Αναγνώριση Προτύπων & Μηχανική Μάθηση

Σοφικίτης Οδυσσέας 10130 sodyssea@ece.auth.gr
Τομπουλίδης Ρωμανός 10041 romanost@ece.auth.gr

Ζητούμενο της εργασίας είναι η εκπαίδευση διαφορετικών ταξινομητών και η σύγκριση των αποτελεσμάτων τους για 2 σύνολα δεδομένων.

Θα χρησιμοποιηθούν οι ταξινομητές:

- Bayes
- K-Nearest Neighbor
- SVM
- Neural Networks

Τα 2 μέρη της εργασίας

1ο Μέρος

Χωρίζοντας ένα σύνολο δεδομένων σε 2 ισάριθμα θα εκπαιδευτούν οι ταξινομητές Bayes, K-NN, SVM με αυτή την σειρά και θα συγκριθούν μορφές τους.

2ο Μέρος

Διαθέτοντας ένα σύνολο δεδομένων για εκπαίδευση καλούμαστε να βρούμε τον καλύτερο ταξινομητή για ένα άγνωστο σύνολο εξέτασης.

1ο μέρος: Ταξινομητής Bayes

Bayes: Maximum Likelihood (1/2)

Στο Bayes ταξινομητή θεωρούμε για τα δεδομένα μας μία κατανομή $p(\mathbf{x}|\omega_i) \sim \mathbf{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$.

Η προσέγγιση της κατανομής γίνεται από τα δεδομένα εκπαίδευσης χρησιμοποιώντας την *Maximum Likelihood* τεχνική.

Η *Maximum Likelihood* βασίζεται στην μεγιστοποίηση του $p(D|\boldsymbol{\theta}) = p(\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}|\boldsymbol{\theta})$ για ένα διάνυσμα παραμέτρων $\boldsymbol{\theta}$.

Στην προκειμένη περίπτωση $\boldsymbol{\theta} = [\theta_1 \ \theta_2] = [\mu \ \Sigma]$, όπου μ η αναμενόμενη τιμή και Σ ο πίνακας συνδιασποράς.

Bayes: Maximum Likelihood (2/2)

Για την μεγιστοποίηση της $p(D|\boldsymbol{\theta})$ παίρνουμε:

$$\theta_1 = \frac{1}{N} \Sigma_{n=1}^{N} x_n$$

Και:

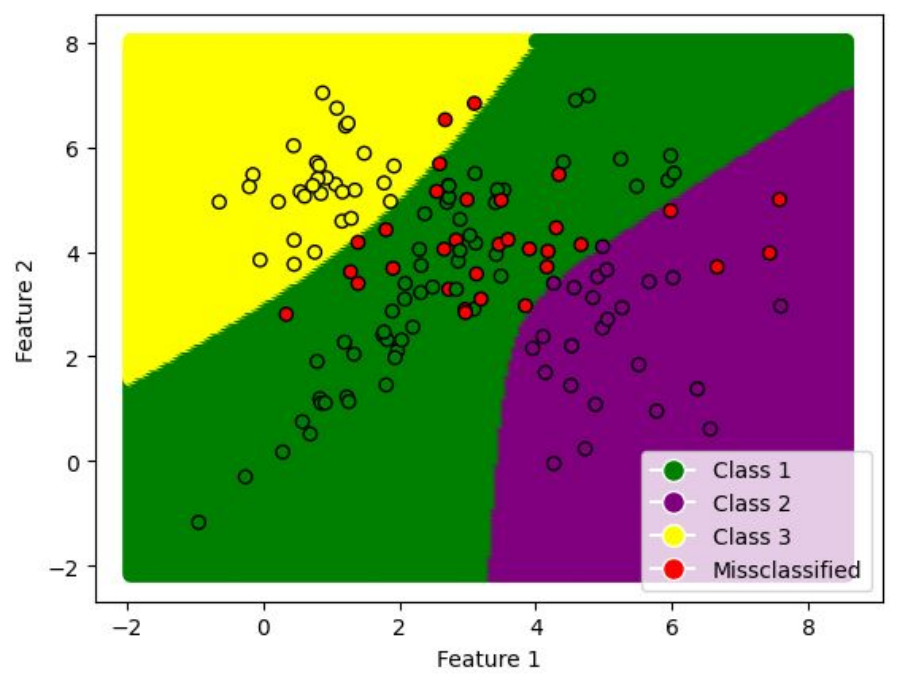
$$\theta_2 = \frac{1}{N} \Sigma_{n=1}^{N} (x_n - \theta_1)^T (x_n - \theta_1)$$

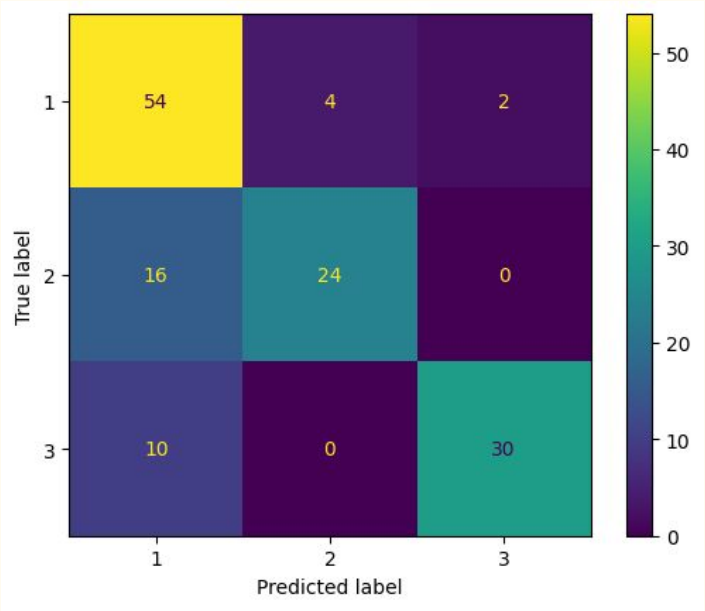
Δηλαδή, την δειγματική μέση τιμή και τον δειγματικό πίνακα συνδιασποράς αντίστοιχα.

Bayes: Πειράματα

1. Ίδιος πίνακας συνδιασποράς για όλες τις κλάσεις
2. Διαφορετικός πίνακας συνδιασποράς για κάθε κλάση

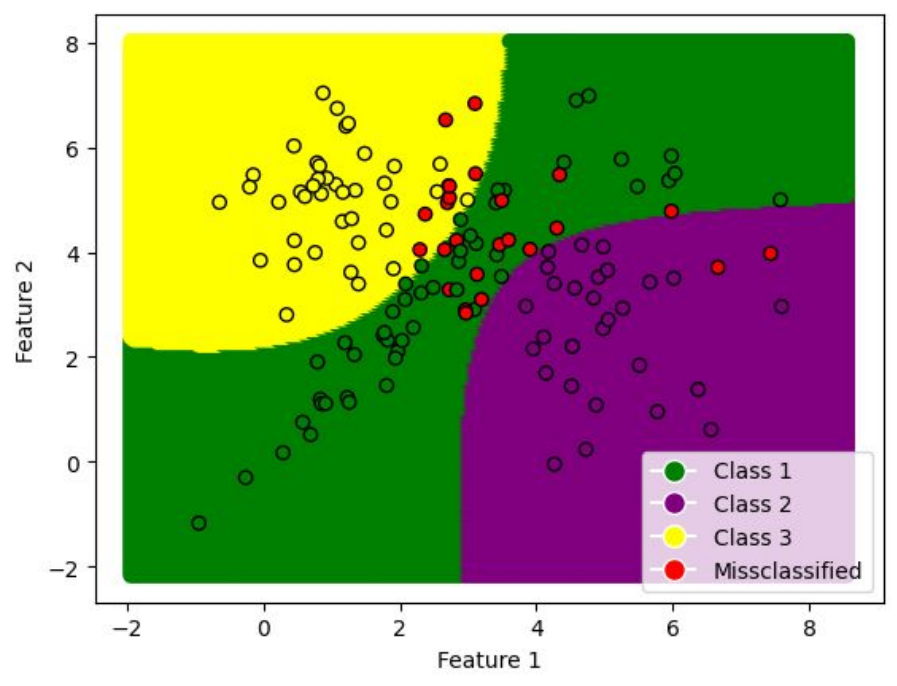
Bayes: Ίδιος Πίνακας Συνδιασποράς

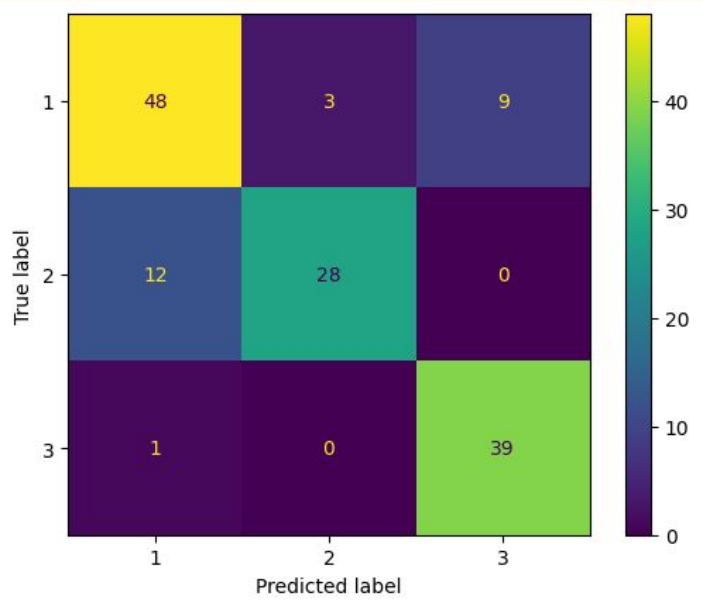




Ακρίβεια: 77.14%

Bayes: Διαφορετικοί Πίνακες Συνδιασποράς





Ακρίβεια: 82.14%

Bayes: Συμπεράσματα

Όπως αναμενόταν ο Bayes με Διαφορετικούς Πίνακες Συνδιασποράς είναι ανώτερος του Bayes με Ίδιο Πίνακα Συνδιασποράς. Παρόλα αυτά δεν μπορούμε να χαρακτηρίσουμε τον ταξινομητή ικανοποιητικό έχοντας 82.14%.

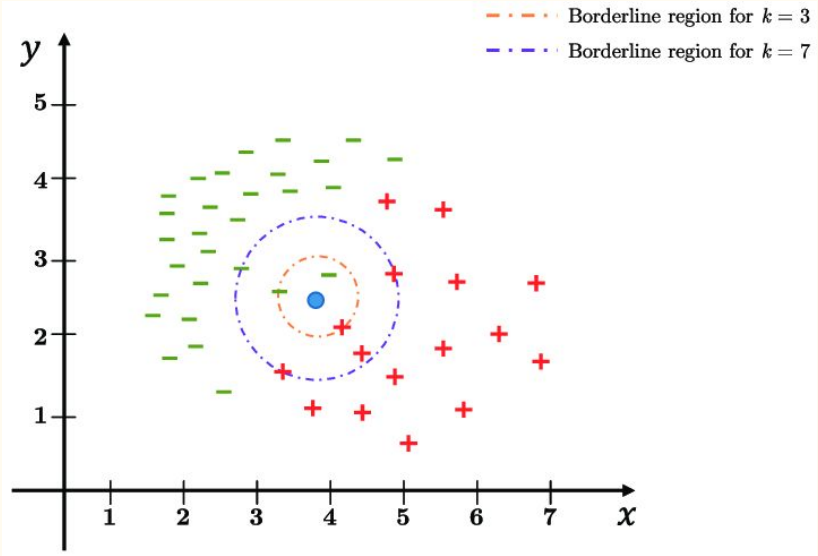
Ταξινομητής K-NN

kNN

Έχοντας τα σημεία εκπαίδευσης στο feature space, αποφασίζουμε για τα καινούρια σημεία από την πλειοψηφία των labels των k κοντινότερων σημείων.

Για τον ταξινομητή k-NN οι υπερ-παράμετροι που χρησιμοποιούνται είναι:

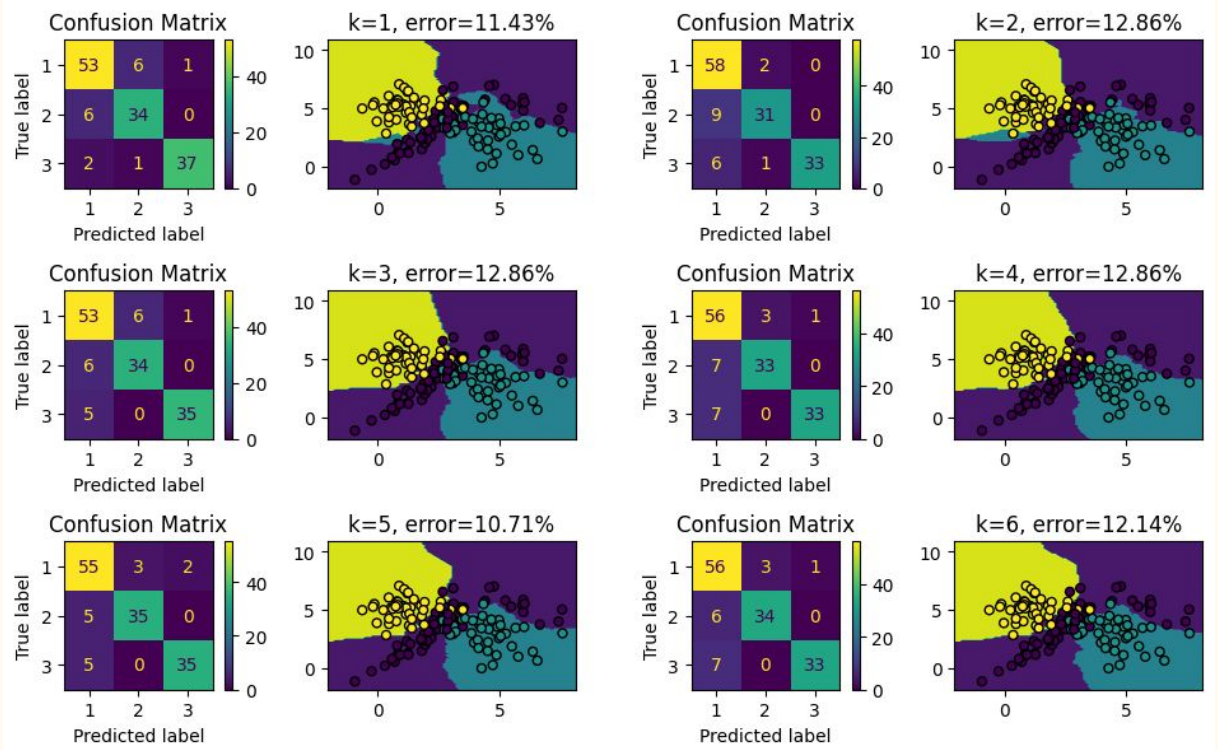
- k, ο αριθμός των γειτόνων
- Ο τρόπος υπολογισμού των βαρών
- Η μετρική απόστασης



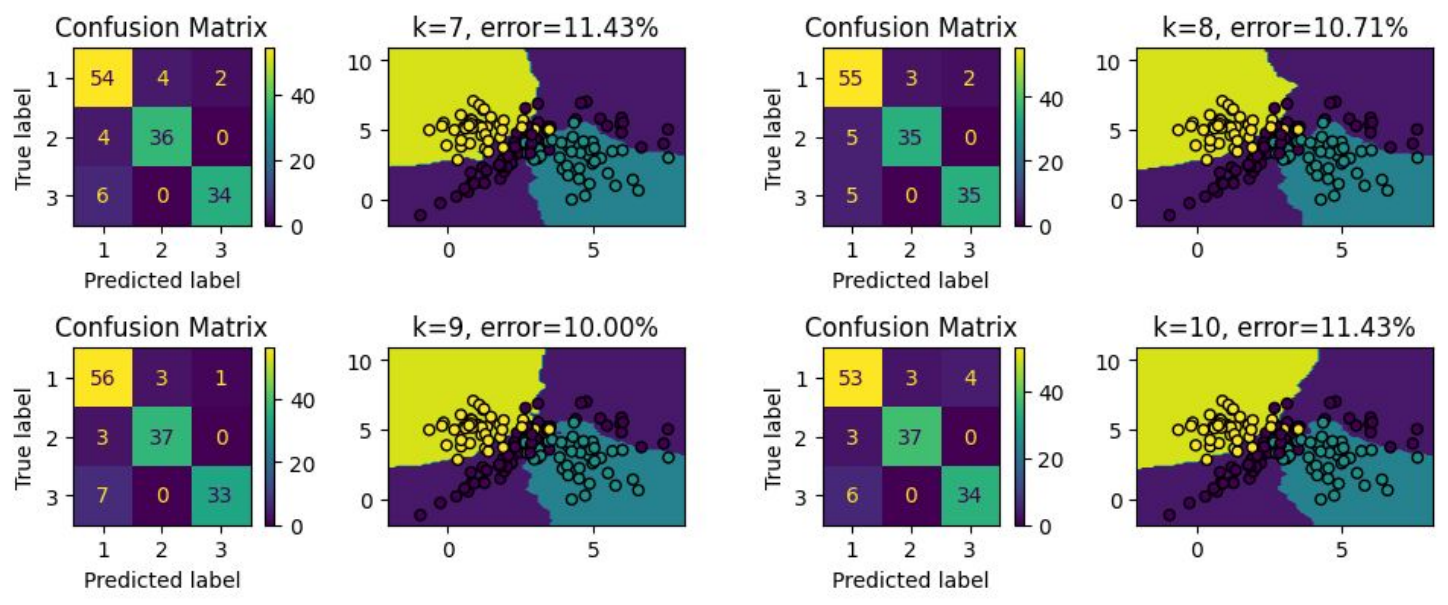
kNN: Πειράματα

1. k-NN ταξινομητής για k από 1 μέχρι 10

kNN: Αποτελέσματα (1/2)



kNN: Αποτελέσματα (2/2)



Καλύτερη επίδοση: K=9, error=10.00%

kNN: Συμπεράσματα

Με 90% ακρίβεια για k=9 βλέπουμε πως ο k-NN δουλεύει πολύ καλά για δεδομένα που είναι αρκετά ομαδοποιημένα.

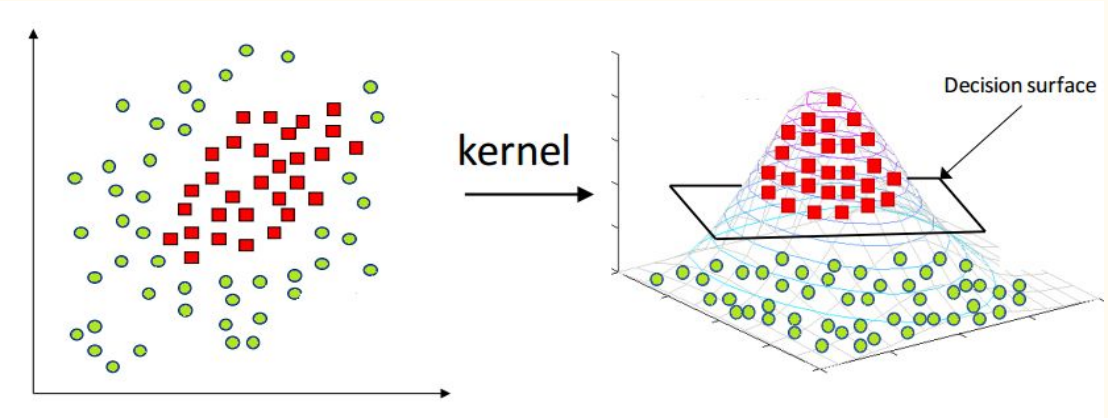
Χαρακτηριστικά, η χειρότερη απόδοση του k-NN ειναι για k=2, 3, 4 με ακρίβεια 87,14%, που παραμένει κατά 5% καλύτερη από τον Bayes με διαφορετικούς πίνακες συνδιασποράς.

Ταξινομητής SVM

SVM

Το κύριο χαρακτηριστικό των ταξινομητών SVM είναι πως χρησιμοποιούν διανύσματα για να χωρίσουν τα δεδομένα στο χώρο. Στην δική μας περίπτωση αν οι κλάσεις μπορούν να χωριστούν από 3 μη τέμνουσες ευθείες ο γραμμικός ταξινομητής SVM θα δουλέψει ικανοποιητικά.

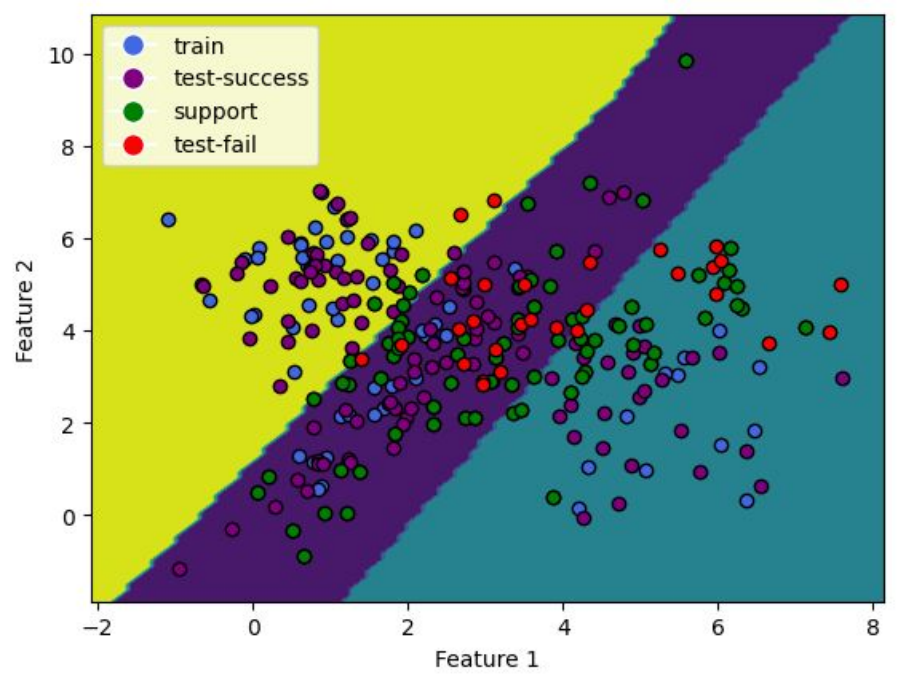
Αν όχι, θα χρησιμοποιηθεί πυρήνας RBF με τον οποίο αυξάνουμε τις διαστάσεις των δεδομένων μας και βρίσκουμε ένα επίπεδο ή υπερεπίπεδο που τα χωρίζει ικανοποιητικά (δηλαδή τα κάνουμε γραμμικά διαχωρίσιμα σε έναν χώρο μεγαλύτερης διάστασης).

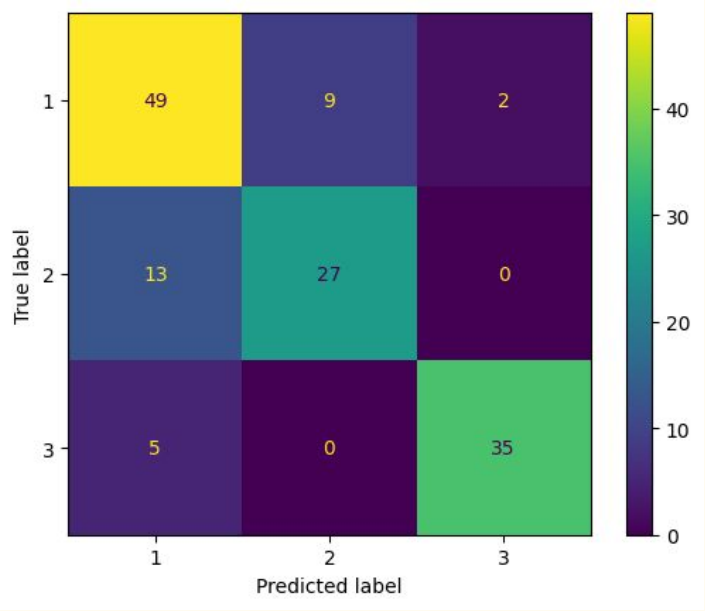


SVM: Πειράματα

1. Linear SVM
2. Grid Search και cross-validation σε RBF
3. Default sklearn SVM

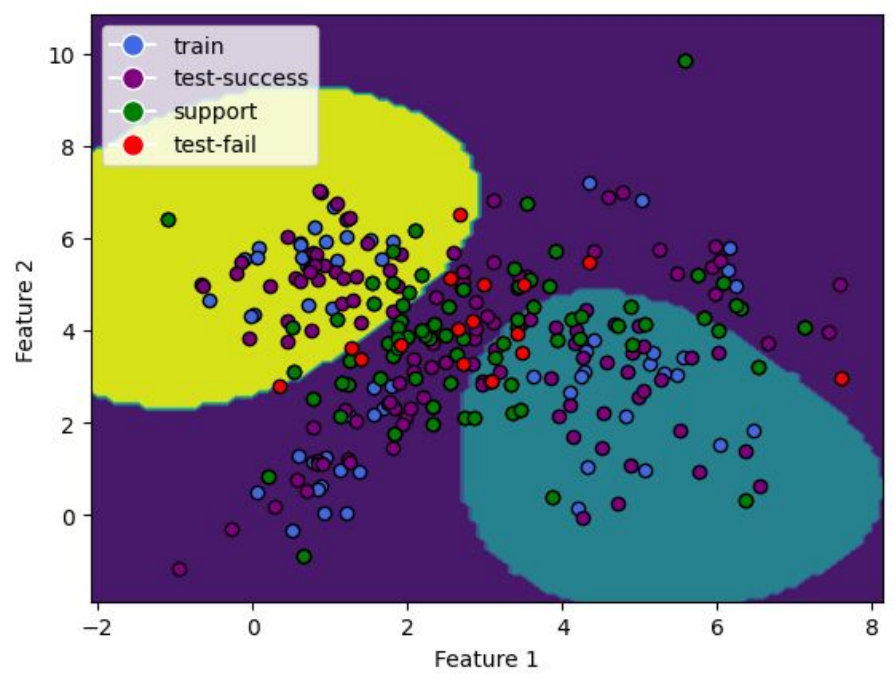
SVM: Linear Αποτελέσματα

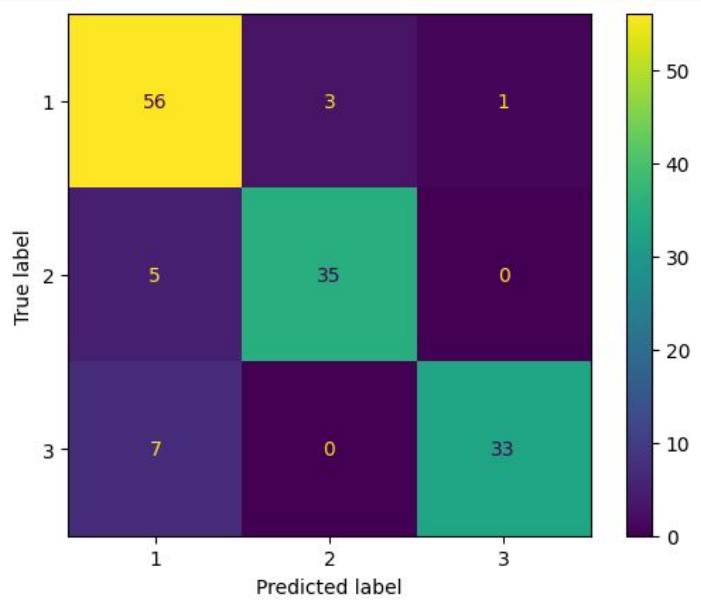




Ακρίβεια: 79.29%

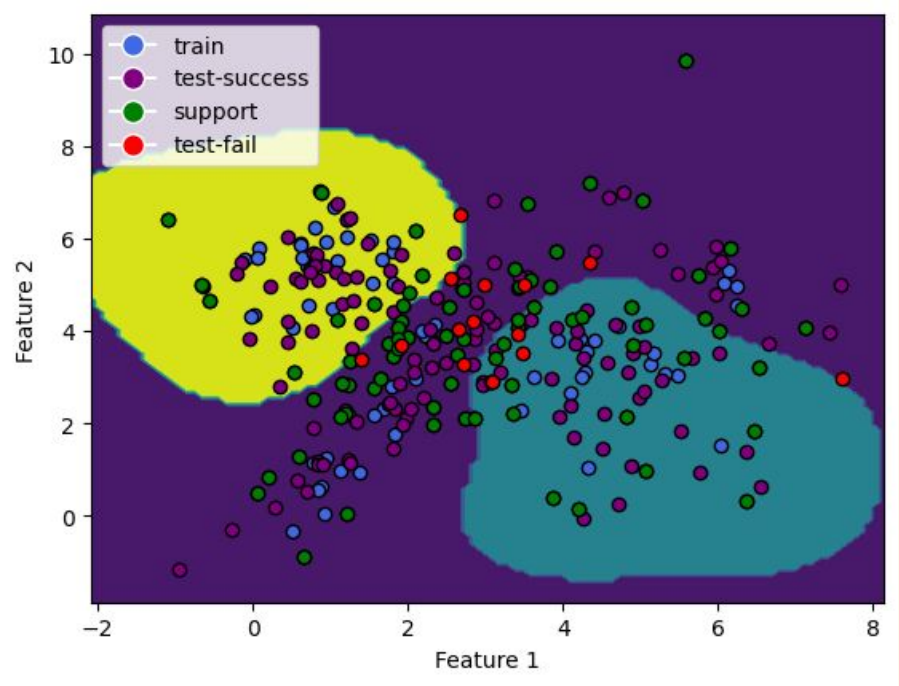
SVM: RBF Αποτελέσματα

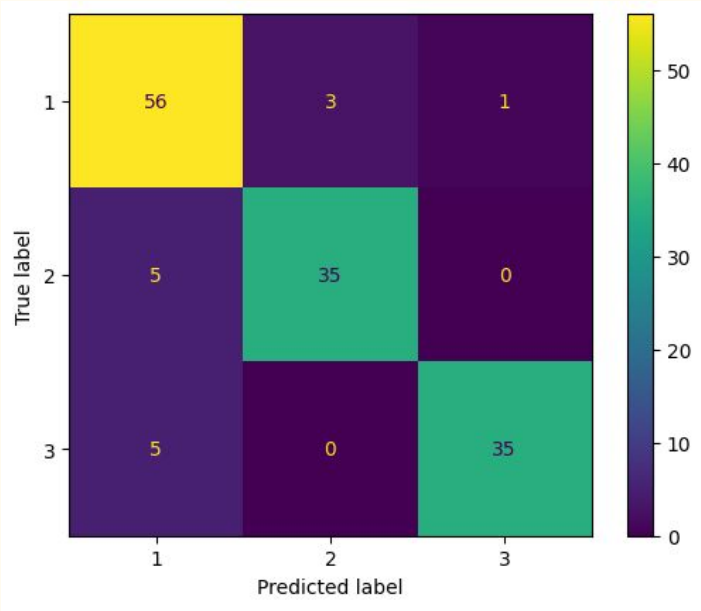




Ακρίβεια: 88.57%
C=0.1, gamma=0.1

SVM: Default RBF Αποτελέσματα





Ακρίβεια: 90.00%

SVM: Συμπεράσματα

Ο Γραμμικός SVM δεν λειτούργησε πολύ καλά, παρόλα αυτά έφτασε ~80% ακρίβεια, 2,5% χειρότερα από τον Maximum Likelihood.

Για τον RBF Kernel δοκιμάστηκαν C = [0.1, 1, 10, 100] και gamma = [‘auto’, 0.01, 0,1, 1, 10] με καλύτερα τα C=1 και gamma=0.1, πετυχαίνοντας ακρίβεια 88.57%.

Για πληρότητα χρησιμοποιείται και ο default SVM ταξινομητής της sklearn με απόδοση 90.00%.

2ο μέρος

Data Preprocessing

—

Data Preprocessing

Πριν την εκπαίδευση των διάφορων μοντέλων προ-επεξεργαζόμαστε τα δεδομένα για να πετύχουμε:

- Κανονικοποίηση (με StandardScaler)
- Dimensionality reduction (με PCA)
- Outlier detection (με reconstruction error από τα PCs)

Τα δεδομένα χωρίστηκαν σε train και validation sets με ποσοστό 80-20.

Dimensionality Reduction

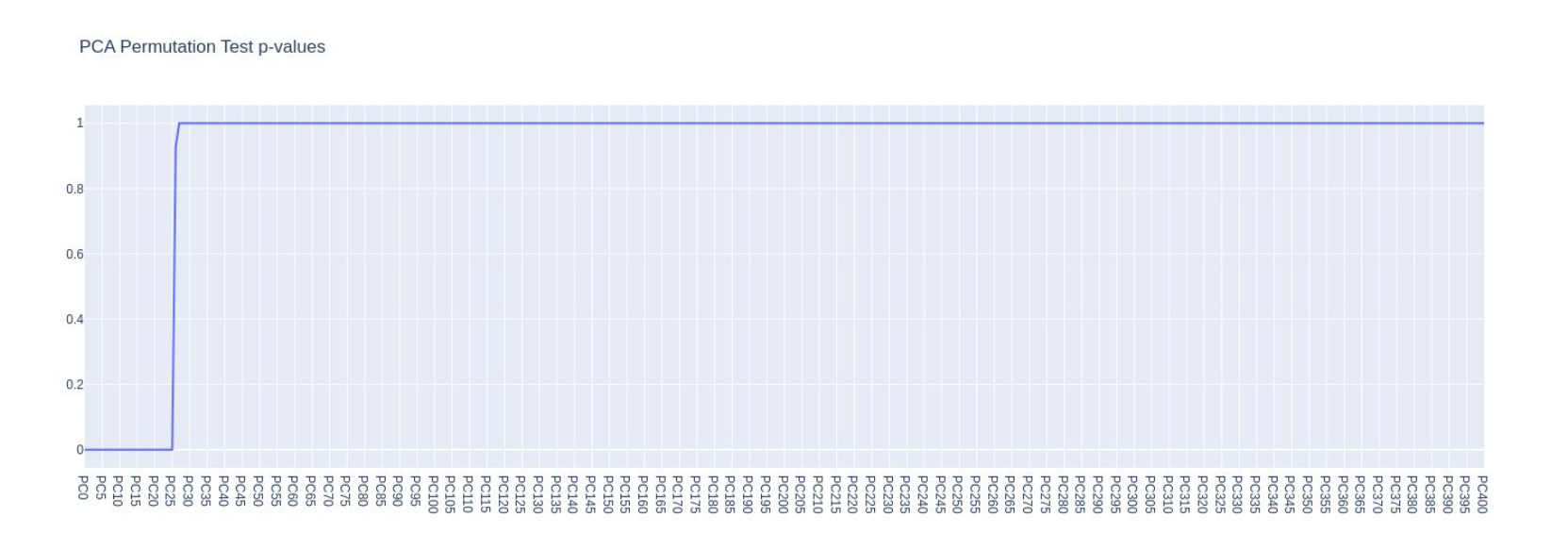
Για την σωστή εφαρμογή του PCA δεν αρκεί απλώς η αρχική explained_variance του dataset, αλλά έχει σημασία και η συσχέτιση των features μεταξύ τους.

Εκτελώντας permutation tests (N=100) στο dataset και εφαρμόζοντας ξανά PCA σε κάθε permutation κρατάμε μόνο τα PCs όπου η καινούρια explained_variance είναι μικρότερη από την αρχική.

Με αυτό τον τρόπο επιλέγουμε τα PCs που η explained_variance έχει σημαντικό αντίκτυπο στα features χωρίς να χρησιμοποιούμε κάποιο αυθαίρετο κατώφλι για την διακύμανση.

Vieira, Vasco, Permutation tests to estimate significances on Principal Components Analysis (2012). Computational Ecology and Software. 2. 103–123.

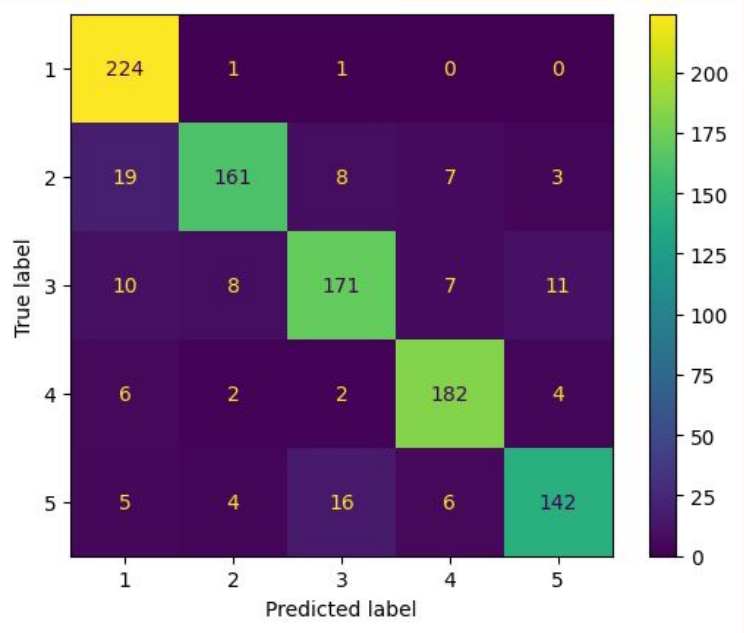
Permutation Test Results



k-NN

kNN

Μετά από grid search για την επιλογή των υπερ-παραμέτρων του kNN, επιλέγεται ο ταξινομητής με k=16, βάρη ανάλογα της απόστασης και χρήση της ευκλίδειας απόστασης.



Ακρίβεια: 87.80%

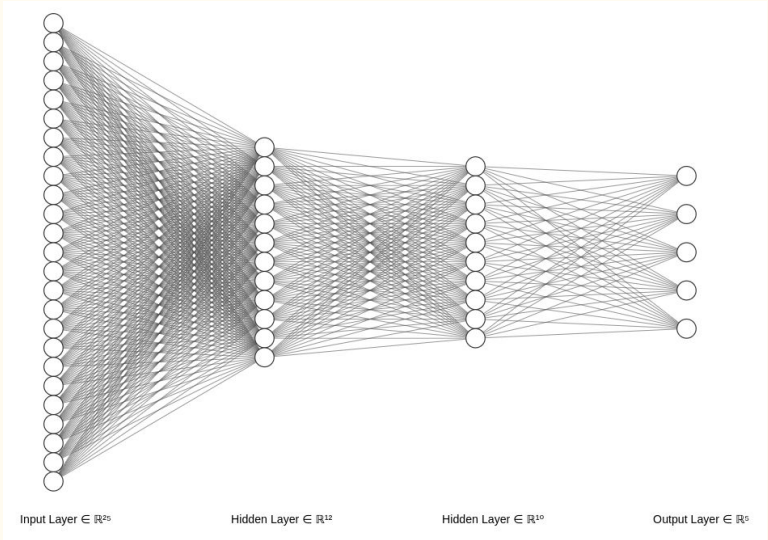
MLP

MLP

Δοκιμάστηκε η χρήση Multi-Layer Perceptron με διάφορα learning rates και αρχιτεκτονική ανάλογη του αριθμού των features και των διαφορετικών κλάσεων.

Κάθε μοντέλο εκπαιδεύτηκε για 100 εποχές και επιλέγεται το καλύτερο learning rate και epoch με βάση το validation set

Ακρίβεια: 86.13%

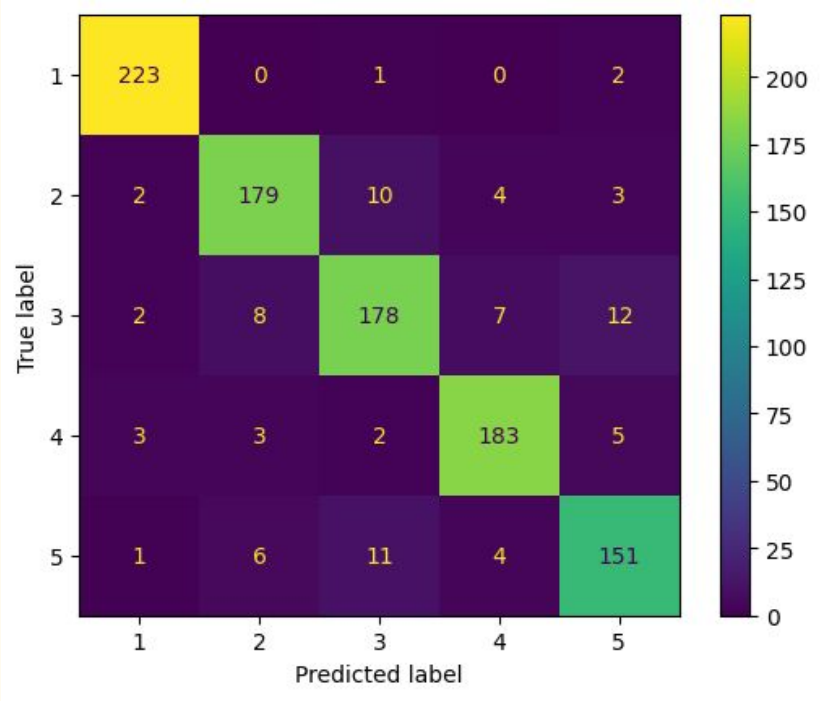


Final Model: SVM

—

SVM

Τελικά χρησιμοποιήθηκε SVM RBF ταξινομητής με C = 1.0 και gamma=’auto’, έπειτα από grid search



Ακρίβεια: 91.40%