

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Πολυτεχνική Σχολή
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΒΑΘΙΑ ΜΑΘΗΣΗ

Support Vector Machines

Κουκουλέτσου Αικατερίνη 10218

Δεκέμβριος 2024

Εισαγωγή

Η παρούσα εργασία παρουσιάζει την δημιουργία ενός Support Vector Machine - SVM τύπο νευρωνικού δικτύου με στόχο την επίλυση του προβλήματος Image Classification στο σετ δεδομένων CIFAR-10. Η εργασία αποτελεί συνέχεια της επίλυσης του προβλήματος ταξινόμησης του dataset, μετά την υλοποίηση των k-Nearest Neighbors, Nearest Centroid Classifier, Convolutional Neural Network και Multilayer Perceptron. Αποτελεί το πρότελευταίο βήμα πριν την ολοκλήρωση που θα έρθει με την υλοποίηση του Radial Basis Function νευρωνικού δικτύου.

Η προσέγγιση που ακολουθήθηκε χωρίζεται σε τρία στάδια. Στο πρώτο στάδιο, πραγματοποιήθηκε εκτενής ανάλυση με διάφορες παραμέτρους και kernels για τον εντοπισμό της βέλτιστης αρχιτεκτονικής του SVM, προκειμένου να επιλυθεί το πρόβλημα ταξινόμησης του συνόλου CIFAR-10. Στο δεύτερο στάδιο, αφού καθορίστηκε η αρχιτεκτονική, η προσοχή επικεντρώθηκε στην προεπεξεργασία των δεδομένων, καθώς και αυτά τα βήματα έχουν σημαντική επίδραση στην απόδοση του μοντέλου. Στο τρίτο στάδιο, χρησιμοποιήθηκε το CNN που αναπτύχθηκε στην προηγούμενη εργασία, καθώς και το έτοιμο μοντέλο EfficientNetV2, ως feature extractors, με το SVM να πραγματοποιεί την τελική ταξινόμηση βασιζόμενο στην έξοδο τους.

One-vs-One ή One-vs-Rest Approach

Καθώς τα μοντέλα SVM σχεδιάζονται αρχικά για την επίλυση Binary Classification προβλημάτων, απαιτείται τροποποίηση όταν αυτά επεκτείνονται σε Multiclass Classification Problems όπως αυτό της CIFAR-10. Οι κύριες προσεγγίσεις που μπορούν να ακολουθηθούν είναι η μέθοδος One-vs-One (OvO) και η μέθοδος One-vs-Rest (OvR).

One-vs-One (OvO):

Στην προσέγγιση One-vs-One, για κάθε ζεύγος κατηγοριών εκπαιδεύεται ένα ξεχωριστό SVM μοντέλο. Δηλαδή, αν το πρόβλημα έχει k κλάσεις δημιουργούνται $\frac{k(k-1)}{2}$ ξεχωριστά SVM μοντέλα, τα οποία εκπαιδεύονται για να διαχωρίσουν κάθε ζεύγος κατηγοριών. Στη συνέχεια, κατά την πρόβλεψη, κάθε μοντέλο «ψηφίζει» για την κατηγορία στην οποία ανήκει το δείγμα, και η τελική απόφαση βασίζεται στις περισσότερες ψήφους.

One-vs-Rest (OvR):

Αντίθετα, στην προσέγγιση One-vs-Rest, για κάθε κατηγορία εκπαιδεύεται ένα ξεχωριστό SVM μοντέλο που διαχωρίζει την κατηγορία αυτή από όλες τις υπόλοιπες. Συνεπώς, αν το πρόβλημα έχει k κατηγορίες, δημιουργούνται k μοντέλα. Κατά την πρόβλεψη, το μοντέλο που αποδίδει τη μεγαλύτερη «πιστότητα» (ή έχει τη μεγαλύτερη πιθανότητα) αποφασίζει ποια κατηγορία θα ανατεθεί στο δείγμα.

Για ένα πρόβλημα με 10 κατηγορίες, η One-vs-One προσέγγιση απαιτεί την εκπαίδευση 45 μοντέλων ενώ η One-vs-Rest προσέγγιση απαιτεί μόνο 10 μοντέλα. Επομένως, και κυρίως λόγω της ανάγκης για οικονομία στους υπολογιστικούς πόρους, επιλέχθηκε η προσέγγιση One-vs-Rest.

Προεπεξεργασία Δεδομένων

Για την προεπεξεργασία των δεδομένων εφαρμόστηκε κανονικοποίηση και reshaping. Δεν χρειάστηκε να γίνει διαχωρισμός του dataset, καθώς το CIFAR-10 παρέχει έτοιμα τα σύνολα εκπαίδευσης και ελέγχου. Αναλυτικότερες πληροφορίες για τα προεπεξεργαστικά βήματα παρατίθενται στην προηγούμενη **αναφορά της ενδιάμεσης εργασίας** και παραλείπονται εδώ για αποφυγή επαναληψιμότητας. Παρουσιάζονται μόνο προεπεξεργαστικές τεχνικές που δεν έχουν ήδη παρουσιαστεί.

Label Binarization

Η διαδικασία της *Label Binarization* μετατρέπει τις multiclass ετικέτες σε δυαδικά -binary- διανύσματα. Συγκεκριμένα, για κάθε ετικέτα, δημιουργείται ένα διάνυσμα διαστάσεων $1 \times N$, όπου N είναι ο αριθμός των κλάσεων (άρα $N=10$). Τα στοιχεία του διανύσματος έχουν όλα την τιμή 0 πλην του στοιχείου που αντιστοιχεί στην κλάση στην οποία ανήκει το δείγμα.

Αυτό επιτρέπει στο σύστημα να επεξεργαστεί την κάθε κατηγορία ως δυαδικό πρόβλημα, όπου το 1 σηματοδοτεί ότι το δείγμα ανήκει στην εν λόγω κατηγορία και το 0 ότι δεν ανήκει. Όπως είναι προφανές, πρόκειται για ένα προεπεξεργαστικό βήμα που είναι απολύτως απαραίτητο σε μια εφαρμογή SVM - One vs The Rest.

Μετρικές Αξιολόγησης

Οι Μετρικές Αξιολόγησης που χρησιμοποιούνται στην παρούσα εργασία περιλαμβάνουν το Confusion Matrix, το Accuracy Score και Classification Report. Αυτές οι μετρικές έχουν ήδη αναλυθεί λεπτομερώς στις εργασίες των δύο προηγούμενων αναφορών. Για λόγους αποφυγής επανάληψης, η αναλυτική παρουσίασή τους παραλείπεται από την παρούσα αναφορά.

Επιλογή του Kernel

Το kernel στα Support Vector Machines (SVM) επιτρέπει την εφαρμογή μη γραμμικών διαχωριστικών υπερεπιπέδων, μετασχηματίζοντας τα δεδομένα σε έναν υψηλότερης διάστασης χώρο. Με αυτόν τον τρόπο, τα SVM μπορούν να λύσουν προβλήματα που δεν είναι γραμμικά διαχωρίσιμα στον αρχικό χώρο.

1. Linear Kernel

$$K(x, y) = x^T y$$

Το linear kernel χρησιμοποιείται όταν τα δεδομένα είναι γραμμικά διαχωρίσιμα, δηλαδή μπορούν να διαχωριστούν με μια ευθεία γραμμή ή υπερεπίπεδο στον αρχικό χώρο.

2. Polynomial Kernel

$$K(x, y) = (x^T y + c)^d$$

Το poly kernel μετατρέπει τα δεδομένα σε έναν υψηλότερο χώρο διάστασης και επιτρέπει πιο σύνθετα, μη γραμμικά όρια απόφασης. Ο βαθμός του πολυωνύμου d ελέγχει την πολυπλοκότητα του ορίου απόφασης, και το c είναι μια σταθερά που μπορεί να ρυθμιστεί.

3. Radial Basis Function (RBF) Kernel

$$K(x, y) = \exp(-\gamma \|x - y\|^2)$$

Το RBF kernel μετατρέπει τα δεδομένα σε έναν άπειρο-διάστατο χώρο και είναι πολύ αποτελεσματικός στην αντιμετώπιση περιπτώσεων όπου το όριο απόφασης είναι ισχυρά μη γραμμικό. Η παράμετρος γ ελέγχει το εύρος της συνάρτησης Γαουσιανής, καθορίζοντας πόσο μακριά φτάνει η επιρροή ενός μόνο παραδείγματος εκπαίδευσης. Λειτουργεί καλά όταν τα δεδομένα έχουν κυκλικό ή σφαιρικό όριο απόφασης.

4. Sigmoid Kernel

$$K(x, y) = \tanh(\alpha x^T y + c)$$

Το sigmoid kernel χρησιμοποιεί τη συνάρτηση υπερβολικής συνάρτησης (sigmoid) για να μετασχηματίσει τα δεδομένα, παρόμοια με την ενεργοποίηση σε νευρωνικά δίκτυα, και χρησιμοποιείται για προβλήματα με πολύπλοκες, μη γραμμικές σχέσεις.

Α Μέρος

Στο πρώτο μέρος της εργασίας πραγματοποιήθηκαν εκτενείς αναλύσεις χρησιμοποιώντας διάφορα kernels και παραμέτρους (C , γ , coef0 κλπ.), με σκοπό την αξιολόγηση των επιδόσεων του μοντέλου σε διαφορετικές ρυθμίσεις.

Πριν την παρουσίαση των μοντέλων είναι απαραίτητη η διασάφιση κάποιων πολύ σημαντικών παραμέτρων και εννοιών.

1. **Η παράμετρος C** στο SVM (Regularization Parameter) ελέγχει την ισορροπία μεταξύ της μεγιστοποίησης του περιθωρίου και της ελαχιστοποίησης του σφάλματος ταξινόμησης. Σε γραμμικούς SVM, ένα μικρό C επιτρέπει μεγαλύτερο margin και περισσότερα σφάλματα, οδηγώντας σε πιο γενικευμένο μοντέλο, ενώ ένα μεγάλο C ελαχιστοποιεί τα σφάλματα. Στα μη γραμμικά kernels, όπως το RBF, το C επηρεάζει το πόσο στενά ακολουθεί ο ταξινομητής την τοπική γεωμετρία των δεδομένων, επηρεάζοντας την πολυπλοκότητα του συνόρου απόφασης και τη γενίκευση σε άγνωστα δεδομένα.
2. **Η παράμετρος γ** στα RBF και Sigmoid kernels (Kernel Coefficient), ρυθμίζει την κλίμακα της επιρροής κάθε σημείου δεδομένων. Δηλαδή το γ καθορίζει τη λεπτομέρεια με την οποία εξετάζεται η κατανομή των δεδομένων, ειδικά σε μη γραμμικά kernel.
3. **Η παράμετρος degree** στο πολυωνυμικό kernel ελέγχει τον βαθμό του πολυωνύμου που χρησιμοποιείται για να μετασχηματίσει τα δεδομένα στον χαρακτηριστικό χώρο.
4. **Η παράμετρος coef0** στον πολυωνυμικό και στο sigmoid kernel (Independent term in kernel function) καθορίζει τη συμβολή της σταθεράς. Στον πολυωνυμικό kernel, το coef0 ρυθμίζει την επίδραση της σταθερής τιμής που προστίθεται στη συνάρτηση του πολυωνύμου, επηρεάζοντας έτσι την "καμπυλότητα" του ορίου απόφασης.

Γραμμικό Kernel

1. Μοντέλο 1:

Kernel	C parameter	Accuracy
Linear	1.0	0.38

2. Μοντέλο 2:

Kernel	C parameter	Accuracy
Linear	10.0	0.36

Αποφασίστηκε να μην πραγματοποιηθούν περαιτέρω αναλύσεις με γραμμικό πυρήνα, καθώς είναι σαφές ότι τα δεδομένα δεν είναι γραμμικά διαχωρίσιμα και, συνεπώς, δεν αναμένεται να προκύψει βέλτιστο αποτέλεσμα από αυτήν τη μέθοδο. Οι αναλύσεις με γραμμικό πυρήνα πραγματοποιήθηκαν αποκλειστικά για λόγους πληρότητας της παρουσίασης και για να εξεταστεί η συμπεριφορά του μοντέλου σε αυτήν την περίπτωση.

Υπολογιστικός Χρόνος 113 λεπτά και 440 λεπτά αντίστοιχα.

Πολυωνυμικό Kernel

1. Μοντέλο 1:

Kernel	C parameter	Degree	Accuracy
Polynomial	1.0	3	0.52

2. Μοντέλο 2:

Kernel	C parameter	Degree	Accuracy
Polynomial	1.0	5	0.50

3. Μοντέλο 3:

Kernel	C parameter	Degree	coef0	Accuracy
Polynomial	1.0	2	1.0	0.52

4. Μοντέλο 4:

Kernel	C parameter	Degree	coef0	Accuracy
Polynomial	10.0	2	1.0	0.52

Παρατηρείται σημαντική βελτίωση στην απόδοση, με τα ποσοστά ακρίβειας να φτάνουν πλέον το 50%. Από τα αποτελέσματα, διαπιστώνεται ότι για τον βαθμό του πολυωνύμου που ανταποκρίνεται καλύτερα στην περίπτωση, δεν απαιτούνται πολύ υψηλές τιμές. Αρκεί η χρήση ενός δευτεροβάθμιου ή τριτοβάθμιου πολυωνύμου. Ωστόσο, αυτό δεν συνιστά το βέλτιστο αποτέλεσμα που εντοπίστηκε.

Υπολογιστικός χρόνος 63 λεπτά, 55 λεπτά, 62 λεπτά και 76 λεπτά αντίστοιχα.

RBF Kernel

1. Μοντέλο 1:

Kernel	C parameter	γ parameter	Accuracy
RBF	1.0	0.1	0.22

2. Μοντέλο 2:

Kernel	C parameter	γ parameter	Accuracy
RBF	10.0	0.1	0.24

3. Μοντέλο 3:

Kernel	C parameter	γ parameter	Accuracy
RBF	1.0	0.01	0.56

4. Μοντέλο 4:

Kernel	C parameter	γ parameter	Accuracy
RBF	1.0	scale	0.54

5. Μοντέλο 5:

Kernel	C parameter	γ parameter	Accuracy
RBF	10.0	scale	0.57

Εδώ είναι και το βέλτιστο μοντέλο, το μοντέλο 5 με Accuracy = 0.57. Με C=10 και $\gamma = \text{scale}$. Αυτό είναι και το μοντέλο που θα χρησιμοποιηθεί στην υπόλοιπη εργασία.

Σε αυτό το σημείο είναι σημαντικό να διευκρινιστεί ότι όταν ορίζεται $\gamma = \text{scale}$, η τιμή του γ υπολογίζεται αυτόματα ως ο αντίστροφος του αριθμού των χαρακτηριστικών του δεδομένου συνόλου, δηλαδή:

$$\gamma = \frac{1}{n \cdot \text{var}(X)}$$

όπου n είναι ο αριθμός των χαρακτηριστικών στα δεδομένα X και $\text{var}(X)$ είναι η διακύμανση των δεδομένων εισόδου.

Υπολογιστικός χρόνος 252 λεπτά, 320 λεπτά, 77 λεπτά, 65 λεπτά,

Sigmoid Kernel

1. Μοντέλο 1:

Kernel	C parameter	γ parameter	coef0	Accuracy
Sigmoid	1.0	0.1	0	0.10

2. Μοντέλο 2:

Kernel	C parameter	γ parameter	coef0	Accuracy
Sigmoid	10.0	0.1	0	0.10

3. Μοντέλο 3:

Kernel	C parameter	γ parameter	coef0	Accuracy
Sigmoid	10.0	scale	0	0.08

Υπολογιστικός χρόνος 118 λεπτά, 114 λεπτά και 65 λεπτά αντίστοιχα.

Μέρος Β

Καθώς έχει προσδιοριστεί η βέλτιστη αρχιτεκτονική του SVM, η προσοχή επικεντρώνεται στην προεπεξεργασία των δεδομένων και στον τρόπο συνδυασμού των διαφόρων προεπεξεργαστικών βημάτων για την επίτευξη της καλύτερης δυνατής απόδοσης.

Τα προεπεξεργαστικά βήματα που τελικά επιλέχθηκαν -καθώς απέδωσαν τα καλύτερα αποτελέσματα- και εφαρμόστηκαν στο μοντέλο, είναι τα ακόλουθα.

Standard Scaler

Το StandardScaler εφαρμόζει z-score normalization. Δηλαδή κάθε χαρακτηριστικό αποκτά μέσο όρο 0 και τυπική απόκλιση 1. Ο αλγόριθμος είναι ο εξής, για κάθε χαρακτηριστικό, υπολογίζεται ο μέσος όρος μ και η τυπική απόκλιση σ από το σύνολο εκπαίδευσης. Κάθε τιμή x ενός χαρακτηριστικού μετασχηματίζεται χρησιμοποιώντας τον τύπο:

$$z = \frac{x - \mu}{\sigma}$$

Στην συνέχεια, τα ίδια (μ και σ) που υπολογίστηκαν από το σύνολο εκπαίδευσης εφαρμόζονται και στα δεδομένα ελέγχου. Με αυτόν τον τρόπο, εξασφαλίζεται ότι όλα τα χαρακτηριστικά να συνεισφέρουν ισότιμα στην ταξινόμηση.

Histogram of Oriented Gradients

Αρχικά η εικόνα μετατρέπεται από RGB σε Grayscale, μειώνοντας τις διαστάσεις από 3 κανάλια σε 1. Για κάθε pixel της εικόνας, υπολογίζονται οι τοπικές αλλαγές φωτεινότητας $\frac{\partial I}{\partial x}$ και $\frac{\partial I}{\partial y}$ (οριζόντιος και κατακόρυφος gradient). Οι αλλαγές αυτές αποτυπώνουν την κατεύθυνση και το μέγεθος της μετάβασης από σκούρο σε φωτεινό. Στην συνέχεια, από τα gradients, υπολογίζεται το μέγεθος και η κατεύθυνση για κάθε pixel:

$$\text{Magnitude: } \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}, \text{ Orientation: } \theta = \arctan\left(\frac{\partial I}{\partial y} / \frac{\partial I}{\partial x}\right)$$

Η εικόνα χωρίζεται σε μικρές περιοχές, που ονομάζονται *cells* (π.χ., 8x8 pixels). Σε κάθε *cell*, δημιουργείται ένα ιστόγραμμα που μετρά την κατανομή των κατευθύνσεων των gradients. Πολλά *cells* ομαδοποιούνται σε *blocks* (π.χ., 2x2 cells), ώστε να υπολογιστεί η κανονικοποίηση των χαρακτηριστικών. Τα κανονικοποιημένα διανύσματα των *blocks* συνενώνονται, σχηματίζοντας το τελικό διάνυσμα χαρακτηριστικών της εικόνας.

PCA (Principal Component Analysis):

Αρχικά, τα δεδομένα μετασχηματίζονται σύμφωνα με τη μέθοδο κανονικοποίησης z-score που παρουσιάστηκε παραπάνω. Έπειτα, δημιουργείται ο πίνακας συνδιακύμανσης C (ο οποίος περιγράφει τη συσχέτιση μεταξύ των χαρακτηριστικών):

$$C = \frac{1}{n-1} X^T X$$

Αμέσως μετά, υπολογίζονται οι ιδιοτιμές λ_i και τα ιδιοδιανύσματα v_i του πίνακα συνδιακύμανσης C . Η ιδιοτιμή λ_i μετράει πόση πληροφορία (δηλαδή διακύμανση) των δεδομένων

βρίσκεται στην κύρια συνιστώσα v_i . Όσο μεγαλύτερη είναι η ιδιοτιμή, τόσο περισσότερη πληροφορία κωδικοποιείται από αυτήν τη διάσταση.

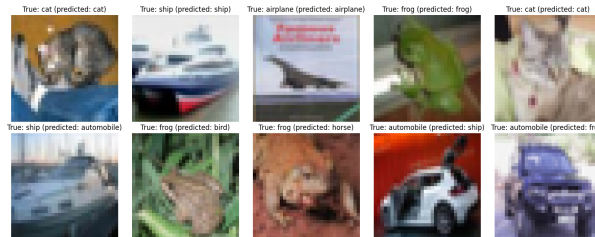
Επομένως, οι ιδιοτιμές ταξινομούνται κατά φθίνουσα σειρά και επιλέγονται τα k πρώτα ιδιοδιανύσματα που αντιστοιχούν στη μεγαλύτερη διακύμανση (στην συγκεκριμένη εργασία ορίστηκε να διατηρείται το 95% της διακύμανσης). Τέλος, τα αρχικά δεδομένα X προβάλλονται στις κύριες συνιστώσες, δημιουργώντας έναν χαμηλότερων διαστάσεων χώρο:

$$X_{PCA} = XW_k$$

όπου W_k είναι ο πίνακας που περιέχει τα k ιδιοδιανύσματα ως στήλες.

Αποτελέσματα Βέλτιστου Μοντέλου

Με την εφαρμογή των τριων παραπάνω προεπεξεργαστικών βημάτων, η απόδοση του μοντέλου παρουσίασε άνοδο από το 57% στο 64%. Παρατίθενται οι μετρικές αξιολόγησης καθώς και παραδείγματα σωστής και λανθασμένης ταξινόμησης.



	0	1	2	3	4	5	6	7	8	9
0	737	17	60	20	31	10	14	11	80	20
1	28	770	7	22	16	5	17	8	55	72
2	77	7	508	70	108	88	75	32	28	7
3	46	22	94	446	74	162	82	48	9	17
4	43	12	98	74	572	54	64	59	10	14
5	23	9	81	165	70	516	51	61	11	13
6	23	17	47	59	60	45	707	13	20	9
7	18	7	58	64	86	60	15	656	6	30
8	107	55	25	13	13	8	11	13	706	49
9	34	81	13	32	24	13	9	20	35	739
	0	1	2	3	4	5	6	7	8	9

Class	Precision	Recall	F1-Score	Support
Airplane	0.65	0.74	0.69	1000
Automobile	0.77	0.77	0.77	1000
Bird	0.51	0.51	0.51	1000
Cat	0.46	0.45	0.45	1000
Deer	0.54	0.57	0.56	1000
Dog	0.54	0.52	0.53	1000
Frog	0.68	0.71	0.69	1000
Horse	0.71	0.66	0.68	1000
Ship	0.74	0.71	0.72	1000
Truck	0.76	0.74	0.75	1000
Accuracy		0.64		10000
Macro Avg	0.64	0.64	0.64	10000
Weighted Avg	0.64	0.64	0.64	10000

Μερος Γ

Τελευταία προσέγγιση ήταν και ο συνδυασμός του SVM με συνελικτικό δίκτυο. Το CNN δίκτυο που χρησιμοποιήθηκε στην προσέγγιση A είναι αυτό που είχε αναπτυχθεί στην προηγούμενη εργασία.

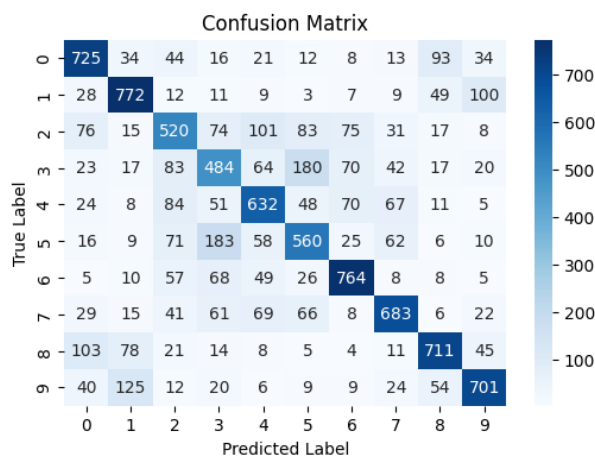
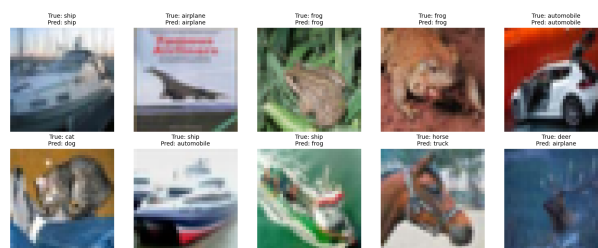
Προσέγγιση A

Αρχικά, επιλέχθηκε η προσέγγιση της εξαγωγής χαρακτηριστικών μέσω συνελικτικού δικτύου (CNN) και της ταξινόμησης με SVM. Η προσέγγιση αυτή προσφέρει σημαντικά πλεονεκτήματα καθώς εκμεταλλεύεται τα πλεονεκτήματα και των δύο τύπων.

Το CNN είναι πολύ δυνατό όσον αφορά την εξαγωγή χρήσιμων χαρακτηριστικών, εντοπίζοντας και αναγνωρίζοντας μοτίβα στην εικόνα μέσω της εφαρμογής του kernel. Παράλληλα, το SVM είναι άριστο στην ταξινόμηση, ακόμη και όταν πρόκειται για δεδομένα που δεν είναι γραμμικά διαχωρίσιμα αφού χρησιμοποιεί το kernel και μετασχηματίζει τα δεδομένα σε υψηλότερη διάσταση όπου η ταξινόμηση μπορεί να γίνει πλέον με ακρίβεια.

Προς αυτήν την κατεύθυνση, το SVM προσφέρει χαμηλότερο υπολογιστικό κόστος, καθώς είναι ταχύτερο στην ταξινόμηση και ικανό να διαχειριστεί αποδοτικά τις μεγάλες διαστάσεις των δεδομένων. Επομένως, αποφασίστηκε πως η συγκεκριμένη συνδυαστική προσέγγιση έχει ενδιαφέρον και αξίζει να μελετηθεί.

Το Accuracy που έδωσε αυτή η διαδικασία είναι 65.52% και ο συνολικός χρόνος εκτέλεσης είναι τα 20 λεπτά. Παρατίθενται οι μετρικές αξιολόγησης καθώς και τα παραδείγματα ορθής και εσφαλμένης ταξινόμησης.



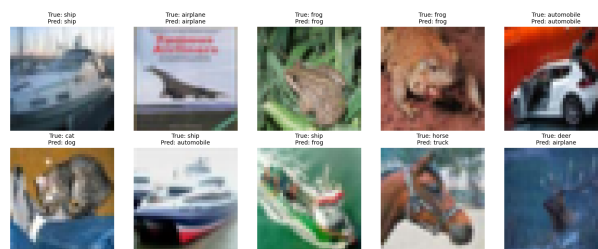
Class	Precision	Recall	F1-Score	Support
Airplane	0.68	0.72	0.70	1000
Automobile	0.71	0.77	0.74	1000
Bird	0.55	0.52	0.53	1000
Cat	0.49	0.48	0.49	1000
Deer	0.62	0.63	0.63	1000
Dog	0.56	0.56	0.56	1000
Frog	0.73	0.76	0.75	1000
Horse	0.72	0.68	0.70	1000
Ship	0.73	0.71	0.72	1000
Truck	0.74	0.70	0.72	1000
Accuracy		0.66		10000
Macro Avg	0.65	0.66	0.65	10000
Weighted Avg	0.65	0.66	0.65	10000

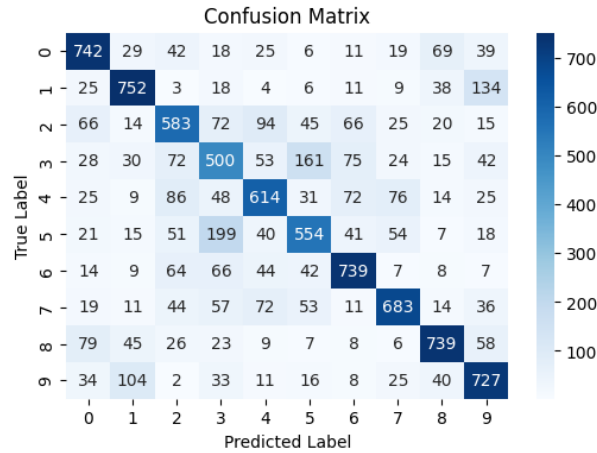
Προσέγγιση B

Η δεύτερη προσέγγιση αξιοποιεί το EfficientNetV2. Το EfficientNetV2 είναι ένα προεκπαιδευμένο συνελικτικό δίκτυο που έχει εκπαιδευτεί ήδη πάνω στο σετ δεδομένων ImageNet με αποτέλεσμα τα χαρακτηριστικά που εξάγονται να είναι πλούσια και γενικευμένα. Αυτό ξεπερνάει κατά πολύ την ποιότητα των χαρακτηριστικών που παράγονται από ένα απλό CNN εκπαιδευμένο αποκλειστικά στο CIFAR-10.

Παράλληλα, η αρχιτεκτονική του είναι σχεδιασμένη με βάση το compound scaling, επιτρέποντας την ταυτόχρονη αύξηση του βάθους, του πλάτους, και της ανάλυσης της εικόνας. Οι αναλογίες αυτές υπολογίζονται μέσω μιας μεθόδου αναζήτησης (neural architecture search), η οποία βρίσκει τον βέλτιστο συνδυασμό.

Το Accuracy που έδωσε αυτή η διαδικασία είναι 66.33% και ο συνολικός χρόνος εκτέλεσης είναι τα 20 λεπτά. Παρατίθενται οι μετρικές αξιολόγησης καθώς και τα παραδείγματα ορθής και εσφαλμένης ταξινόμησης.





Class	Precision	Recall	F1-Score	Support
Airplane	0.70	0.74	0.72	1000
Automobile	0.74	0.75	0.75	1000
Bird	0.60	0.58	0.59	1000
Cat	0.48	0.50	0.49	1000
Deer	0.64	0.61	0.62	1000
Dog	0.60	0.55	0.58	1000
Frog	0.71	0.74	0.72	1000
Horse	0.74	0.68	0.71	1000
Ship	0.77	0.74	0.75	1000
Truck	0.66	0.73	0.69	1000
Accuracy		0.66		10000
Macro Avg	0.66	0.66	0.66	10000
Weighted Avg	0.66	0.66	0.66	10000

Συνολική Αξιολόγηση

Προσέγγιση	Ακρίβεια (%)
SVM, C=10, $\gamma = \text{scale}$	57
Με Προεπεξεργαστικά Βήματα (HoG, PCA, Standard Scaler)	64
Με Συνελικτικό Μοντέλο Από την Προηγούμενη Εργασία (Feature Extraction)	65.52
Με EfficientNetV2 για Feature Extraction και SVM για Ταξινόμηση	66.33

Σύγκριση με kNN και NCC

Ο αλγόριθμος SVM (Support Vector Machine) φαίνεται να αποδίδει καλύτερα από το KNN και το NCC. Αυτό ίσως οφείλεται στην ικανότητά του να χειρίζεται πιο αποτελεσματικά τα περιθώρια διαφοροποίησης μεταξύ των κατηγοριών. Το SVM όπως αναφέρθηκε και προηγουμένως, επιλέγει το υπέρτατο υπερεπίπεδο που μεγιστοποιεί την απόσταση μεταξύ των

κατηγοριών, γεγονός που το καθιστά λιγότερο ευαίσθητο στις εξαιρέσεις και τους θορύβους των δεδομένων, σε αντίθεση με το KNN, το οποίο βασίζεται στην τοπική γειτνίαση και μπορεί να είναι επιρρεπές σε θορύβους ή ασυνήθιστες τιμές κοντινών σημείων. Το NCC, αν και απλός και γρήγορος, στηρίζεται στην υπόθεση ότι οι κατηγορίες είναι συμμετρικές και ισχυρά επικεντρωμένες γύρω από έναν κεντρικό μέσο όρο, κάτι που δεν ισχύει πάντα, ιδιαίτερα σε πολύπλοκα και μη γραμμικά διαχωρίσιμα δεδομένα, όπως το CIFAR-10. Αντίθετα, το SVM, ειδικά με χρήση του kernel RBF, μπορεί να εντοπίσει μη γραμμικά όρια, προσφέροντας καλύτερη γενίκευση και ανθεκτικότητα στις παραμορφώσεις των δεδομένων. Η ικανότητα του SVM να προσαρμόζεται σε πιο σύνθετα διαχωρίσιμα προβλήματα το καθιστά πιο αποδοτικό για τέτοια είδη δεδομένων, επιτυγχάνοντας υψηλότερη ακρίβεια σε σύγκριση με τους άλλους αλγόριθμους ταξινόμησης.

Σύγκριση με MLP

Το μοντέλο, MLP με μόνο ένα κρυφό επίπεδο, αποδίδει σε σχετικά χαμηλά ποσοστά (περίπου 20%), κάτι που είναι σημαντικά χαμηλότερο σε σύγκριση με το SVM με RBF kernel. Η βασική αιτία αυτής της χαμηλής απόδοσης ίσως μπορεί να αποδοθεί στην απλότητα της αρχιτεκτονικής του μοντέλου.

Το MLP με μόνο ένα κρυφό επίπεδο και 512 νευρώνες δεν έχει την ικανότητα να μάθει τις σύνθετες, μη γραμμικές σχέσεις που υπάρχουν στο σύνολο δεδομένων CIFAR-10. Αντιθέτως, το RBF kernel μπορεί να προβάλλει τα δεδομένα σε υψηλότερες διαστάσεις, επιτρέποντας μια πιο ακριβή διάκριση.

Συνολικά, το MLP με μόνο ένα κρυφό επίπεδο δεν είναι επαρκές για την επίλυση του προβλήματος ταξινόμησης του CIFAR-10 με την απαιτούμενη ακρίβεια, καθώς δεν έχει την απαραίτητη ικανότητα να κατανοήσει τις σύνθετες και μη γραμμικές σχέσεις που απαιτούνται.