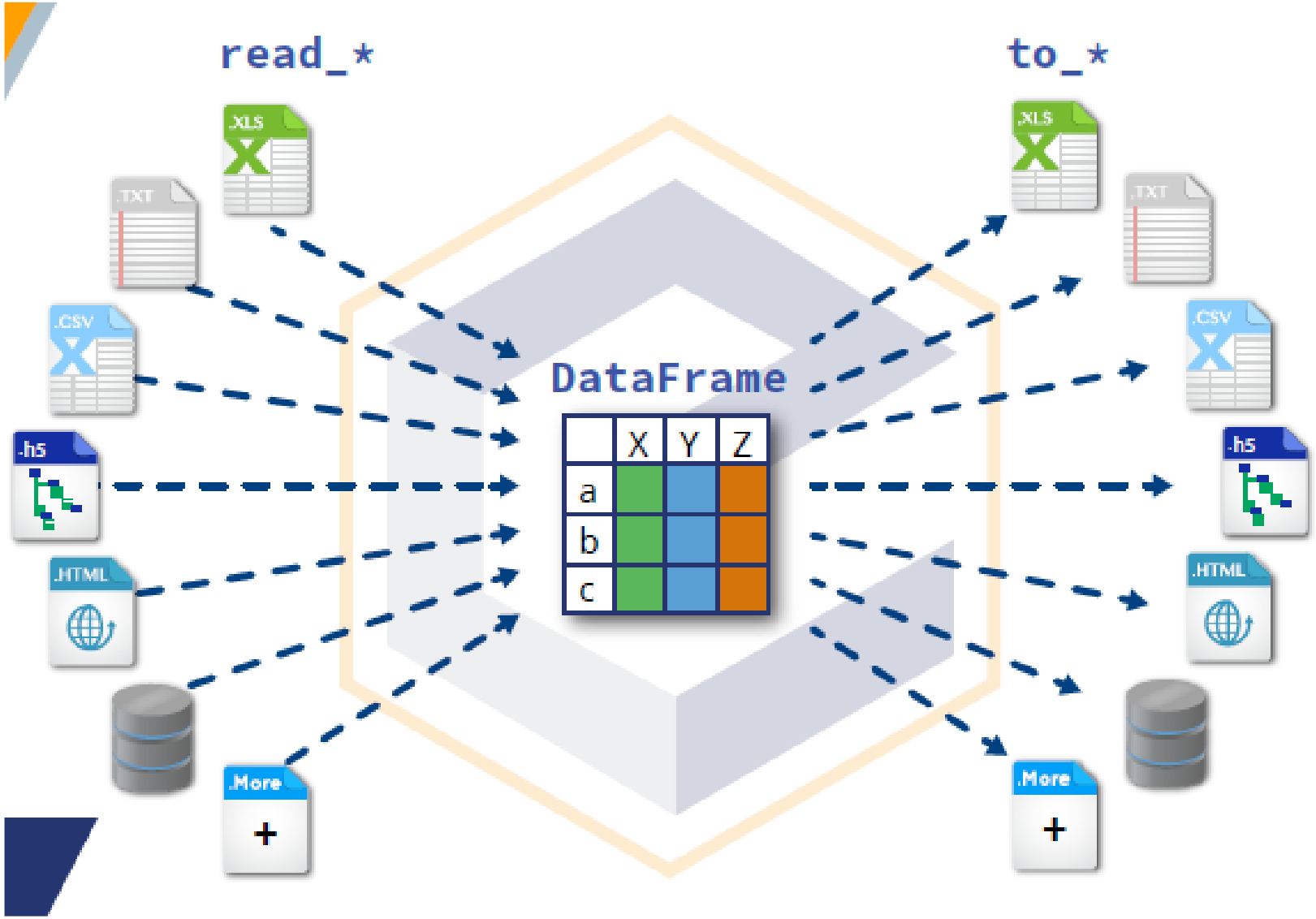
Γ) Υψηλή **ευελιξία** στην ανάλυση δεδομένων

Δ) **Συνεργασία με Scipy** (και άλλες βιβλιοθήκες) για εφαρμογή μεθόδων επεξεργασίας δεδομένων

Pandas

- *import pandas as pd*
- Frame (read_ & to_)

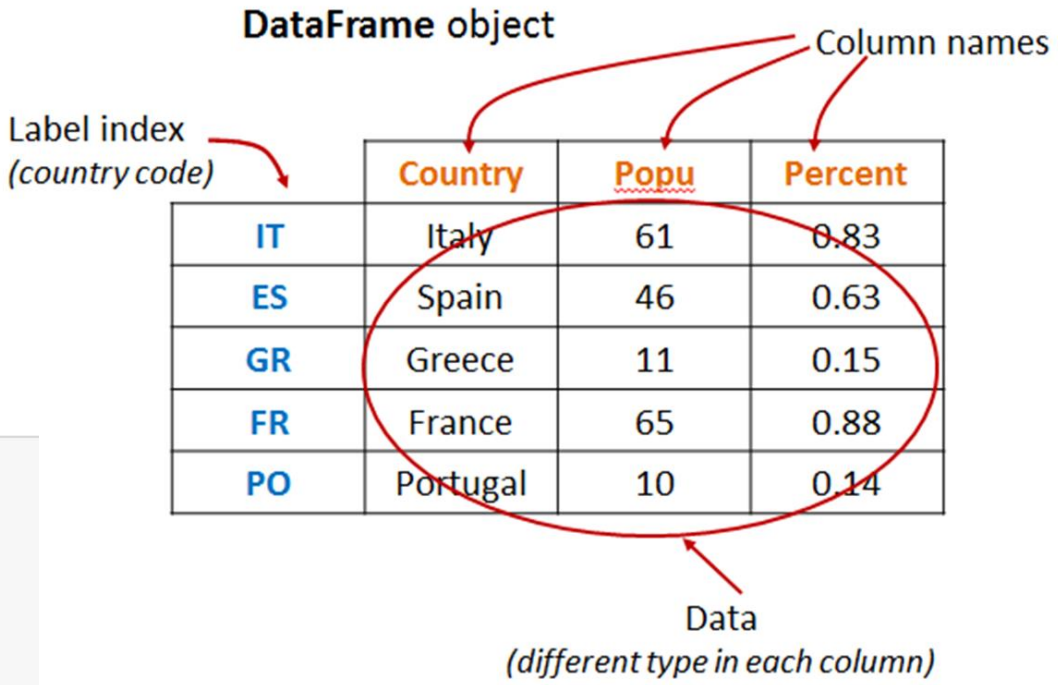
- read_csv
- read_excel
- read_json



DataFrame (Pandas)

```
data = {'Country': ['Cyprus','Germany','Denmark','Spain','Greece'],
        'Population': [855, 82800, 5749, 46529, 10757],
        'Percent': [0.17,16.18,1.12,9.09,2.10]}

df = pd.DataFrame(data,
                  columns = ['Country', 'Population', 'Percent'],
                  index=['CYP', 'DEU', 'DNK', 'ESP', 'GRC'])
df
```




```
# Εξοδος σε αρχείο xls

dfout = df[['Country']]
dfout

dfout.to_excel("output.xlsx",sheet_name='Country_names')
```

Series (Pandas)

```
import pandas as pd

s = pd.Series([10,20,30,40,50], dtype='int')
print(s)
print(s.index, s.values)
```

```
import numpy as np
import pandas as pd

s = pd.Series(np.array([chr(i) for i in range(65,70)]))
print(s)
print(s.index, s.values)
```

```
# Series

s = df['Population']
s
type(s)
```

```
CYP      855
DEU    82800
DNK     5749
ESP    46529
GRC    10757
Name: Population, dtype: int64

pandas.core.series.Series
```

Έλεγχος υποθέσεων (Hypothesis testing)

- Research design:
- Quantitative paradigm (rigorous testing/predicting methods)
- Qualitative paradigm (deeper understanding)
- Mixed-method paradigm

- Experimental design

Κατανομές (Distributions)

import scipy.stats as stats # Imports the entire scipy.stats (all distributions)

from scipy.stats import **norm** # imports only the normal distribution

from scipy.stats import **t** # imports only the Student's t distribution (see t-test)

from scipy.stats import **f** # imports only the Fisher's f distribution (F-test)

norm.pdf(x) = exp(-x**2/2)/sqrt(2*pi)

median(), mean(), std(), var(), stats()

Statistical controls

- Descriptive statistics:

mean(), median(), std(), var(), min(), max(), skew(), kurt()

data.corr(method='spearman')

data.cov()

Test for differences between 2 groups

- t-test (independent samples)

stats.shapiro() #Shapiro-Wilk test for normality criterion

stats.levene() #Variance criterion

stats.ttest_ind()

Stats.mannwhitneyu() #non parametric… of the t-test

- Paired t-test
- One sample t-test

Paired t-test

- t-test (related samples)

stats.shapiro() #Shapiro-Wilk test for normality criterion

stats.levene() #Variance criterion

stats.ttest_rel()

stats.wilcoxon() #non parametric… of the paired t-test

One sample t-test

- t-test (related samples)

stats.shapiro() #Shapiro-Wilk test for normality criterion

stats.ttest_1samp()

Test for differences between 3 or more groups

- One-way ANOVA

stats.shapiro()

stats.levene()

F, p = stats.f_oneway(dC, dT1, dT2)

tukeyhsd()

Building models in Python (supervised learning)

- Γραμμική Παλινδρόμηση (Linear Regression)

Statsmodels

Scikit-learn

Scipy library

```
import statsmodels.api as sm

X = df["RM"]
y = target["MEDV"]

# Note the difference in argument order
model = sm.OLS(y, X).fit()
predictions = model.predict(X) # make the predictions by the model

# Print out the statistics
model.summary()
```

Building models in Python (unsupervised learning)

```
# import statements
from sklearn.datasets import make_blobs
import numpy as np
import matplotlib.pyplot as plt

# create blobs
data = make_blobs(n_samples=200, n_features=2, centers=4,
cluster_std=1.6, random_state=50)

# create np array for data points
points = data[0]

# create scatter plot
plt.scatter(data[0][:,0], data[0][:,1], c=data[1], cmap='viridis')
plt.xlim(-15,15)
plt.ylim(-15,15)
```

```
# import KMeans
from sklearn.cluster import KMeans

# create kmeans object
kmeans = KMeans(n_clusters=4)

# fit kmeans object to data
kmeans.fit(points)

# print location of clusters learned by kmeans obje
print(kmeans.cluster_centers_)

# save new clusters for chart
y_km = kmeans.fit_predict(points)
```

Τεχνητή Νοημοσύνη με Python (text mining)

e.g. sentiment analysis, text classification

NLTK

import nltk
sent_tokenize / word_tokenize #tokenization
FreqDist() #frequency distribution
top words = noise to the text
Lexicon normalization
Part Of Speech Tagging

Χρήσιμοι σύνδεσμοι

- **http://pytolearn.csd.auth.gr/**
- https://analytics.jiscinvolve.org/wp/ (JISC)
- https://solaresearch.org/
- https://lak19.solaresearch.org/
- https://jedm.educationaldatamining.org/index.php/JEDM
- http://bit.ly/2Qcsg4s
- http://bit.ly/2QdonMI

Take Away

- …

Σας ευχαριστώ για την προσοχή σας!

http://blogs.sch.gr/dtzimas