

Ένωση Πληροφορικών Ελλάδας



Το Α.Ι. ήταν πάντα εδώ:
Απλά εργαλεία για σύνθετα προβλήματα

Χάρης Γεωργίου (MSc, PhD)

Ένωση Πληροφορικών Ελλάδας

Στόχοι:

- Πρώτος “καθολικός” φορέας εκπροσώπησης πτυχιούχων Πληροφορικής.
- Αρμόδιος φορέας εκπροσώπησης επαγγελματιών Πληροφορικής.
- Αρμόδιος επιστημονικός “συμβουλευτικός” φορέας για το Δημόσιο.
- Αρωγός της Εθνικής Ψηφιακής Στρατηγικής & Παιδείας της χώρας.

<https://www.epe.org.gr>



Τομείς παρέμβασης

Ποιοι είναι οι κύριοι τομείς παρεμβάσεων της ΕΠΕ;

- 1 Εθνική Ψηφιακή Στρατηγική & Οικονομία
- 2 Εργασιακά (ΤΠΕ), Δημόσιος & ιδιωτικός τομέας
- 3 Παιδεία (Α', Β', Γ')
- 4 Έρευνα & Τεχνολογία
- 5 Έργα & υπηρεσίες ΤΠΕ
- 6 Ασφάλεια συστημάτων & δεδομένων
- 7 Ανοικτά συστήματα & πρότυπα
- 8 Χρήση ΕΛ/ΛΑΚ
- 9 Πνευματικά δικαιώματα
- 10 Κώδικας Δεοντολογίας (ΤΠΕ)
- 11 Κοινωνική μέριμνα (ICT4D)





Harris Georgiou (MSc, PhD) – <https://github.com/xgeorgio/info>

- R&D: Associate post-doc researcher and lecturer with the University Athens (NKUA) and University of Piraeus (UniPi)
- Consultant in Medical Imaging, Machine Learning, Data Analytics, Signal Processing, Process Optimization, Dynamic Systems, Complexity & Emergent A.I., Game Theory
- HRTA member since 2009, LEAR / scientific advisor
- HRTA field operator (USAR, scuba diver)
- Wilderness first aid, paediatric (child/infant)
- Humanitarian aid & disaster relief in Ghana, Lesvos, Piraeus
- Support of unaccomp. minors, teacher in community schools
- Streetwork training, psychological first aid & victim support
- 2+4 books, 170+ scientific papers/articles (and 5 marathons)

Επισκόπηση

- Περιεχόμενα:
 - Τι είναι η Μηχανική Μάθηση και η Αναλυτική Δεδομένων (ML/DA).
 - Προπαρασκευή δεδομένων (pre-processing), είδη προβλημάτων ML/DA.
 - Επεξεργασία αρχείων, command-line tools, SQL, αρχεία κειμένου (imports).
 - Προηγμένα εργαλεία στην κονσόλα (command-line), σε IDE, στο cloud.
- Αναφορές:
 - «Εισαγωγή στη Μηχανική Μάθηση και στην Αναλυτική Δεδομένων», Χ. Γεωργίου, Α' κύκλος ανοικτών μαθημάτων ΕΠΕ – <https://youtu.be/mlU4SvyfRqA>
 - «Εισαγωγή στην Αναλυτική Δεδομένων με τη γλώσσα R», Χ. Γεωργίου, Β' κύκλος ανοικτών μαθημάτων ΕΠΕ – <https://youtu.be/7aaNnXxyzgl>
 - Dunham: Data Mining – Introductory and Advanced Topics. Prentice Hall, 2003.
 - Tan, Steinbach, Kumar: Introduction to Data Mining. Addison Wesley, 2006.

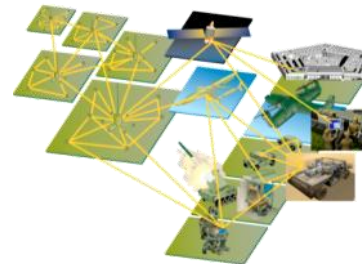
Μέρος I: Οργάνωση της εργασίας (ML/DA)

1. Μηχανική Μάθηση και Αναλυτική Δεδομένων (ML/DA).
2. Προεπεξεργασία / προπαρασκευή δεδομένων.
3. Τι είναι το ML/DA pipeline;
4. Τι προβλήματα έχουμε;



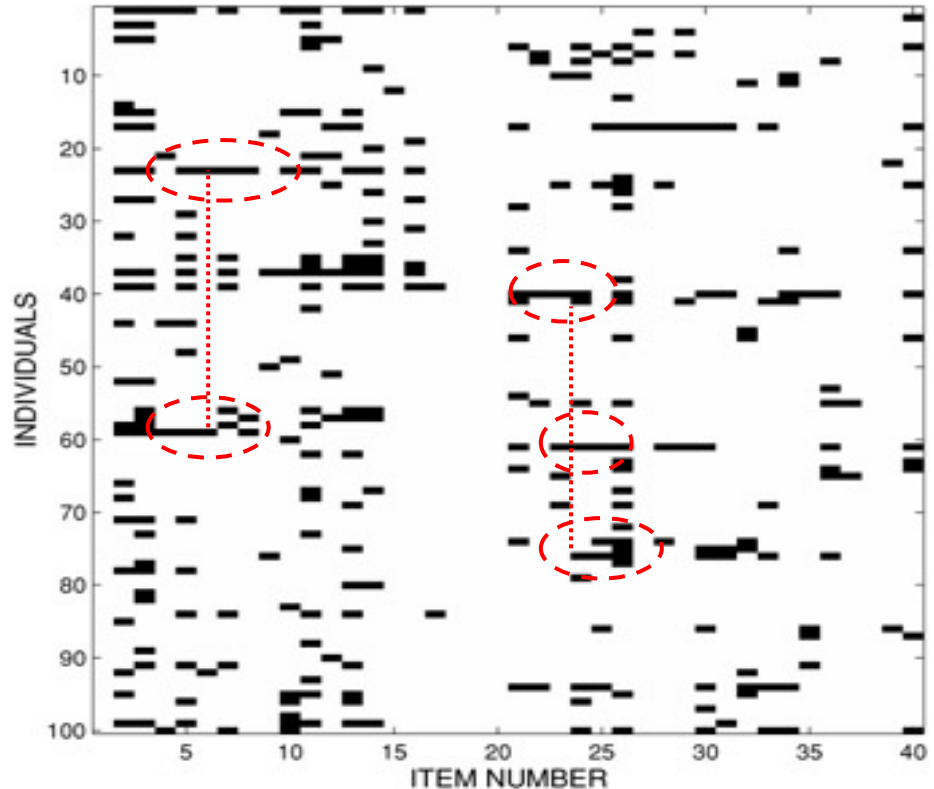
Δεδομένα παντού ...

- Παράγονται όλο και περισσότερα δεδομένα:
 - Τραπεζικά, τηλεπικοινωνιακά, ...
 - Επιστημονικά δεδομένα: αστρονομικά, βιολογικά κλπ.
 - Κείμενα στο web κ.α.
- Αποθηκεύονται όλο και περισσότερα δεδομένα:
 - Γρήγορη / φθηνή τεχνολογία αποθήκευσης
 - Ικανά ΣΔΒΔ για μεγάλες ΒΔ



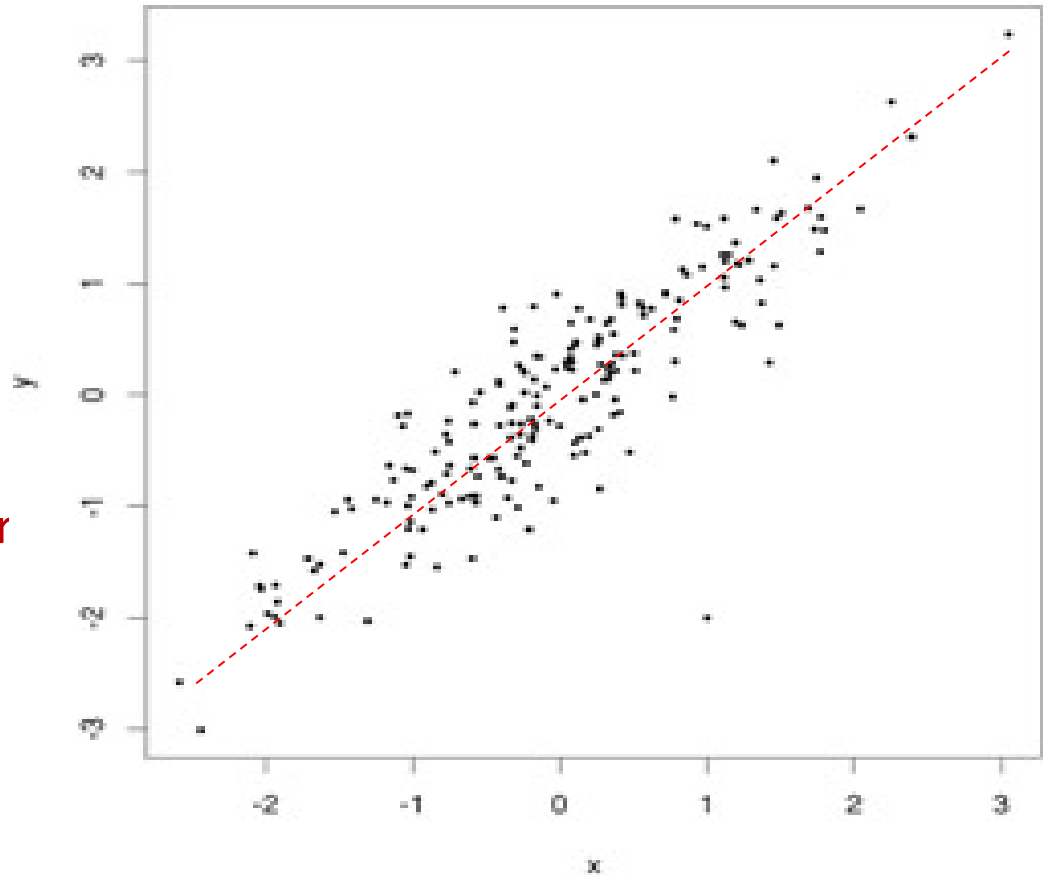
Δεδομένα...

- 2-D πίνακας: πελάτες (γραμμές) προς προϊόντα (στήλες)
- Στόχος ανάλυσης: (εύκολος;) εντοπισμός συλλογικής συμπεριφοράς
- Αποτέλεσμα: Αναγνώριση ισχυρών εξαρτήσεων μεταξύ μεταβλητών
- Στον πίνακα: «συνεμφάνιση» τιμών ή γεγονότων

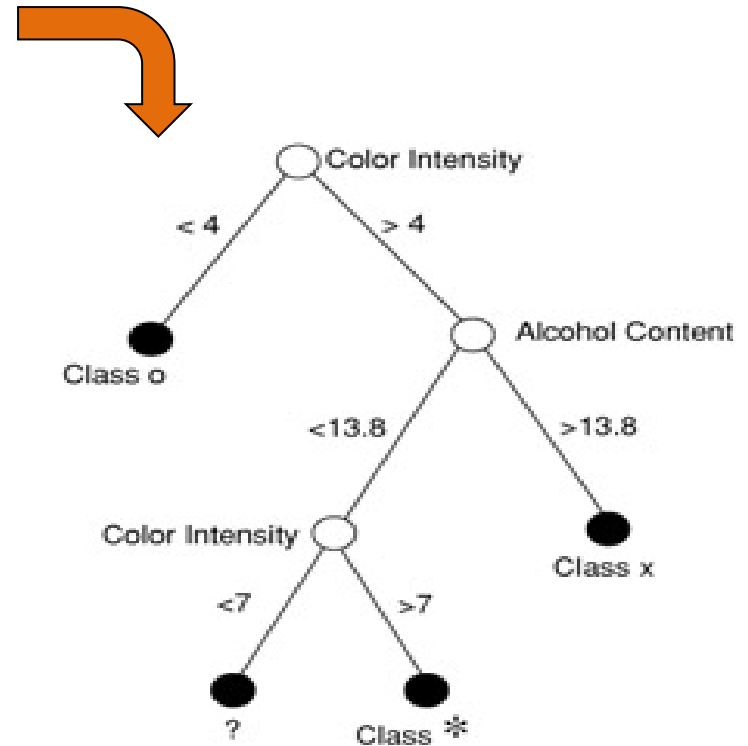
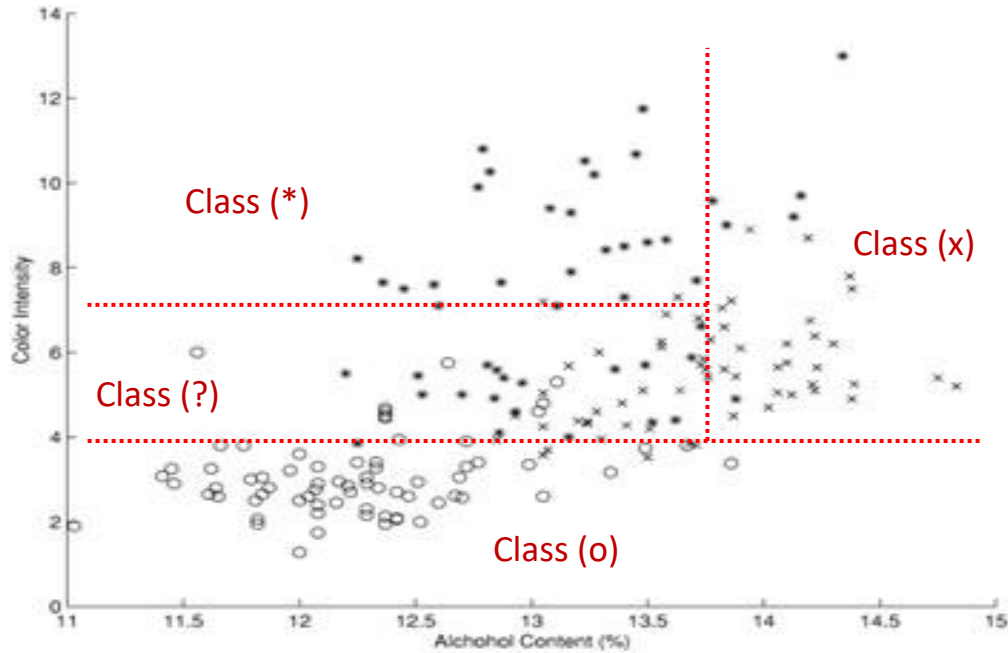


Δεδομένα...

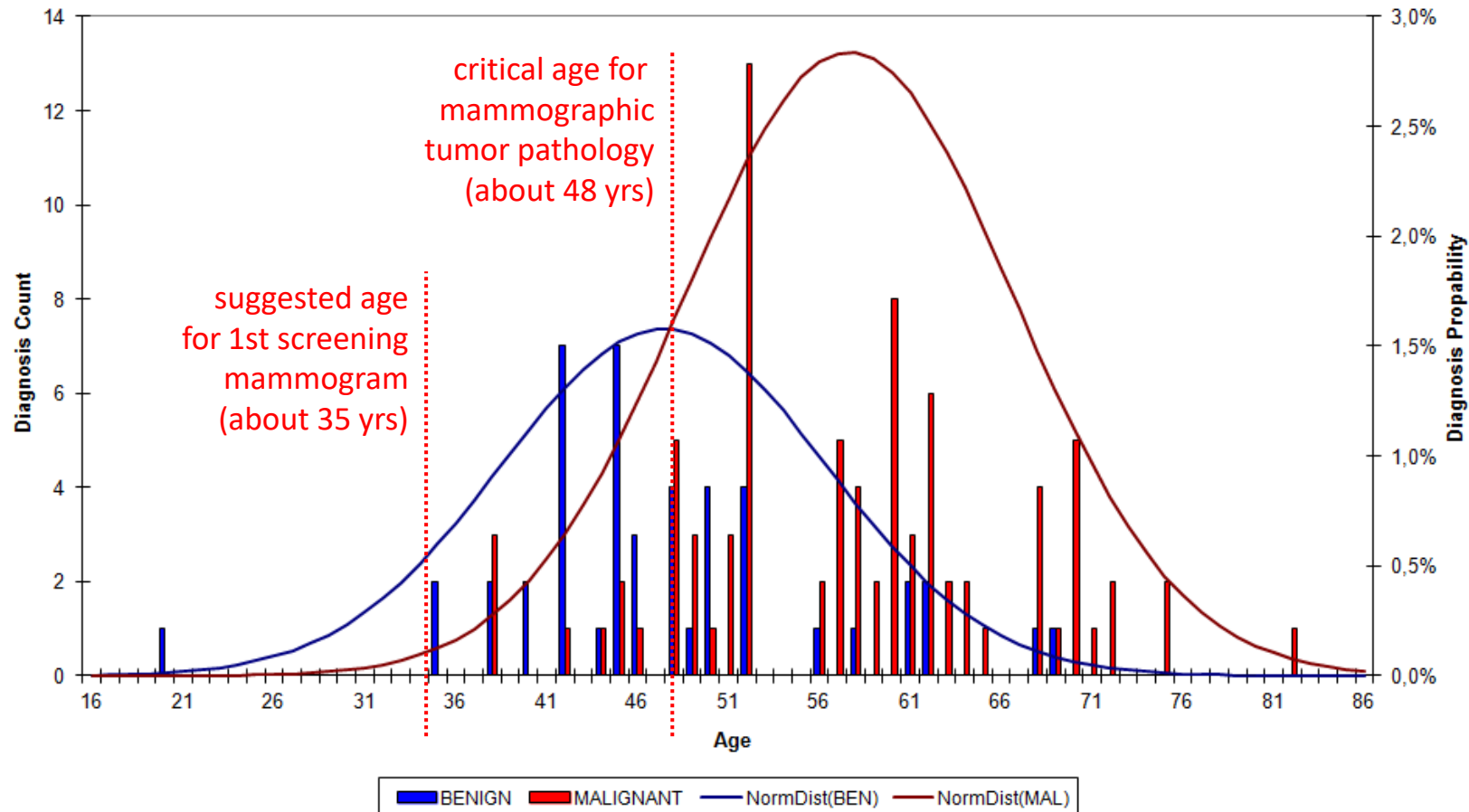
- 2-D διάνυσμα (x,y)
- Στόχος ανάλυσης: πρόβλεψη «αναμενόμενης» εξόδου για οποιαδήποτε είσοδο
- Εναλλακτικά: εντοπισμός «ακραίας συμπεριφοράς»
- Γραμμική προσαρμογή (linear regression): η «καλύτερη» ευθεία που «εξηγεί» τα δεδομένα



Από τα δεδομένα στη γνώση...



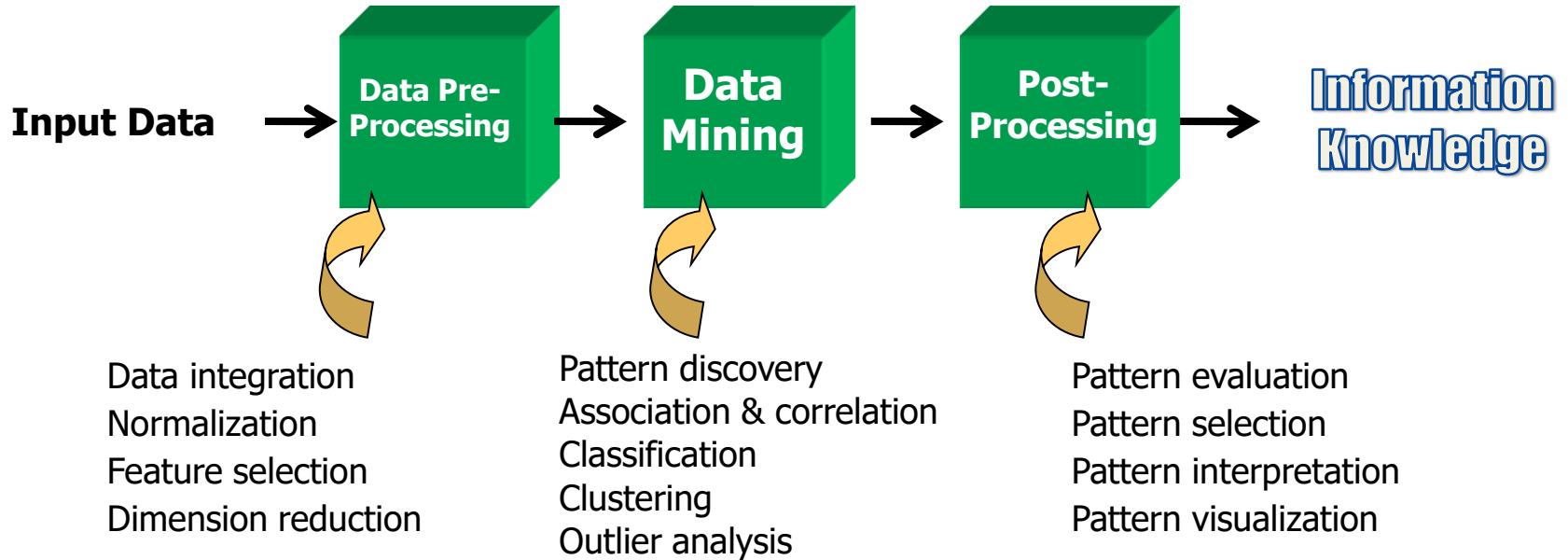
Age Distributions vs Benign/Malignant



Η “σκάλα” της διαδικασίας KDD



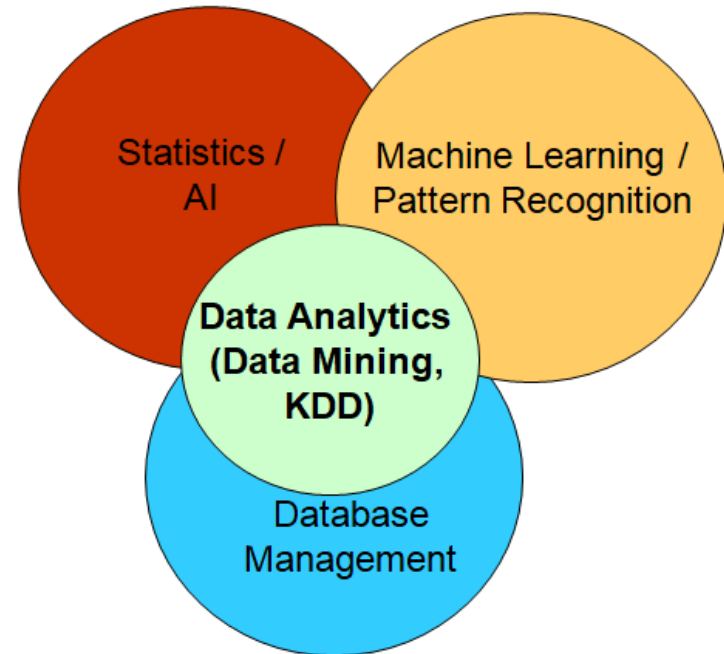
Εναλλακτικά...



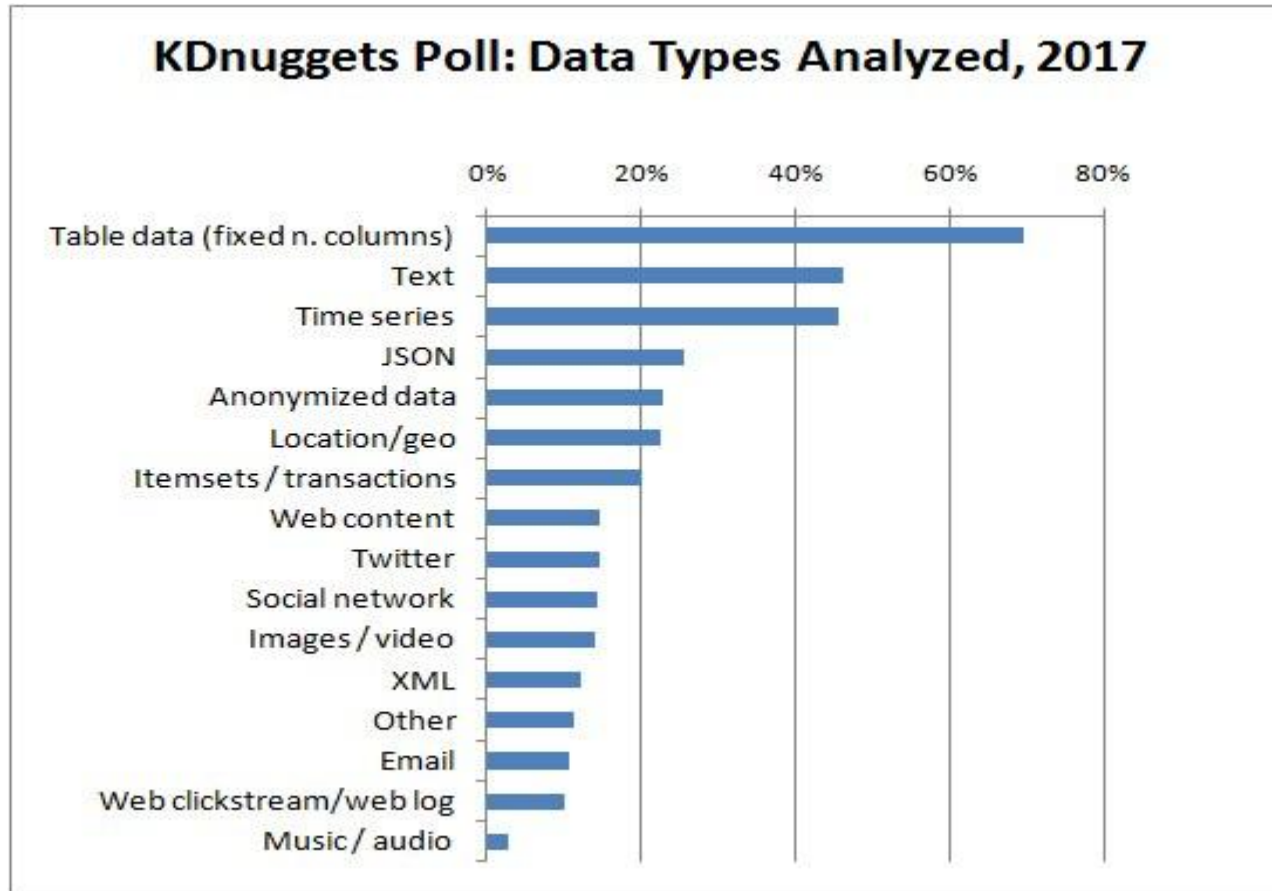
- Τυπική θεώρηση από τη σκοπιά της Στατιστικής και της Μηχανικής Μάθησης

Σχετικά επιστημονικά πεδία

- Στατιστική / «Τεχνητή Νοημοσύνη», Μηχανική Μάθηση / Αναγνώριση Προτύπων, Διαχείριση Βάσεων Δεδομένων
- Οι παραδοσιακές τεχνικές επεξεργασίας δεδομένων που μας προσφέρουν αυτές οι επιστημονικές περιοχές μπορεί να είναι ανεφάρμοστες λόγω:
 - του μεγάλου όγκου,
 - των πολλών διαστάσεων,
 - της ετερογένειας των δεδομένων,
 - των απαιτήσεων επεξεργασίας,
 - ...

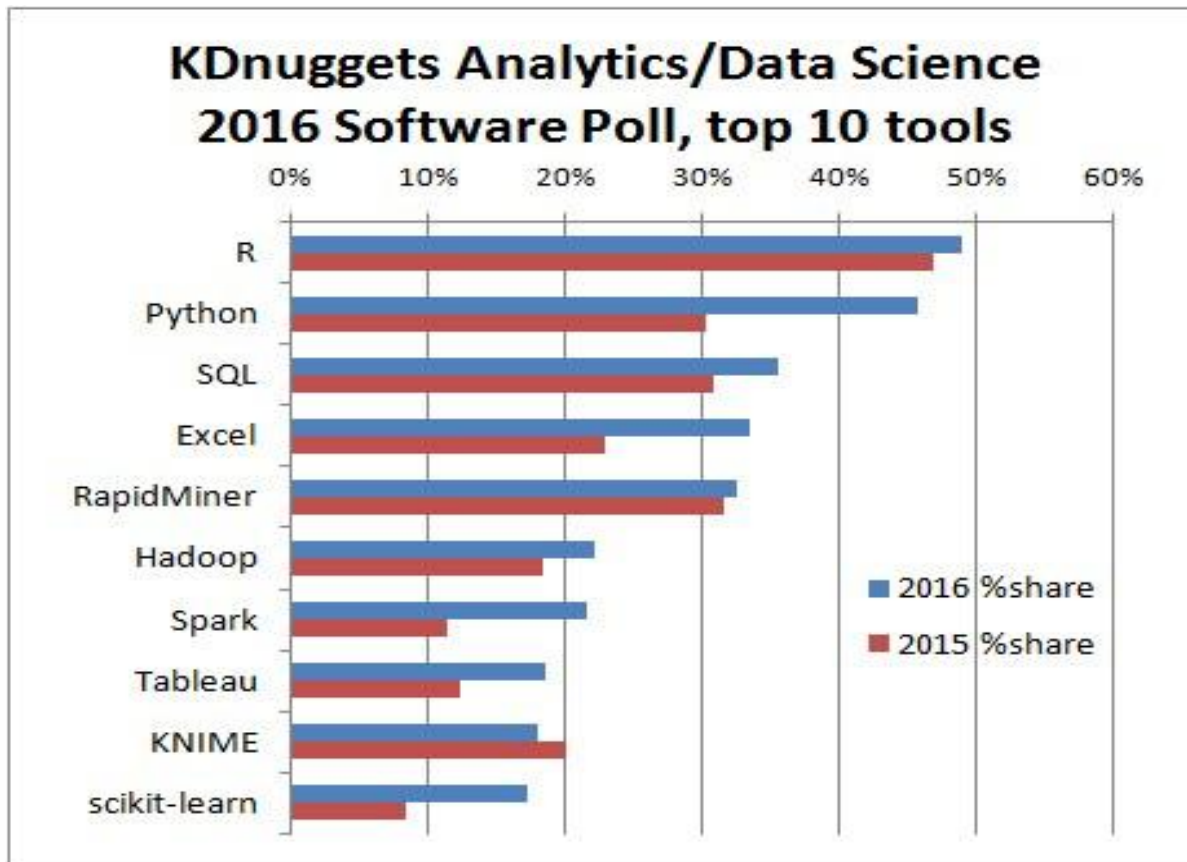


Τι δεδομένα αναλύουμε συνήθως ...



πηγή:
kdnuggets.com

Με τι λογισμικό ...



πηγή:
kdnuggets.com

Βασικές έννοιες & γνώσεις

- **Γνωριμία με τα δεδομένα**
 - Προπαρασκευή δεδομένων
- **Τεχνικές και αλγόριθμοι**
 - Κατηγοριοποίηση / ταξινόμηση (classification)
 - Ανάλυση συστάδων (cluster analysis)
 - Μοντέλα προσαρμογής (regression)
 - Εξόρυξη συχνών προτύπων (frequent pattern mining)
- **Παραγόμενα αποτελέσματα**
 - Προτυποποίηση κώδικα
 - Βελτιστοποίηση διαδικασιών (process optimization)
 - Ενσωματωμένα συστήματα (IoT, edge, ...)



Μέρος II: Επιλογή απλών εργαλείων

1. Διαχείριση & βασική επεξεργασία αρχείων.
2. AWK ως εργαλείο προγραμματισμού.
3. SQL για in-server επεξεργασία δεδομένων.
4. Εισαγωγή αρχείων κειμένου (imports).



CMD: tar, gzip, split/merge

```
tar -czvf name-of-archive.tar.gz /path/to/directory-or-file
```

Here's what those switches actually mean:

- -c: Create an archive.
- -z: Compress the archive.
- -v: Display progress in verbose mode. The v is always visible.
- -f: Allows you to specify the filename of the archive.

Gzip

The most basic way to use gzip to compress a file is:

```
% gzip filename
```

This will create a compressed version of *filename.gz* in the same directory as the original file. To specify a different process type:

```
% gzip -d filename.gz or % gunzip filename.gz
```

Which removes *filename.gz* and replaces it with the original file.

To prevent gzip from overwriting a file that you already have, you can use the `-k` switch:

```
(i.e. % gzip -c filename > compressedfile.gz -k)
```

```
split -b 1M bigfile.txt bigfile-part-
```

In the above command, `-b` is used to specify the maximum size of each part, which is 1 MB in this case.

`bigfile.txt` is the input file, and `bigfile-part-` is the prefix that will be used for the output files.

The split command will create multiple output files, each with the prefix `bigfile-part-` followed by a two-letter suffix. For example, the first file will be named `bigfile-part-aa`, the second file will be named `bigfile-part-ab`, and so on.

File Concatenation

To concatenate multiple files into a single file, you can use the `cat` command along with the file names in the desired order. For example, let's say we have three files named `file1.txt`, `file2.txt`, and `file3.txt` that we want to concatenate into a single file named `allfiles.txt`.

Here's how we can use the `cat` command to achieve that:

```
...
```

```
cat file1.txt file2.txt file3.txt > allfiles.txt
```

In the above command, `file1.txt`, `file2.txt`, and `file3.txt` are the input files that will be concatenated into a single file. The `>` operator is used to redirect the output to a new file named `allfiles.txt`.

CMD: sed, find, grep, awk

NAME
awk - pattern scanning and processing language

SYNOPSIS
awk [-F *sepstring*] [-v *assignment*]... *program* [*argument*...]
awk [-F *sepstring*] -f *progfile* [-f *progfile*]... [-v *assignment*]... [*argument*...]

DESCRIPTION

OPTIONS

OPERANDS

STDIN

INPUT FILES

ENVIRONMENT VARIABLES

LANG	LC_MESSAGES
LC_ALL	LC_NUMERIC
LC_COLLATE	NLSPATH
LC_CTYPE	PATH

ASYNCHRONOUS EVENTS

STDOUT

STDERR

OUTPUT FILES

EXTENDED DESCRIPTION

Overall Program Structure
pattern { *action* }

SED(1) User Commands SED(1)

NAME
sed - stream editor for filtering and transforming text

SYNOPSIS
sed [OPTION]... [-e *script-only-if-no-other-*

just one task. In fact, one of the options in the `find` command on whatever results `find` returns. `find` is used to search for a file by *content* rather than

```
-exec grep -Hi penguin {} \;
```

like penguins.
penguins are fun.

```
--debug  
annotate program execution  
  
-e script, --expression=script  
add the script to the commands to be executed
```

stream editor is used for transformations on input from a file or standard input similar to an editor (such as `ed`) but one pass over the file and the ability to process multiple files particularly types of

of pattern

AWK: data filtering

```
flights.csv
1 4744622522;7569618;GES011;LEPA;LPPT;I
;M;BE40;30/4/2016;30/4/2016;06:38:04;39.553612;2.758046;5;153.26;60.298;133.125;75.937;297
5;1461998284
2 4744625850;7569618;GES011;LEPA;LPPT;I
;M;BE40;30/4/2016;30/4/2016;06:38:09;39.55558;2.76227;8;156.289;59.743;135;78.75;3231;1461
998289
3 4744627330;7569618;GES011;LEPA;LPPT;I
;M;BE40;30/4/2016;30/4/2016;06:38:14;39.557549;2.766688;10;160.474;59.839;138.75;80.625;30
75;1461998294
```

```
LPPT LEPA 7/4/2016
LPPT LEPA 7/4/2016
LPPT LEPA 7/4/2016
LPPT LEPA 7/4/2016
LPPT LEPA 7/4/2016
LPPT LEPA 7/4/2016

xgeorgio@CircaQ8 ~
$ awk --field-separator=';' '$9 ~ /??\.4.2016/ {print $4 " " $5 " " $9}' flights
.csv | uniq
LEPA LPPT 30/4/2016
LPPT LEPA 7/4/2016

xgeorgio@CircaQ8 ~
$ awk --field-separator=';' '$9 ~ /??\.4.2016/ {print $4 " " $5 " " $9}' flights
.csv | uniq -c
  1082 LEPA LPPT 30/4/2016
   997 LPPT LEPA 7/4/2016

xgeorgio@CircaQ8 ~
$ wc flights.csv
 2078   4159 314407 flights.csv
```

AWK: linear interpolation

```
TSdata.txt - Notepad
File Edit Format View Help
1 12
2 8
3 2
4 -3
5 -7
```

```
ts.csv x interp1.awk x
awk 'BEGIN{dt=0.5;L=0;N=0;t0=0;y0=0;t=0;y=0;s1=0;b=0}; N>0{L++;N++;s1=($2-y0)/($1-t0);b=$2-s1*$1; t=t0;while(t<$1){y=s1*t+b;print L ":" N ":" [" t0 "," $1 "]: \t(" s1 ")t+(" b "): \t(" t "," y ")};t+=dt;N++;t0=$1;y0=$2}; L==0{t0=$1;y0=$2};L++;N++;}; END{print L " -> " N " samples"}' TSdata.txt >res.txt

2:2:[1,2]: (-4)t+(16): (1,12)
2:3:[1,2]: (-4)t+(16): (1.5,10)
3:5:[2,3]: (-6)t+(20): (2,8)
3:6:[2,3]: (-6)t+(20): (2.5,5)
4:8:[3,4]: (-5)t+(17): (3,2)
4:9:[3,4]: (-5)t+(17): (3.5,-0.5)
5:11:[4,5]: (-4)t+(13): (4,-3)
5:12:[4,5]: (-4)t+(13): (4.5,-5)
6:14:[5,7]: (-2)t+(3): (5,-7)
6:15:[5,7]: (-2)t+(3): (5.5,-8)
6:16:[5,7]: (-2)t+(3): (6,-9)
6:17:[5,7]: (-2)t+(3): (6.5,-10)
```

AWK: outliers detection

XYdata.txt - Notepad

File Edit Format View Help

```
-24.99588356591498 -17.45541063914179 1
-25.00507675642022 -23.58771947692979 1
-28.11636326588339 -41.32301770122036 1
-28.23947057191837 -36.75930825368243 1
-54.51817561994246 -18.79054706385705 1
-23.32942515695047 -15.04279029099857 1
-2.117280177237994 -55.68658447435079 1
```

```
xgeorgio@CircaQ8 ~
```

```
$ awk 'BEGIN{L=2;n1=0;m1x=-30;m1y=-30;s1x=15;s1y=12;n2=0;m2x=70;m2y=70;s2x=10;s2y=7;dx=0;dy=0}; $3==1{dx=($1-m1x)/s1x;dx=(dx<0)?-dx:dx;dy=($2-m1y)/s1y;dy=(dy<0)?-dy:dy;if((dx>L)|| (dy>L)){n1++;print "1: " dx " " dy}}; $3==2{dx=($1-m2x)/s2x;dx=(dx<0)?-dx:dx;dy=($2-m2y)/s2y;dy=(dy<0)?-dy:dy;if((dx>L)|| (dy>L)){n2++;print "2: " dx " " dy}}; END{print "Outliers: N1=" n1 " , N2=" n2}' XYdata.txt
```

```
1: 1.85885 2.14055
```

```
1: 1.12024 2.3831
```

```
1: 0.873892 2.10238
```

```
1: 2.29492 0.909004
```

```
2: 0.486175 2.02169
```

```
2: 2.46721 1.26579
```

```
2: 2.18952 0.355079
```

```
2: 2.7715 1.48925
```

```
2: 0.145815 2.06641
```

```
Outliers: N1=4, N2=5
```

AWK: MDC classifier

XYdata.txt - Notepad

File Edit Format View Help

```
-24.99588356591498 -17.45541063914179 1  
-25.00507675642022 -23.58771947692979 1
```

```
57.83235125529826 74.52355100626579 2  
70.29677449823477 64.48368848816855 2  
74.17214649392676 78.31174156292951 2  
66.27717399315232 75.91534413755186 2  
97.71502733449357 80.42478078169438 2  
62.75443458191603 71.00979441911306 2  
68.581296878849 63.11891481559147 2  
83.061131722715 73.89127488307699 2  
85.62846221271349 64.02996796593014 2  
73.57920512645296 76.11874245572909 2  
69.13089429947014 75.8186395018914 2  
54.65691319463033 67.91187962341608 2  
82.67747215691931 73.77860757022388 2  
72.16555384914786 63.5658671557541 2  
62.26641707266825 77.04233803851464 2  
68.54184781282795 55.53516295633679 2
```

```
Classes: N1=60, N2=40  
1: (-28.8887, -30.9343)  
2: (70.4564, 68.8968)
```

```
xgeorgio@CircaQ8 ~
```

```
$ awk 'BEGIN{n1=0;m1x=0;m1y=0;n2=0;m2x=0;m2y=0}; $3==1{n1++;m1x+=$1;m1y+=$2}; $3==2{n2++;m2x+=$1;m2y+=$2}; END{print "Classes: N1=" n1 " , N2=" n2 "\n1: (" m1x/n1 " , " m1y/n1 " )\n2: (" m2x/n2 " , " m2y/n2 " )}' XYdata.txt
```


AWK: k-nn classifier

XYdata.txt - Notepad

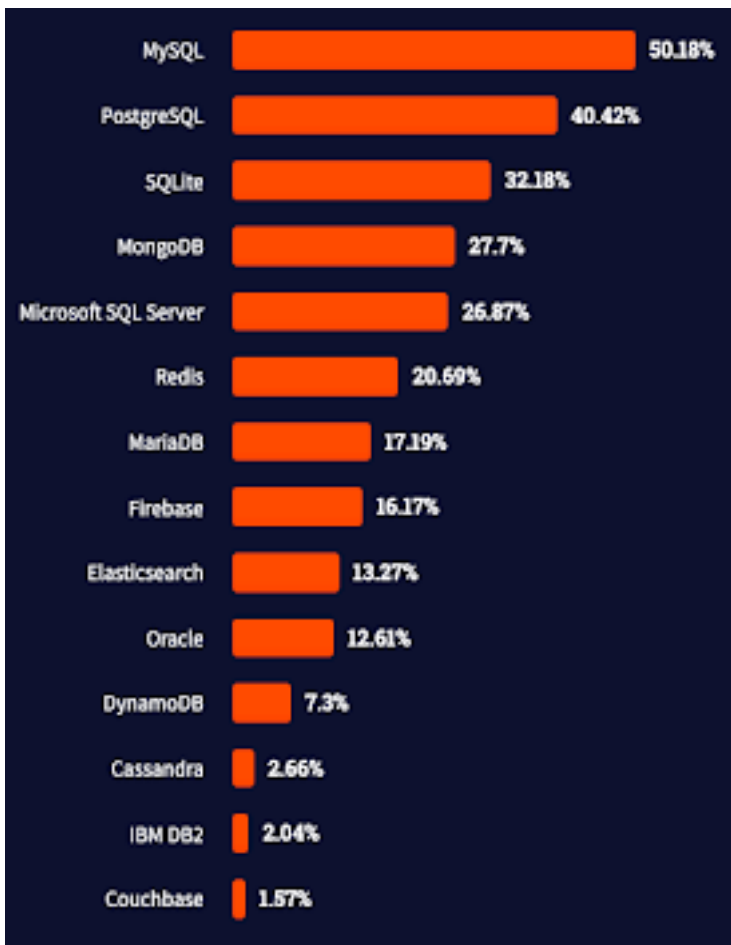
File Edit Format View Help

```
-24.99588356591498 -17.45541063914179 1
-25.00507675642022 -23.58771947692979 1
-28.11636326588339 -41.32301770122036 1
-28.23947057191837 -36.75930825368243 1
-54.51817561994246 -18.79054706385705 1
-23.32942515695047 -15.04279029099857 1
-2.117280177237994 -55.68658447435079 1
```

```
91: 58.2384 -> 2
92: 55.6203 -> 2
93: 52.1697 -> 2
94: 48.7571 -> 2
95: 33.3263 -> 2
96: 57.8583 -> 2
97: 43.4798 -> 2
98: 45.1941 -> 2
99: 36.5558 -> 2
min.dist[80]: 27.1149 -> class.id: 2
```

```
xgeorgio@CircaQ8 ~
```

```
$ awk 'BEGIN{x=35;y=41;i=0;mx=0} {ds[i]=sqrt((x-$1)*(x-$1)+(y-$2)*(y-$2));cl[i]
=$3;print i ": " ds[i] " -> " cl[i];i++;} END{for(k=1;k<length(ds);k++){if(ds[k]
]<ds[mx]){mx=k}};print "min.dist[" mx "]: " ds[mx] " -> class.id: " cl[mx]}' XY
data.txt
```



SQL: linear regression

```
-- insert data samples --  
INSERT INTO DATAXY VALUES ( 1, 1.0, -1.0, -1); -- classID: '-1' --  
--1.0, -1);  
--1.0, -1);  
--0.0, -1);  
--2.0, -1);  
  
--1.0, 1); -- classID: '+1' --  
--1.0, 1);  
--1.0, 1);  
--0.0, 1);  
--2.0, 1);
```

```
select avg(x) as x_bar,  
       avg(y) as y_bar  
from ols;
```

```
select x, avg(x) over () as x_bar,  
       y, avg(y) over () as y_bar  
from ols;
```

```
select sum((x - x_bar) * (y - y_bar)) / sum((x - x_bar) * (x - x_bar)) as slope  
from (  
  select x, avg(x) over () as x_bar,  
         y, avg(y) over () as y_bar  
  from ols) s;
```

```
select slope,  
       y_bar_max - x_bar_max * slope as intercept  
from (  
  select sum((x - x_bar) * (y - y_bar)) / sum((x - x_bar) * (x - x_bar)) as slope,  
         max(x_bar) as x_bar_max,  
         max(y_bar) as y_bar_max  
  from (  
    select x, avg(x) over () as x_bar,  
           y, avg(y) over () as y_bar  
    from ols) s;  
)
```

$$y = mx + b$$

$$m = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2}$$

$$b = \bar{y} - m\bar{x}$$

SQL: k-nn classifier

```
-- insert data samples --
INSERT INTO DATAXY VALUES ( 1, 1.0, -1.0, -1); -- classID: '-1' --
                             2.0, -1.0, -1);
                             2.5,  1.0, -1);
                             3.0,  0.0, -1);
                             3.5,  2.0, -1);
                             11.0, -1.0,  1); -- classID: '+1' --
                             12.0, -1.0,  1);
                             12.5,  1.0,  1);
                             13.0,  0.0,  1);
                             13.5,  2.0,  1);

-- k-nn fetch, simple version (works for N=1 test sample) --
-- Note: As it is, for N>1 the results of all testing
-- samples are merged into one table without grouping
-- i.e., some FOR-EACH structure is necessary for this.

SELECT
  --id1, id2, cidk  -- display the top K neighbours
  id2, SUM(cidk)/ABS(SUM(cidk))  -- display only the result
FROM TESTXY, (
  SELECT
    DATAXY.id AS id1, TESTXY.id AS id2,
    -- calculate the Euclidean distance (squared) --
    ((DATAXY.x-TESTXY.x) * (DATAXY.x-TESTXY.x)
    + (DATAXY.y-TESTXY.y) * (DATAXY.y-TESTXY.y)) AS dist,
    -- keep the corresponding class ID --
    DATAXY.cid AS cidk
  FROM DATAXY, TESTXY
  ORDER BY id2, dist ASC -- sort against distances
  LIMIT 5 -- this is the K parameter
);
```

Data import (text)

flights.csv

```
1 4744622522;7569618;GES011;LEPA;LPPT;I
;M;BE40;30/4/2016;30/4/2016;06:38:04;39.553612;2.758046;5;153.26;60.298;133.125;75.937;297
5;1461998284
```

Text Import Wizard - Step 2 of 3

This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

Delimiters

- Tab
- Semicolon
- Comma
- Space
- Other:

Treat consecutive delimiters as one

Text qualifier: "

Data preview

4744622522	7569618	GES011	LEPA	LPPT	I	M	BE40	30/4/2016	30/4/2016
4744625850	7569618	GES011	LEPA	LPPT	I	M	BE40	30/4/2016	30/4/2016
4744627330	7569618	GES011	LEPA	LPPT	I	M	BE40	30/4/2016	30/4/2016
4744630159	7569618	GES011	LEPA	LPPT	I	M	BE40	30/4/2016	30/4/2016
4744633517	7569618	GES011	LEPA	LPPT	I	M	BE40	30/4/2016	30/4/2016

Cancel < Back Next > Finish

Μέρος III: Προηγμένα εργαλεία

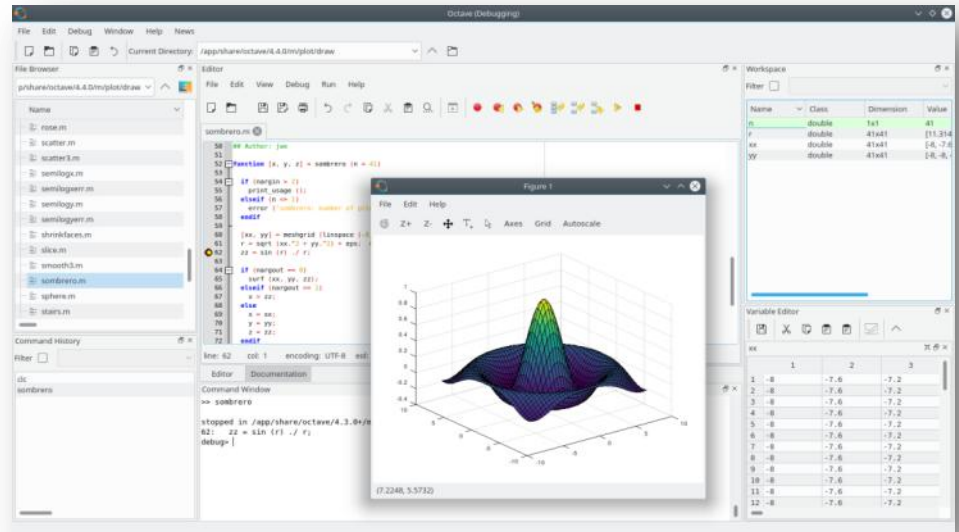
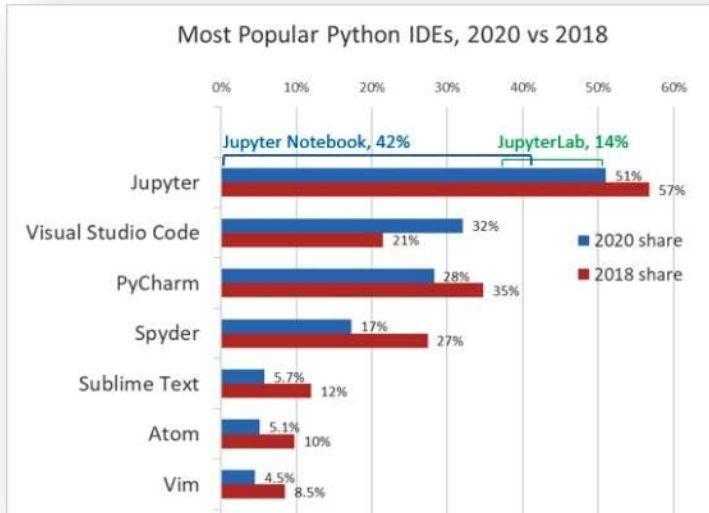
1. Χρήση στην κονσόλα (command-line).
2. Χρήση σε περιβάλλον IDE.
3. Χρήση online στο cloud.
4. Και όχι μόνο...



Specialized scripting: Python, Octave/Matlab, ...

```
import numpy as np
x = np.arange(data)
y = np.arange(data)
m, b = np.polyfit(x, y, 1)
```

$$A = [X + \text{ones}(n,1)] \setminus Y$$



Linear Regression



```
## correlation between CPI and year / quarter
```

```
cor(year, cpi)
```

```
## [1] 0.9096316
```

$$\text{cpi} = c_0 + c_1 * \text{year} + c_2 * \text{quarter},$$

```
cor(quarter, cpi)
```

```
## [1] 0.3738028
```

```
## build a linear regression model with function lm()
```

```
fit <- lm(cpi ~ year + quarter)
```

```
fit
```

```
##
```

```
## Call:
```

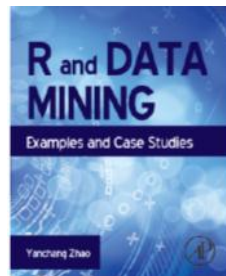
```
## lm(formula = cpi ~ year + quarter)
```

```
##
```

```
## Coefficients:
```

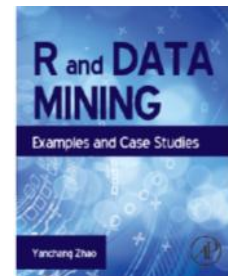
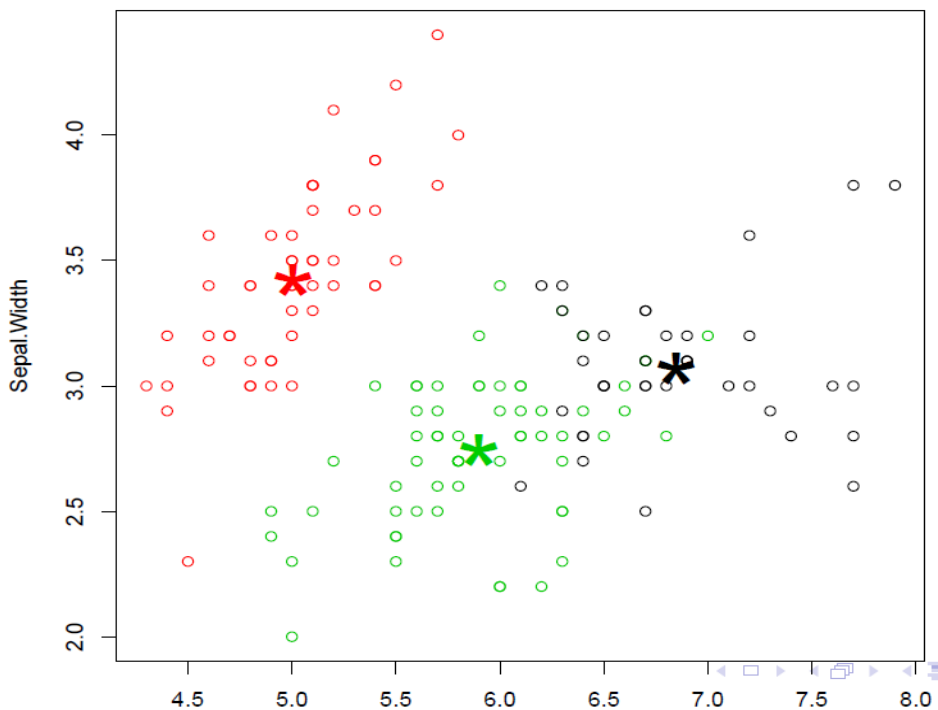
```
## (Intercept)          year          quarter
```

```
##   -7644.488          3.888           1.167
```




```
# plot clusters and their centers
```

```
plot(iris2[c("Sepal.Length", "Sepal.Width")], col=iris.kmeans$cluster)  
points(iris.kmeans$centers[, c("Sepal.Length", "Sepal.Width")],  
       col=1:3, pch="*", cex=5)
```



Weka (ML)

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter

Choose None

Current relation

Relation: breast-cancer Instances: 286

Attributes: 10 Sum of weights: 286

Attributes

All None Invert Pattern

No.	Name
1	<input checked="" type="checkbox"/> age
2	<input type="checkbox"/> menopause
3	<input type="checkbox"/> tumor-size
4	<input type="checkbox"/> inv-nodes
5	<input type="checkbox"/> node-caps
6	<input type="checkbox"/> deg-malig
7	<input type="checkbox"/> breast
8	<input type="checkbox"/> breast-quad
9	<input type="checkbox"/> irradiat

Remove

Status

OK

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose Logistic -R 1.0E-8 -M -1 -num-decimal-places 4

Test options

Use training set

Supplied test set Set...

Cross-validation Folds 10

Percentage split % 66

More options...

(Nom) class

Start Stop

Result list (right-click for options)

11:01:54 - functions.Logistic

Status

OK

Classifier output

```
Correctly Classified Instances      601      78.2552 %
Incorrectly Classified Instances    167      21.7448 %
Kappa statistic                    0.4966
Mean absolute error                 0.3063
Root mean squared error             0.3908
Relative absolute error             67.3928 %
Root relative squared error        81.9907 %
Total Number of Instances          768

=== Detailed Accuracy By Class ===

                TP Rate  FP Rate  Precision  Recall  F-Measure  MCC
Weighted Avg.   0.783   0.310   0.778     0.783   0.775     0.504

=== Confusion Matrix ===

  a  b  <-- classified as
445 55 |  a = tested_negative
112 156 |  b = tested_positive
```

Log x 0

WatsonX (IBM)

The image displays three overlapping screenshots of the IBM Watson Studio interface, illustrating different stages of a machine learning workflow.

Top-Left Screenshot: Join Operation Configuration

This screenshot shows the configuration for a 'Join' operation. The 'Left join' dropdown is selected. The 'Source' is set to 'ORDER_DETAILS' with a suffix of '.x'. The 'Data set to be joined' is 'ORDER_M...' with a suffix of '.y'. The 'JOIN KEYS' section shows 'ORDER_DETAILS' joined to 'ORDER_HEAD...' and 'ORDER_DETAIL_CO...' joined to 'ORDER_NUM...'. A 'Cancel' button is visible at the bottom.

Middle Screenshot: Search Nodes

This screenshot shows the 'Search Nodes' interface. A search bar is at the top. Below it, a list of nodes is displayed with expandable dropdown menus for each: Input, Activation, Convolution, Core, Metric, Loss, Normalization, Embedding, Recurrent, and Optimizer. A blue arrow points from the 'Flatt' node in the middle screenshot to the 'Flatt' node in this screenshot.

Bottom-Right Screenshot: NYC Learning Multiple Attr Evaluation

This screenshot shows the 'NYC Learning Multiple Attr' model's 'Evaluation' tab. It features an 'Evaluation Events' graph and a 'Last Evaluation Result' table.

Evaluation Events Graph

The graph plots 'accuracy' (y-axis, 0 to 0.8) against 'Evaluation time' (x-axis, from 10/13/17 08:39 to 11/09/17 13:02). A teal line shows accuracy fluctuating between approximately 0.3 and 0.6. A horizontal red line is drawn at an accuracy of 0.6. A legend indicates 'accuracy' is selected.

Last Evaluation Result Table

Property	Value
Version	90697ee3-99cc-45b5-91f1-a54af74eb075
Phase	monitoring
Accuracy	0.633
WeightedFMeasure	0.66
WeightedRecall	0.633
WeightedPrecision	0.693

Below the table, there is a 'Performance Monitoring' section with an 'Edit configuration' link.

```
In [1]: # setup the matplotlib graphics library and configure it to show
# figures inline in the notebook
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
```

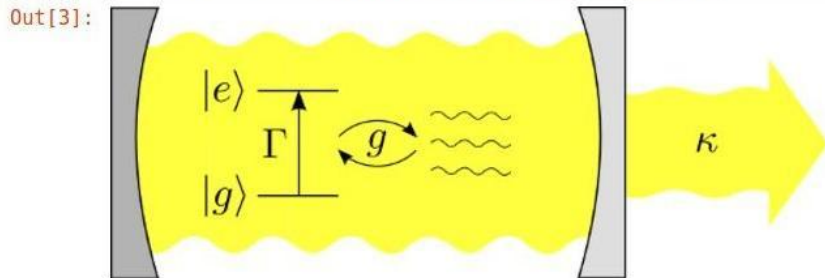
```
In [2]: # make qutip available in the rest of the notebook
from qutip import *

from IPython.display import Image
```

Introduction and model

Consider a single atom coupled to a single cavity mode, as illustrated in the figure below. If there atom excitation rate Γ exceeds the relaxation rate, a population inversion can occur in the atom, and if coupled to the cavity the atom can then act as a photon pump on the cavity.

```
In [3]: Image(filename='images/schematic-lasing-model.png')
```



In Depth: Linear Regression

Just as naive Bayes (discussed earlier in *In Depth: Naive Bayes Classification*) is a good starting point for classification tasks, linear regression models are a good starting point for regression tasks. Such models are popular because they can be fit very quickly, and are very interpretable. You are probably familiar with the simplest form of a linear regression model (i.e., fitting a straight line to data) but such models can be extended to model more complicated data behavior:

In this section we will start with a quick intuitive walk-through of the mathematics behind this well-known problem, before seeing how before moving on to see how linear models can be generalized to account for more complicated patterns in data.

We begin with a simple example:

```
import numpy as np
import matplotlib.pyplot as plt

# Generate data for analysis
x = np.linspace(0, 2 * np.pi, 100)
y = np.sin(x)

# Add noise
noise = np.random.normal(scale=0.1, size=x.shape)
y += noise

# Perform calculations
smoothness = 3.0 # Smoothness parameter
knots = 2 # Spline order
knots = -1 # Estimate of number of knots needed (-1 = maxval)

# Find the knot points
knot_points, u = splprep([x, y, z], s=smoothness, k=knots, nestp=-1)

# Evaluate spline, including interpolated points
xnew, ynew, znew = splev(linspace(0, 1, 600), knot_points)

# Plot results
plt.subplot(2, 1, 1)
plt.plot(x, y, 'bo-', label='Data with X-Y Cross Section')
```

Seattle Weather: 2012-2015

Maximum Daily Temperature (F)

Julia

```
using Plots
plot(datasets)
```

```
eigen(x)
```

```
eigen(complex)
```

```
60: Array{ComplexF64, 1}
eigenvalues:
10-element Array{ComplexF64, 1}
-0.79881511 + 0.60453446im
```

```
using Plots
plot(x, y)
```

```
using Plots
plot(x, y)
```

```
using Plots
plot(x, y)
```

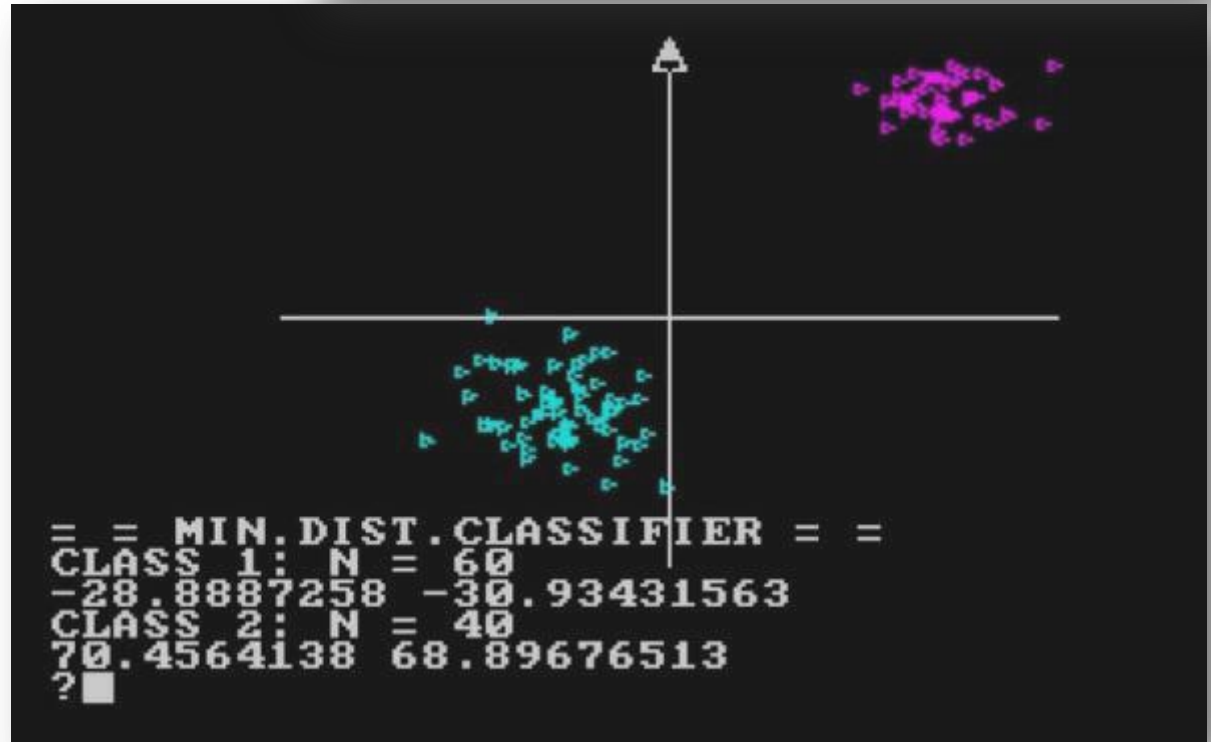
MDC even in Logo...

```
TO RDREC1
MAKE "XYC READLIST
IF :XYC = [] [STOP]
PRINT :XYC
MAKE "X (FIRST :XYC)
MAKE "Y (FIRST BUTFIRST :XYC)
MAKE "C INT (LAST :XYC)
IF :C = 1 [UPDC1]
IF :C = 2 [UPDC2]
MAKE "X ROUND :X
MAKE "Y ROUND :Y
MARKXY :X :Y :C
RDREC1
END

TO MDC
CS
CT
OPEN "XYDATA.TXT
SETREAD "XYDATA.TXT
MAKE "SX1 0
MAKE "SY1 0
MAKE "N1 0
MAKE "SX2 0
MAKE "SY2 0
```

```
IBM Personal Computer Logo Version 1.00
(C) Copyright IBM Corp. 1983
(C) Copyright LCS1 1983
Serial Number 0767405259
```

```
WELCOME TO LOGO
?
```



Σύνοψη

- Περιεχόμενα:
 - Τι είναι η Μηχανική Μάθηση και η Αναλυτική Δεδομένων (ML/DA).
 - Προπαρασκευή δεδομένων (pre-processing), είδη προβλημάτων ML/DA.
 - Επεξεργασία αρχείων, command-line tools, SQL, αρχεία κειμένου (imports).
 - Προηγμένα εργαλεία στην κονσόλα (command-line), σε IDE, στο cloud.
- Αναφορές:
 - «Εισαγωγή στη Μηχανική Μάθηση και στην Αναλυτική Δεδομένων», Χ. Γεωργίου, Α' κύκλος ανοικτών μαθημάτων ΕΠΕ – <https://youtu.be/mlU4SvyfRqA>
 - «Εισαγωγή στην Αναλυτική Δεδομένων με τη γλώσσα R», Χ. Γεωργίου, Β' κύκλος ανοικτών μαθημάτων ΕΠΕ – <https://youtu.be/7aaNnXxyzgl>
 - Dunham: Data Mining – Introductory and Advanced Topics. Prentice Hall, 2003.
 - Tan, Steinbach, Kumar: Introduction to Data Mining. Addison Wesley, 2006.

```

MOVE 1 TO DATA-C(N-T).
ADD 1 TO N-CHANGED.
GO TO LOOP-SCAN.
SELECT-CL2.
ADD DATA-X(N-T) TO SUM2-X.
ADD DATA-Y(N-T) TO SUM2-Y.
ADD 1 TO N-CL2.
IF DATA-C(N-T) EQUAL 2 GO TO LOOP-SCAN.
MOVE 2 TO DATA-C(N-T).
ADD 1 TO N-CHANGED.

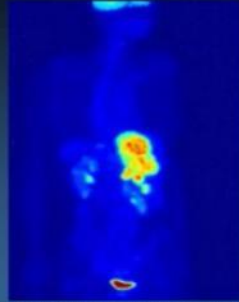
```

```

91 id : Integer := 0; -- target ID (counter)
92 det : Integer := 0; -- detection slots in sequence
93 pur : Integer := 0; -- rel. power of detection
94 pur0 : Integer := detLimit; -- rel. power baseline (adapt
95 disp : Boolean := False; -- target reporting (flag)
96
97 begin
98 -- process the FOV slots --
99 for p in 1..(seekerData'length)-1 loop
100 -- rel. power is current detection 'step'
101 pur := abs(seekerData(p+1)-seekerData(p));
102 if pur >= detLimit then
103 -- detection valid, continue analysis
104 if pur > pur0+detLimit then
105 -- strong new 'step' from baseline (new target)
106 pur0 := pur; -- update the baseline
107 det := 0; -- reset the run-length
108 disp := False; -- enable target reporting
109 end if;
110
111 det := det + 1;
112
113 if
114
115 d:
116 p:
117 d:

```

Ανάλυση Ιατρικής Εικόνας μέσω Η/Υ



- Εικόνα (2-D): Επικλυτόμενες δομές ιστών
- Τομογραφία (3-D): Όγκος πληροφοριών
- Διαφορετικές τεχνολογίες απεικόνισης
- Διαφορετικά διαγνωστικά χαρακτηριστικά
- Η διαγνωστική πληροφορία συνήθως δεν είναι καλώς ορισμένη (θόρυβος, ασάφειες δομών)
- Η διαγνωστική διαδικασία είναι συνήθως ασαφής, πολύπλοκη και βασίζεται στην εμπειρία (ιατρός)

- ⇒ Η χρήση Η/Υ επιτρέπει την αυτόματη επεξεργασία και ενοποίηση (τομογραφία) μεγάλου όγκου δεδομένων απεικόνισης
- ⇒ ...αλλά εξακολουθεί να έχει σημαντικούς περιορισμούς ως προς τη σημασιολογική ερμηνεία τους (διαγνωστική πληροφορία)

Παράδειγμα τριτοβάθμιας ανακάλυψης μόνιμης τριτοβάθμιας (γυναικα) - Wikipedia.org

12:40 / 2:00:20

- Hamming (7,4) error correction codes in **R**
- Kmeans clustering in **COBOL**
- Bi-directional Associative Memory (BAM) in **Arduino/C**
- Linear Regression in **SQL, Matlab**
- ...

YouTube:

@ApneaCoding



<https://www.youtube.com/@apneacoding>

<https://www.facebook.com/apneacoding>

Github:

@xgeorgio



<https://github.com/xgeorgio>

<http://apneacoding.eu>

Ένας ψηφιακός κόσμος γεμάτος γνώση για όλους

Σύμφωνα με το Καταστατικό της Ένωσης Πληροφορικών Ελλάδας, ένας από τους βασικούς σκοπούς της λειτουργίας της είναι η προώθηση της γνώσης και χρήσης των πληροφορικών αγαθών από το κοινωνικό σύνολο και η εξάλειψη της τεχνοφοβίας και του "αναλφαριθμισμού" στην Πληροφορική.



<https://courses.epe.org.gr>

Σχετικά με τα ανοικτά μαθήματα της Ένωσης Πληροφορικών Ελλάδας:

- ✓ Τα μαθήματα πραγματοποιούνται εξ ολοκλήρου διαδικτυακά, ζωντανά μέσω της πλατφόρμας Zoom.
- ✓ Η συμμετοχή σε όλα τα μαθήματα είναι ελεύθερη για οποιονδήποτε από οπουδήποτε στην Ελλάδα ή στο εξωτερικό.
- ✓ Δεν υπάρχει οικονομικό κόστος ή άλλες προϋποθέσεις συμμετοχής.
- ✓ Οι Εισηγητές είναι μέλη της Ένωσης Πληροφορικών Ελλάδας και πραγματοποιούν τα μαθήματα εθελοντικά.
- ✓ Τα μαθήματα μαγνητοσκοπούνται και παραμένουν διαθέσιμα για σύγχρονη παρακολούθηση στο Αρχείο Μαθημάτων.
- ✓ Η εκπαίδευση που παρέχεται μέσω των ανοικτών διαδικτυακών μαθημάτων είναι άτυπη και δεν παρέχονται βεβαιώσεις παρακολούθησης στους συμμετέχοντες.



Ερωτήσεις



Χάρης Γεωργίου (MSc, PhD)

<https://www.linkedin.com/in/xgeorgio/>

https://twitter.com/xgeorgio_gr