

Πανεπιστήμιο Πατρών  
Τμήμα Μηχανικών Η/Υ και Πληροφορικής

ΔΠΜΣ ΥΔΑ & ΠΠΣ ΤΜΗΥΠ  
Μάθημα: Τεχνολογίες Αποκεντρωμένων Δεδομένων  
Διδάσκων: Σ. Σιούτας

# Εισαγωγή στη Μηχανική Μάθηση (Machine Learning) με τη γλώσσα προγραμματισμού Python

**Χρήστος-Παναγιώτης Μπαλατσούρας**

Υποψήφιος Διδάκτορας στο Τμήμα Μηχανικών Η/Υ και Πληροφορικής του Πανεπιστημίου Πατρών

[balatsouras@ceid.upatras.gr](mailto:balatsouras@ceid.upatras.gr)

# Περίγραμμα Διάλεξης

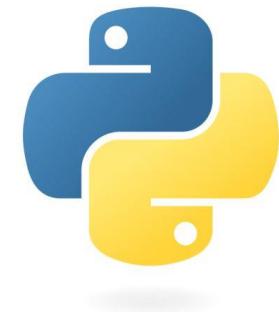
- Η γλώσσα προγραμματισμού Python και οι περιπτώσεις χρήσης αυτής.
- Η έννοια των Dataframes και η βιβλιοθήκη Pandas.
- Μηχανική Μάθηση (ML) με τη βιβλιοθήκη Sci-kit Learn.
- Παράδειγμα εφαρμογής Machine Learning.

# Εισαγωγή στο Πλαίσιο της Διάλεξης

- Στα πλαίσια των τεχνολογιών αποκεντρωμένων δεδομένων είναι απαραίτητη η χρήση τεχνικών αποδοτικής διαχείρισης δεδομένων.
- Μηχανική Μάθηση (Machine Learning – ML) → Αρκετά επίκαιρη και δημοφιλής έννοια που περιλαμβάνει τεχνικές με σκοπό την πρόβλεψη και την εξαγωγή συμπερασμάτων από τα δεδομένα.
- Στόχος → Η εφαρμογή των παραπάνω για εξαγωγή χρήσιμης γνώσης και λήψη αποφάσεων.
- Για την υποστήριξη των παραπάνω χρησιμοποιούνται εργαλεία όπως η Γλώσσα Προγραμματισμού Python και οι βιβλιοθήκες: Pandas, Numpy, Scikit-Learn, TensorFlow και άλλα.

# Η Γλώσσα Προγραμματισμού Python

- Προτάθηκε αρχικά από τον Guido van Rossum το 1991.
- Γλώσσα υψηλού επιπέδου και γενικού σκοπού με μια πληθώρα εφαρμογών.
- Χρήση σε ευρύ φάσμα εφαρμογών από WEB Development μέχρι Machine Learning.
- Υποστηρίζει πολλαπλά προγραμματιστικά μοντέλα (multi-paradigm), όπως διαδικαστικό (sequential), αντικειμενοστραφή (object-oriented) και συναρτησιακό (functional).
- Υποστηρίζει μεταβλητές με δυναμικούς τύπους δεδομένων.
- Η βασική λειτουργικότητα της γλώσσας περιλαμβάνεται στο standard library.
- Βασική σύνταξη της γλώσσας:
  - Εστιάζει στην αναγνωσιμότητα του κώδικα.
  - Κάθε εντολή ξεκινάει σε νέα γραμμή χωρίς τη χρήση τερματικού χαρακτήρα (όπως το «;»).
  - Η εμβέλεια των τμημάτων του κώδικα καθορίζεται με τη χρήση tabs (indentation).



# Python: Βασικές Περιπτώσεις Χρήσης & Χρήσιμες Βιβλιοθήκες

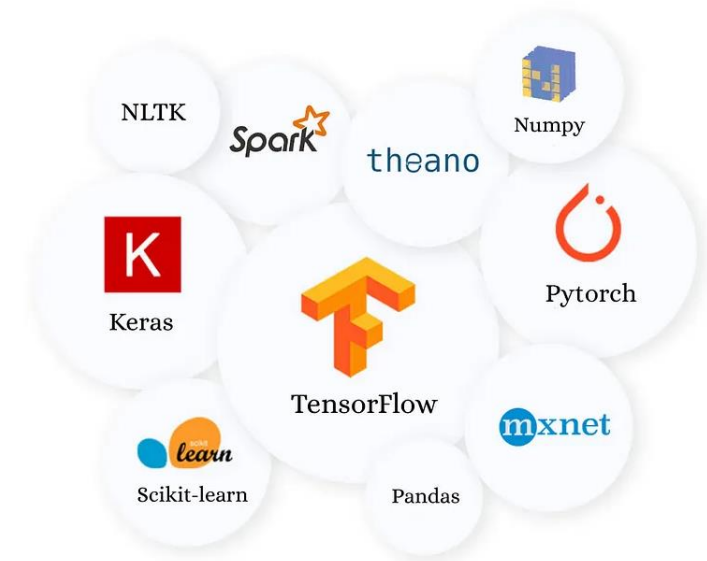
**WEB Development:** Υλοποίηση εφαρμογών του διαδικτύου με τη χρήση frameworks όπως Flask, Django και λοιπά.

**Data Science:** Διαχείριση, επεξεργασία, ανάλυση και οπτικοποίηση συνόλων δεδομένων με βιβλιοθήκες όπως Pandas, Numpy και Matplotlib.

**Machine Learning & AI:** Εκπαίδευση και αξιολόγηση μοντέλων μηχανικής μάθησης και νευρωνικών δικτύων με βιβλιοθήκες Scikit-Learn, TensorFlow και άλλα.

**Υλοποίηση εφαρμογών επιφάνειας εργασίας:** Εφαρμογές γραφικού περιβάλλοντος (GUI) με βιβλιοθήκες όπως το Tkinter και το PyQt.

**WEB Crawler Applications:** Ανάκτηση πληροφορίας από το διαδίκτυο με τη βασική βιβλιοθήκη requests της python.



Εικόνα: Δημοφιλείς βιβλιοθήκες για Data Science & Machine Learning  
(Πηγή: [towardsdatascience.com](https://towardsdatascience.com))

# Python: Διαχείριση δεδομένων

- Αναπαράσταση δεδομένων:
  - Αριθμητικά δεδομένα: Integer, Float, Complex Number
  - Ακολουθίες: Αλφαριθμητικό (String), Πίνακας (List), Πλειάδα (Tuple)
  - Δυαδικές τιμές (Boolean)
  - Συλλογές (Sets)
  - Λεξικά (Dictionaries)
- Προέλευση και αποθήκευση δεδομένων:
  - Αρχείο (Κείμενο, CSV, JSON)
  - Βάση Δεδομένων (mySQL, SQLite, MongoDB)
  - Τοποθεσία WEB (Χρήση WEB Crawler)

# Εισαγωγή στη βιβλιοθήκη Pandas

- Αξιοσημείωτη βιβλιοθήκη που χρησιμοποιείται σε μια πληθώρα περιπτώσεων χρήσης για ανάλυση δεδομένων.
- Προτάθηκε αρχικά από τον Wes McKinney το 2008.
- Περιλαμβάνει τις δομές δεδομένων: Series και DataFrame.
- Υποστηρίζει λειτουργίες όπως:
  - Φόρτωση δεδομένων από αρχεία CSV, Microsoft Excel, SQL Βάσεις Δεδομένων, κλπ.
  - Καθαρισμός συνόλου δεδομένων: Χειρισμός λανθασμένων τιμών και διπλότυπων.
  - Μετασχηματισμός συνόλου δεδομένων: Ομαδοποίηση, συγκέντρωση και φιλτράρισμα δεδομένων.
  - Αποθήκευση μετασχηματισμένων δεδομένων σε αρχεία και βάσεις δεδομένων.
- Τα δεδομένα που αποθηκεύονται στα Pandas DataFrames χρησιμοποιούνται για οπτικοποίηση σε διαγράμματα και εκπαίδευση μοντέλων μηχανικής μάθησης.



Λογότυπο βιβλιοθήκης  
Pandas

# Pandas: Η έννοια των DataFrames

- **Series:** Μονοδιάστατοι ονοματισμένοι πίνακες, σαν μια στήλη σε ένα λογιστικό φύλλο.
- **DataFrames:** Πίνακες 2 διαστάσεων που αποτελούνται από ονοματισμένες στήλες. Ουσιαστικά ένα DataFrame είναι μια συλλογή από Series. Η λογική των DataFrames είναι παρόμοια με αυτή των λογιστικών φύλλων.

	Series		Series		DataFrame																																			
	<table border="1"><thead><tr><th></th><th>apples</th></tr></thead><tbody><tr><td>0</td><td>3</td></tr><tr><td>1</td><td>2</td></tr><tr><td>2</td><td>0</td></tr><tr><td>3</td><td>1</td></tr></tbody></table>		apples	0	3	1	2	2	0	3	1	+	<table border="1"><thead><tr><th></th><th>oranges</th></tr></thead><tbody><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>3</td></tr><tr><td>2</td><td>7</td></tr><tr><td>3</td><td>2</td></tr></tbody></table>		oranges	0	0	1	3	2	7	3	2	=	<table border="1"><thead><tr><th></th><th>apples</th><th>oranges</th></tr></thead><tbody><tr><td>0</td><td>3</td><td>0</td></tr><tr><td>1</td><td>2</td><td>3</td></tr><tr><td>2</td><td>0</td><td>7</td></tr><tr><td>3</td><td>1</td><td>2</td></tr></tbody></table>		apples	oranges	0	3	0	1	2	3	2	0	7	3	1	2
	apples																																							
0	3																																							
1	2																																							
2	0																																							
3	1																																							
	oranges																																							
0	0																																							
1	3																																							
2	7																																							
3	2																																							
	apples	oranges																																						
0	3	0																																						
1	2	3																																						
2	0	7																																						
3	1	2																																						

Παράδειγμα Series και DataFrames (Πηγή: learndatasci.com)



# Pandas: Βασικός Χειρισμός DataFrames (1)

- Φόρτωση βιβλιοθήκης:
  - `import pandas as pd`
- Φόρτωση εγγραφών από αρχείο
  - `df_csv = pd.read_csv("file.csv")`
- Αποθήκευση σε αρχείο
  - `df.to_csv("output.csv", index=False)`
- Φόρτωση των πρώτων 10 εγγραφών από το DataFrame
  - `print(df.head(10))`
- Εκτύπωση μιας στήλης με βάση το όνομα
  - `print(df["Name"])`

# Pandas: Βασικός Χειρισμός DataFrames (2)

- Φιλτράρισμα γραμμών βάσει συνθήκης
  - `filtered_df = df[df['Age'] > 25]`
- Διαγραφή γραμμών στις οποίες λείπουν τιμές
  - `df_cleaned = df.dropna()`
- Αντικατάσταση τιμών που λείπουν με μηδέν
  - `df_filled = df.fillna(0)`
- Ομαδοποίηση τιμών για υπολογισμό μέσης τιμής
  - `grouped_df = df.groupby('City')['Age'].mean()`

# Εισαγωγικές έννοιες Μηχανικής Μάθησης (1)

- Απαραίτητη διαδικασία για την εξόρυξη γνώσης από δεδομένα και για την εξαγωγή συμπερασμάτων με σκοπό τη λήψη αποφάσεων.
- Ένα μοντέλο μηχανικής μάθησης «εκπαιδύεται» πάνω σε ένα σύνολο εκπαιδευτικών δεδομένων, ώστε να εξάγει τους απαραίτητους κανόνες συσχέτισης των τιμών της εισόδου με τις αντίστοιχες τιμές της εξόδου.
- Με αυτόν τον τρόπο, όταν το εκπαιδευμένο μοντέλο λάβει νέα δεδομένα εισόδου για τα οποία η έξοδος αρχικά είναι άγνωστη, μπορεί να προβλέψει την κατάλληλη τιμή της εξόδου για αυτά τα δεδομένα.
- Στο εκπαιδευτικό σύνολο δεδομένων, τα δεδομένα εισόδου ονομάζονται "features" και τα δεδομένα εξόδου ονομάζονται προβλέψεις ή "labels".

# Εισαγωγικές έννοιες Μηχανικής Μάθησης (2)

- Κατηγοριοποίηση μοντέλων μηχανικής μάθησης με βάση τη διαδικασία εκπαίδευσης:
  - **Supervised Learning:** Τα δεδομένα εισόδου συνοδεύονται από τα αντίστοιχα δεδομένα εξόδου στο training dataset.
  - **Un-supervised Learning:** Τα δεδομένα εισόδου αποτελούνται μόνο από features και η έξοδος είναι άγνωστη.
- Βασικές κατηγορίες μοντέλων μηχανικής μάθησης:
  - **Παλινδρόμηση (Regression):** Γραμμική και Λογιστική
  - **Κατηγοριοποίηση (Classification):** Decision Tree, k-NN, Naïve Bayes και λοιπά
  - **Συσταδοποίηση:** k-Means, k-Medians, Hierarchical, DBSCAN και λοιπά

# Η Βιβλιοθήκη Scikit-Learn

- Βιβλιοθήκη της Python που περιλαμβάνει υλοποιήσεις των πιο γνωστών μοντέλων μηχανικής μάθησης.
- Επίσης περιλαμβάνει τα παρακάτω:
  - Μεθόδους μετασχηματισμού των δεδομένων όπως: Standard Scaler και Label Encoder.
  - Χρήσιμες μεθόδους για τον υπολογισμό μετρικών αξιολόγησης των μοντέλων.
  - Μια πληθώρα συνόλων δεδομένων (datasets) για τη δοκιμή των μοντέλων μηχανικής μάθησης.
- Προτάθηκε αρχικά το 2007 από τον David Cournapeau.



Λογότυπο βιβλιοθήκης  
Scikit-Learn

# Εκπαίδευση Μοντέλων ML

- Βασικά βήματα εκπαίδευσης ενός Decision Tree με το Scikit-Learn:
  - Διαχωρισμός του συνόλου δεδομένων εκπαίδευσης σε training/test σε 80%/20%:  

```
>>> X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```
  - Αρχικοποίηση ενός μοντέλου Decision Tree:  

```
>>> decision_tree_model = DecisionTreeClassifier()
```
  - Εκπαίδευση του μοντέλου στο training dataset με τη συνάρτηση fit():  

```
>>> decision_tree_model.fit(X_train, y_train)
```
  - Παραγωγή προβλέψεων πάνω στο test dataset για αξιολόγηση της εκπαίδευσης:  

```
>>> y_pred = decision_tree_model.predict(X_test)
```
  - Αξιολόγηση των προβλέψεων με τη μετρική Accuracy  

```
>>> accuracy = accuracy_score(y_test, y_pred)
```

# Αξιολόγηση Μοντέλων ML

- Μερικές από τις πιο γνωστές μετρικές, ιδιαίτερα στην περίπτωση της κατηγοριοποίησης:
  - **Μετρική Accuracy:** Ποσοστό των σωστών προβλέψεων ως προς το σύνολο των προβλέψεων.
  - **Μετρική Precision:** Ποσοστό των αντικειμένων που έχουν προβλεφθεί σωστά από το μοντέλο ότι ανήκουν σε μία κλάση  $C_1$ , ως προς το σύνολο των αντικειμένων που έχουν προβλεφθεί ότι ανήκουν σε αυτή την κλάση  $C_1$ .
  - **Μετρική Recall:** Ποσοστό των αντικειμένων που έχουν προβλεφθεί σωστά από το μοντέλο ότι ανήκουν σε μία κλάση  $C_1$ , ως προς το σύνολο των αντικειμένων που ανήκουν πραγματικά σε αυτή την κλάση  $C_1$ .
  - **Μετρική F1:** Αρμονικός μέσος όρος των Precision και Recall.

# Machine Learning Example: Predicting Plant Diseases in Precision Agriculture (1)

- Απλό παράδειγμα εφαρμογής τεχνικών Machine Learning για την πρόβλεψη ασθενειών σε καλλιέργειες αμπελώνων με βάση δεδομένα αισθητήρων.
- Προτάθηκε στο case study της [δημοσίευσης](#) στο περιοδικό MDPI Sensors, με τίτλο: «WiCHORD+: A Scalable, Sustainable, and P2P Chord-Based Ecosystem for Smart Agriculture Applications», από τους: C.-P. Balatsouras, A. Karras, C. Karras, I. Karydis & S. Sioutas.
- Χρήση των μοντέλων Decision Tree, Random Forest και Naïve Bayer από τη βιβλιοθήκη Scikit-Learn της γλώσσας Python.



Article

## WiCHORD+: A Scalable, Sustainable, and P2P Chord-Based Ecosystem for Smart Agriculture Applications

Christos-Panagiotis Balatsouras <sup>1,\*</sup>, Aristeidis Karras <sup>1</sup>, Christos Karras <sup>1</sup>, Ioannis Karydis <sup>2,+</sup> and Spyros Sioutas <sup>1</sup>

<sup>1</sup> Computer Engineering and Informatics Department, University of Patras, 26504 Patras, Greece; akarras@ceid.upatras.gr (A.K.); c.karras@ceid.upatras.gr (C.K.); sioutas@ceid.upatras.gr (S.S.)  
<sup>2</sup> Department of Informatics, Ionian University, 49100 Kerkira, Greece  
\* Correspondence: balatsouras@ceid.upatras.gr (C.-P.B.); karydis@ionio.gr (I.K.)

**Abstract:** In the evolving landscape of Industry 4.0, the convergence of peer-to-peer (P2P) systems, LoRa-enabled wireless sensor networks (WSNs), and distributed hash tables (DHTs) represents a major advancement that enhances sustainability in the modern agriculture framework and its applications. In this study, we propose a P2P Chord-based ecosystem for sustainable and smart agriculture applications, inspired by the inner workings of the Chord protocol. The node-centric approach of WiCHORD+ is a standout feature, streamlining operations in WSNs and leading to more energy-efficient and straightforward system interactions. Instead of traditional key-centric methods, WiCHORD+ is a node-centric protocol that is compatible with the inherent characteristics of WSNs. This unique design integrates seamlessly with distributed hash tables (DHTs), providing an efficient mechanism to locate nodes and ensure robust data retrieval while reducing energy consumption. Additionally, by utilizing the MAC address of each node in data routing, WiCHORD+ offers a more direct and efficient data lookup mechanism, essential for the timely and energy-efficient operation of WSNs. While the increasing dependence of smart agriculture on cloud computing environments for data storage and machine learning techniques for real-time prediction and analytics continues, frameworks like the proposed WiCHORD+ appear promising for future IoT applications due to their compatibility with modern devices and peripherals. Ultimately, the proposed approach aims to effectively incorporate LoRa, WSNs, DHTs, cloud computing, and machine learning, by providing practical solutions to the ongoing challenges in the current smart agriculture landscape and IoT applications.

**Keywords:** Internet of things (IoT); wireless sensor networks (WSNs); peer-to-peer (P2P) systems; distributed hash tables (DHTs); Chord; LoRa; machine learning; precision agriculture; ubiquitous computing; sustainable water use



Citation: Balatsouras, C.-P.; Karras, A.; Karras, C.; Karydis, I.; Sioutas, S. WiCHORD+: A Scalable, Sustainable, and P2P Chord-Based Ecosystem for Smart Agriculture Applications. *Sensors* **2023**, *23*, 9486. <https://doi.org/10.3390/s23239486>



# Machine Learning Example: Predicting Plant Diseases in Precision Agriculture (2)

- Για το παράδειγμα αυτής της διάλεξης, θα μελετηθεί η εμφάνιση της ασθένειας του περονόσπορου (Grape Downy Mildew) σε καλλιέργειες αμπελώνων, με τη χρήση μεθόδων μηχανικής μάθησης σε δεδομένα που έχουν συλλεχθεί από αισθητήρες.
- Η ασθένεια του περονόσπορου είναι αρκετά διαδεδομένη, έχει ως αποτέλεσμα τη μείωση της ποιότητας και της ποσότητας του παραγόμενου αγροτικού προϊόντος και επηρεάζεται από περιβαλλοντολογικές συνθήκες όπως η θερμοκρασία και η υγρασία του αέρα.
- Στο case study της παραπάνω [δημοσίευσης](#), υλοποιείται ένα ασύρματο δίκτυο αισθητήρων που βασίζεται στο πρωτόκολλο [WiCHORD](#) από την προηγούμενη διάλεξη, το οποίο συλλέγει δεδομένα θερμοκρασίας και υγρασίας από περιοχές ενός αμπελώνα ανά τακτά χρονικά διαστήματα.
- Τα δεδομένα θερμοκρασίας και υγρασίας που συλλέχθηκαν από το δίκτυο αισθητήρων, αντιστοιχίζονται σε κλάσεις ευαλωτότητας (Susceptibility: Positive/Negative) και διαστήματος επώασης (Incubation: σε ημέρες) στην ασθένεια του Περονόσπορου, με βάση τις συνθήκες που καθορίζονται από τη σχετική βιβλιογραφία, για τη δημιουργία του αρχικού [training dataset](#).

# Machine Learning Example: Predicting Plant Diseases in Precision Agriculture (3)

- Οι γενικές συνθήκες θερμοκρασίας και υγρασίας που επηρεάζουν την εμφάνιση και το διάστημα επώασης του περονόσπορου, με βάση τις οποίες χαρακτηρίστηκε το αρχικό training dataset, όπως καθορίζονται από τη σχετική βιβλιογραφία, παρουσιάζονται στον επόμενο [πίνακα](#).
- Σε αυτό το παράδειγμα, ένα μοντέλο Naïve Bayes εκπαιδεύεται στο σύνολο δεδομένων που περιέχει ζεύγη τιμών θερμοκρασίας και υγρασίας (features), τα οποία έχουν αντιστοιχιστεί σε κλάσεις (labels) ευαλωτότητας (Susceptibility) και διαστήματος επώασης (Incubation) στην ασθένεια του Περονόσπορου.
- Μετά την εκπαίδευση, αυτό το μοντέλο θα μπορεί να προβλέψει την κλάση στην οποία ανήκει ένα άγνωστο ζεύγος τιμών θερμοκρασίας-υγρασίας.
- Το μοντέλο κατηγοριοποίησης Naïve Bayes βασίζεται στο θεώρημα Bayes από τη θεωρία πιθανοτήτων.
- Η βασική προϋπόθεση του μοντέλου Naïve Bayes είναι ότι θεωρεί τα features ενός αντικειμένου προς κατηγοριοποίηση στοχαστικά ανεξάρτητα μεταξύ τους, δοθείσης της κλάσης αυτού του αντικειμένου.

Θερμοκρασία	Υγρασία	Ευαλωτότητα (Susceptibility)	Διάστημα Επώασης (Incubation)
13 - 16 °C	≥ 85 %	Positive	8-9 days
16 - 20 °C	≥ 85 %	Positive	7 days
20 - 28 °C	≥ 85 %	Positive	5-6 days
≤ 13 °C or ≥ 28 °C	< 85 %	Negative	No incubation

Πίνακας: Συνθήκες εμφάνισης περονόσπορου ([Πηγή](#))

# Machine Learning Example: Data Preprocessing (1)

- Φόρτωση του αρχικού training dataset από το CSV αρχείο:
  - `training_dataset = pd.read_csv(dataset.csv)`
- Σε αυτό το παράδειγμα θα γίνει δυαδική κατηγοριοποίηση (binary classification) για την πρόβλεψη της ευαλωτότητας (susceptibility) του αμπελώνα στην ασθένεια του περονόσπορου, με τη χρήση του μοντέλου Naïve Bayes.
- Η πρόβλεψη του χρονικού διαστήματος της επώασης του περονόσπορου (Incubation period) αποτελεί πρόβλημα κατηγοριοποίησης πολλαπλών κλάσεων (multiclass classification) και η διαδικασία είναι παρόμοια με τη χρήση του μοντέλου Naïve Bayes.

temperature	humidity	downy_mildew_susceptibility	downy_mildew_incubation
24.1	63.4	Negative	No incubation
23.1	68.7	Negative	No incubation
16.1	76.9	Negative	No incubation
14.4	79.4	Negative	No incubation
18.6	69.7	Negative	No incubation
...	...	...	...
17.3	99.9	Positive	7 days
17.3	99.9	Positive	7 days
17.3	99.9	Positive	7 days
17.2	99.5	Positive	7 days
17.3	99.7	Positive	7 days

Εικόνα: Το pandas dataframe του training dataset μετά από τη φόρτωση από το αρχείο CSV

# Machine Learning Example: Data Preprocessing (2)

- Επιλογή features για το X dataset που αποτελεί την είσοδο του μοντέλου:
  - `X = numpy.array(dataset_df[["temperature", "humidity"]])`
- Επιλογή features για το y dataset που αποτελεί την έξοδο του μοντέλου:
  - `y = numpy.array(dataset_df["downy_mildew_susceptibility"])`
- Κωδικοποίηση της εξόδου του μοντέλου με τη χρήση του μετασχηματισμού Label Encoder από τη βιβλιοθήκη Scikit-Learn:
  - `le = LabelEncoder() # define the LabelEncoder Object`
  - `y = le.fit_transform(y) # encode the y dataset`

y dataset (label encoded): `[0 0 0 ... 1 1 1]`
- Ο μετασχηματισμός του y dataset χρησιμεύει για τη μετατροπή των ονομάτων των κλάσεων σε μορφή που είναι κατανοητή από το μοντέλο.
- Διαχωρισμός του αρχικού training dataset σε Train / Test datasets για την εκπαίδευση και την αξιολόγηση του μοντέλου, με αναλογία 70%-30%:
  - `X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)`

```
X dataset:  
[[24.1 63.4]  
 [23.1 68.7]  
 [16.1 76.9]  
 ...  
 [17.3 99.9]  
 [17.2 99.5]  
 [17.3 99.7]]
```

```
y dataset:  
[['Negative']  
 ['Negative']  
 ['Negative']  
 ...  
 ['Positive']  
 ['Positive']  
 ['Positive']]
```

# Machine Learning Example: ML Model Training

- Ορισμός του μοντέλου Gaussian Naïve Bayes Classifier από τη βιβλιοθήκη Scikit-learn με τις προεπιλεγμένες παραμέτρους:
  - `nb_model = GaussianNB()`
- Εκπαίδευση του μοντέλου Naïve Bayes Classifier στα `X_train` & `y_train` datasets:
  - `nb_model.fit(X_train, y_train)`
- Παραγωγή προβλέψεων στο `X_test` dataset για αξιολόγηση της εκπαίδευσης του μοντέλου.
  - `y_pred = nb_model.predict(X_test)`

# Machine Learning Example: ML Model Evaluation

- Η αξιολόγηση της εκπαίδευσης του μοντέλου γίνεται αρχικά με την παραγωγή προβλέψεων στο Test dataset ( $X_{\text{test}}$ ), για το οποίο η έξοδος είναι γνωστή ( $y_{\text{test}}$ ) και έπειτα με τη σύγκριση των προβλέψεων στο Test dataset ( $y_{\text{pred}}$ ) με τις πραγματικές τιμές της εξόδου του ( $y_{\text{test}}$ ).
- Οι παρακάτω μετρικές αξιολόγησης περιλαμβάνονται στη βιβλιοθήκη Scikit-learn.
- Αξιολόγηση με τη μετρική accuracy score:
  - `accuracy = accuracy_score(y_test, y_pred)`
- Αξιολόγηση με τη μετρική precision score:
  - `precision = precision_score(y_test, y_pred)`
- Αξιολόγηση με τη μετρική recall score:
  - `recall = recall_score(y_test, y_pred)`
- Αξιολόγηση με τη μετρική f1 score:
  - `f1_score(y_test, y_pred)`

Μετρική	Τιμή
Accuracy	0.90
Precision	0.86
Recall	0.95
F1-score	0.90

Πίνακας: Αποτελέσματα αξιολόγησης του μοντέλου Gaussian Naïve Bayes

**Σας ευχαριστώ για την προσοχή σας!**