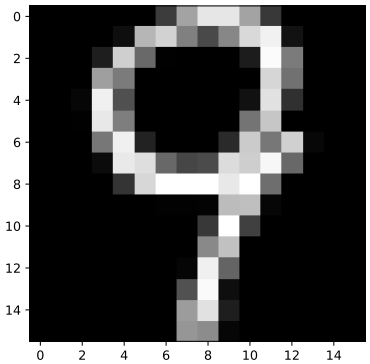
1 Εισαγωγη

Το περιεχόμενο αυτής της ενότητας αφορά τα Βήματα 1-9 της εργασίας.

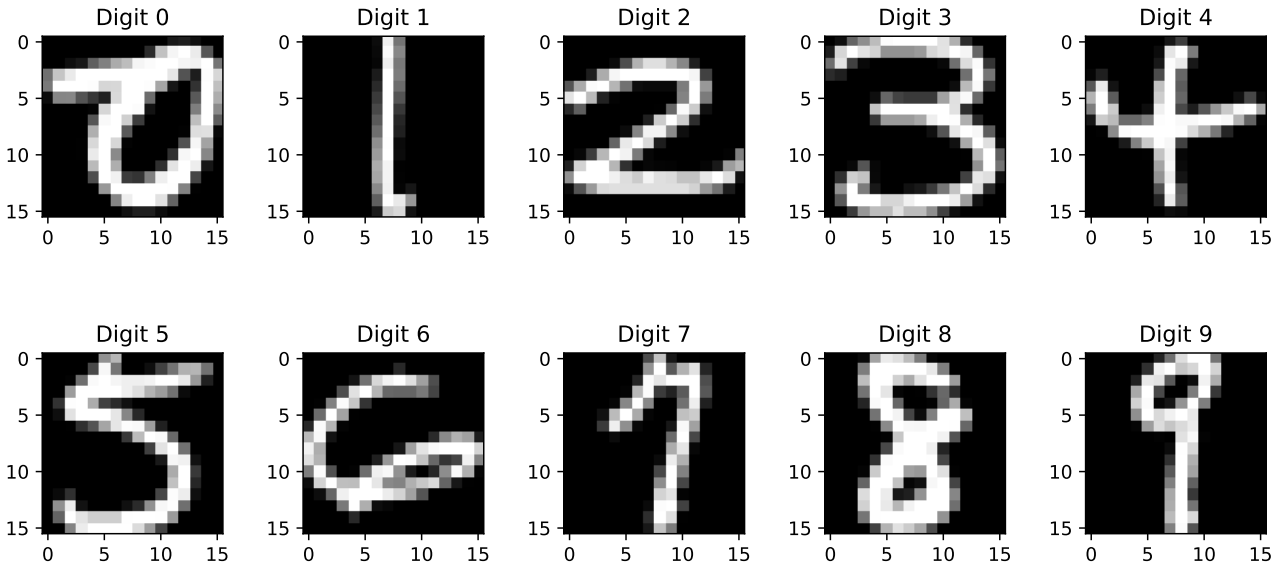
Το περιεχόμενο της πρώτης εργαστηριακής άσκησης και κατ' επέκταση της παρούσας εργαστηριακής αναφοράς αποτέλεσε η υλοποίηση συστημάτων οπτικής αναγνώρισης ψηφίων. Τα δεδομένα που χρησιμοποιήθηκαν προήλθαν από την US Postal Service και αποτελούν εικόνες ψηφίων από το 0 έως και το 9. Οι εικόνες αυτές έχουν διακριτοποιηθεί σε pixels, με την κάθε εικόνα να αποτελείται από 256 (16x16) συνολικά pixels, τα οποία είναι και τα χαρακτηριστικά των δεδομένων. Σε κάθε pixel έχει ανατεθεί μια αριθμητική τιμή από το -1.0 έως το 1.0, η οποία αντιστοιχεί σε κάποια απόχρωση του μαύρου (με το ένα άκρο να αντιστοιχεί στο μαύρο και το άλλο στο λευκό), ούτως ώστε τα ψηφία να μπορούν να αναπαρασταθούν ως ασπρόμαυρες εικόνες. Εκτός από τα χαρακτηριστικά των δεδομένων, παρέχονται και οι αντίστοιχες κλάσεις στις οποίες ανήκουν, ώστε να είναι εφικτή η κατασκευή μοντέλων χρησιμοποιώντας ένα υποσύνολο των δεδομένων για εκπαίδευση και κατόπιν η αξιολόγησή τους, χρησιμοποιώντας το συμπληρωματικό υποσύνολο.

Αρχικά (Βήμα 1), τα δεδομένα διαβάστηκαν υπό τη μορφή numpy arrays και χωρίστηκαν στα ζεύγη $(X, y)_{\text{train}}$ και $(X, y)_{\text{test}}$, για την εκπαίδευση και για την τελική αξιολόγηση, αντίστοιχα. Στα πλαίσια ενός απλού ελέγχου σχεδιάστηκε το ψηφίο στη θέση 131 των δεδομένων εκπαίδευσης (Βήμα 2 - βλ. Εικόνα 1.1), το οποίο είναι εμφανές πως αντιστοιχεί στον αριθμό 9. Προκειμένου να υπάρχει μια ιδέα σχετικά με το τι είδους δεδομένα υπάρχουν για κάθε κλάση (τύπος ψηφίου), σχεδιάστηκε στη συνέχεια (Βήμα 3) ένα τυχαίο δείγμα για κάθε κλάση (βλ. Εικόνα 1.2).

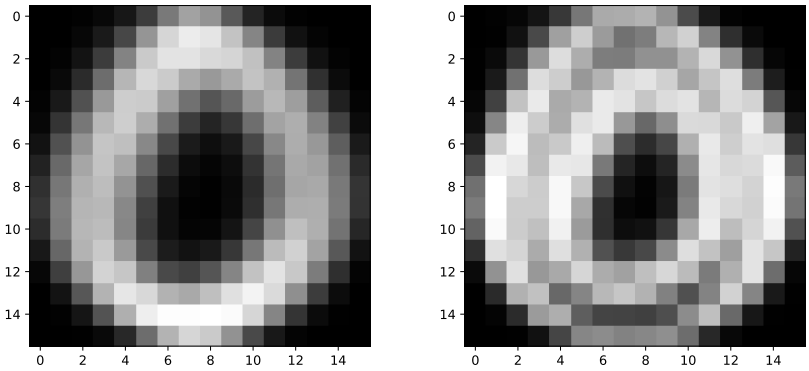


Εικόνα 1.1: Απεικόνιση του δεδομένου στη θέση 131 σε grayscale κλίμακα.

Έχοντας κατανοήσει το πώς τα αριθμητικά δεδομένα αντιστοιχίζονται σε εικόνες, το επόμενο βήμα της αρχικής επεξεργασίας των δεδομένων ήταν ο υπολογισμός χαρακτηριστικών που αφορούν ένα υποσύνολο ή το σύνολο των δεδομένων. Για παράδειγμα, υπολογίστηκε η μέση τιμή του χαρακτηριστικού που αντιστοιχεί στο pixel (10,10) για τα ψηφία που ανήκουν στην κλάση 0 (Βήμα 4) και βρέθηκε ίση με -0.504, καθώς και η αντίστοιχη διασπορά (Βήμα 5), η οποία βρέθηκε ίση με 0.524. Κατόπιν (Βήμα 6), η διαδικασία αυτή επαναλήφθηκε για κάθε pixel της κλάσης 0. Με βάση τα αποτελέσματα των υπολογισμών αυτών, σχεδιάστηκε το «μέσο 0» (Βήμα 7), δηλαδή η εικόνα που αντιστοιχεί στο ψηφίο 0 με βάση τις υπολογισμένες μέσες τιμές για κάθε pixel, καθώς και η διασπορά για το 0 (Βήμα 8 - βλ. Εικόνα 1.3). Όπως είναι αναμενόμενο, τα pixels όπου η ένταση είναι υψηλή (δηλαδή υπερισχύει το λευκό χρώμα) για την εικόνα του μέσου 0 αντιστοιχούν σε pixels χαμηλότερης έντασης για την εικόνα της διασποράς του 0. Αντίθετα, στην εικόνα της διασποράς για το 0, η ένταση είναι υψηλότερη στα pixels που βρίσκονται «γύρω» από το μέσο 0.

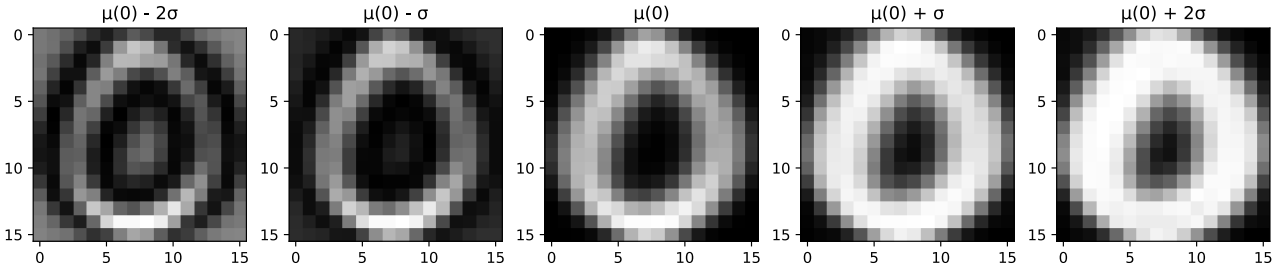


Εικόνα 1.2: Τυχαία επιλεγμένα δείγματα για κάθε κλάση από τα δεδομένα εκπαίδευσης.



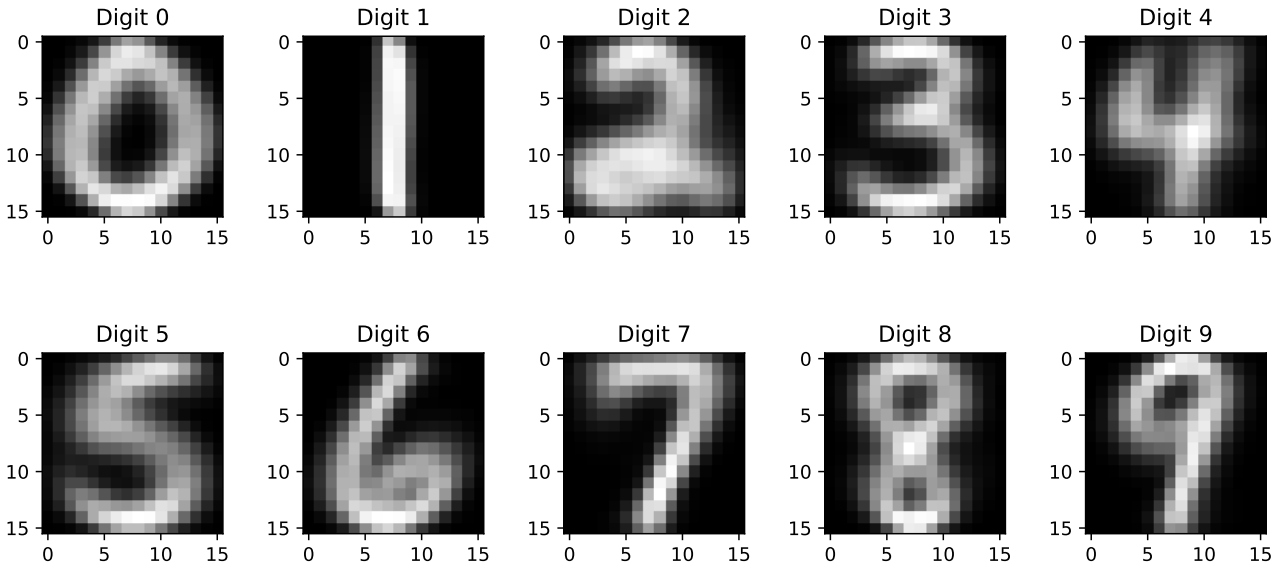
Εικόνα 1.3: Αριστερά: Το μέσο 0. Δεξιά: Η διασπορά του 0.

Συμβολίζοντας το μέσο 0 ως $\mu(0)$ και την τυπική απόκλιση του 0 ως σ (όπου η τυπική απόκλιση είναι η ρίζα της διασποράς), έχει σχεδιαστεί στην Εικόνα 1.4 το μέσο 0, μαζί με τα $\mu(0) \pm \sigma$ και $\mu(0) \pm 2\sigma$. Είναι εμφανές πως ακόμα και μερικές τυπικές αποκλίσεις μακριά από τη μέση τιμή, το ψηφίο παραμένει αναγνωρίσιμο.



Εικόνα 1.4: Οι εικόνες που αντιστοιχούν στα $\mu(0)$, $\mu(0) \pm \sigma$ και $\mu(0) \pm 2\sigma$.

Η προαναφερθείσα διαδικασία του υπολογισμού της μέσης τιμής και της διασποράς όλων των pixels ακολουθήθηκε και για τα υπόλοιπα ψηφία χρησιμοποιώντας τα δεδομένα εκπαίδευσης. Στην Εικόνα 1.5 απεικονίζονται τα αποτελέσματα για τη μέση τιμή κάθε κλάσης.



Εικόνα 1.5: Οι εικόνες που αντιστοιχούν στη μέση τιμή κάθε ψηφίου.

2 Ευκλειδειος Ταξινομητης

Το περιεχόμενο αυτής της ενότητας αφορά τα Βήματα 10-13 της εργασίας.

Ένας από τους τρόπους με τον οποίο αξιοποιήθηκαν οι προαναφερθείσες μέσες τιμές ήταν η κατασκευή ενός Ευκλείδειου Ταξινομητή (ΕΤ). Η αρχή λειτουργίας του ΕΤ είναι η ακόλουθη: αρχικά, για κάθε κλάση y υπολογίζεται η μέση τιμή των χαρακτηριστικών της, $\vec{\mu}(y)$. Έπειτα, για κάθε δείγμα $\vec{x}$ υπολογίζεται η ευκλείδεια απόστασή του από τη μέση τιμή των χαρακτηριστικών κάθε κλάσης. Η ταξινόμηση του δείγματος γίνεται στην κλάση $\hat{y}$ η οποία απέχει την ελάχιστη απόσταση από το δείγμα. Με άλλα λόγια,

$$\hat{y} = \underset{y}{\operatorname{argmin}} \|\vec{x} - \vec{\mu}(y)\|_2. \tag{2.1}$$

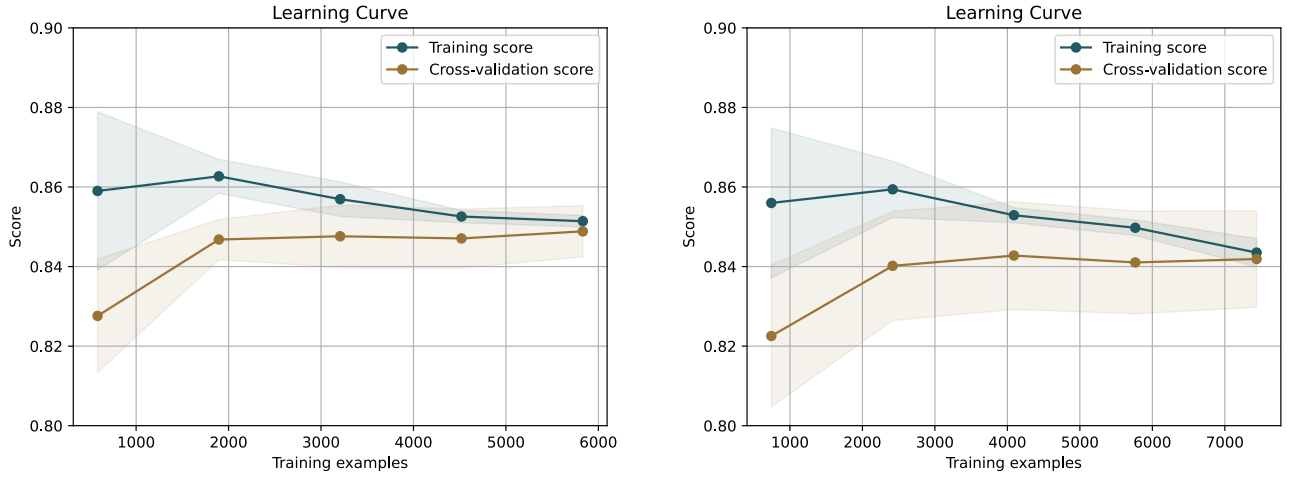
Η εκπαίδευση του ΕΤ πραγματοποιήθηκε στο σύνολο των δεδομένων εκπαίδευσης, αφότου πρώτα υλοποιήθηκε στα πρότυπα ενός ταξινομητή της βιβλιοθήκης scikit-learn. Δεδομένης της απλότητάς του, είναι αναμενόμενο η ακρίβεια του συγκεκριμένου ταξινομητή να μη συγκρίνεται με την ακρίβεια άλλων ταξινομητών. Για παράδειγμα, προσπαθώντας να ταξινομήσει το δείγμα στη θέση 101 των δεδομένων αξιολόγησης (Βήμα 10), επιλέγει εσφαλμένα την κλάση 0, τη στιγμή που η κλάση στην οποία ανήκει το δεδομένο είναι στην πραγματικότητα η 6. Η συνολική απόδοση του ταξινομητή αξιολογήθηκε βάσει της ακρίβειάς του (accuracy) στην πρόβλεψη των δεδομένων αξιολόγησης ίση με 81.415%.

Σε ό,τι αφορά την αξιολόγηση του ταξινομητή με τη χρήση 5-fold cross-validation, το validation score του υπολογίστηκε ίσο με 84.858%, χρησιμοποιώντας το σύνολο των δεδομένων εκπαίδευσης. Από τη στιγμή που τα δεδομένα αξιολόγησης ήταν διαθέσιμα, ο ΕΤ αξιολογήθηκε με χρήση 5-fold cross-validation και στην ένωση των δεδομένων εκπαίδευσης και αξιολόγησης, με validation score ίσο με 84.072%, δηλαδή ελαφρώς χαμηλότερο απ' ότι χρησιμοποιώντας μόνο τα δεδομένα εκπαίδευσης. Φυσικά, η διαδικασία αυτή έγινε καθαρά για διερευνητικούς σκοπούς, αφού σε ρεαλιστικές συνθήκες τα δεδομένα αξιολόγησης είτε δεν υπάρχουν, είτε δεν είναι διαθέσιμα. Τα αποτελέσματα αξιολόγησης του ΕΤ βρίσκονται συγκεντρωμένα στον Πίνακα 2.1.

	Accuracy	5-fold CV (train)	5-fold CV (full)
Ευκλείδειος Ταξινομητής	81.415%	84.858%	84.072%

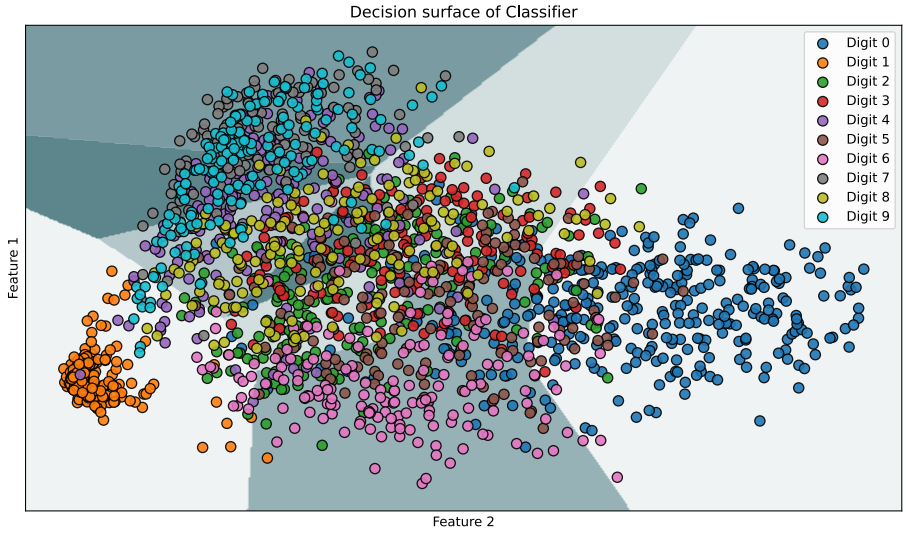
Πίνακας 2.1: Αξιολόγηση Ευκλείδειου Ταξινομητή.

Προκειμένου να υπάρχει μια εποπτεία του τρόπου εκμάθησης του ET, ανάλογα με τον όγκο των δεδομένων, για καθεμιά από τις δύο αυτές περιπτώσεις (μόνο δεδομένα εκπαίδευσης και ένωση δεδομένων εκπαίδευσης-αξιολόγησης), σχεδιάστηκε η καμπύλη εκμάθησης του ET (βλ. Εικόνα 2.1). Όπως φάνηκε και προηγουμένως, η σύγκλιση στη δεύτερη περίπτωση επιτυγχάνεται σε score το οποίο είναι ελαφρώς μικρότερο από αυτό της πρώτης περίπτωσης. Και στις δύο περιπτώσεις, η απόκλιση μεταξύ Training score και CV score είναι σχετικά υψηλή για μικρό όγκο δεδομένων.



Εικόνα 2.1: Καμπύλες εκμάθησης του Ευκλείδειου Ταξινομητή για (αριστερά) μόνο τα δεδομένα εκπαίδευσης και (δεξιά) για την ένωση των δεδομένων εκπαίδευσης-αξιολόγησης.

Το τελευταίο βήμα της ανάλυσης του ET αποτέλεσε ο προσδιορισμός των περιοχών απόφασής του. Φυσικά, η οπτικοποίηση των περιοχών απόφασης δε μπορεί να πραγματοποιηθεί για δεδομένα που περιγράφονται από 256 χαρακτηριστικά. Για το σκοπό αυτό, χρησιμοποιήθηκε η κλάση PCA (Principal Component Analysis) της scikit-learn, προκειμένου να πραγματοποιηθεί μια προβολή του συνόλου των χαρακτηριστικών σε ένα χώρο δύο διαστάσεων με το βέλτιστο τρόπο, ώστε η απεικόνιση των ίδιων καθώς και των περιοχών απόφασης να καθίσταται εφικτή. Έτσι, ο ET επανεκπαιδεύτηκε στο σύνολο των μετασχηματισμένων δεδομένων εκπαίδευσης και αξιολογήθηκε εκ νέου στα μετασχηματισμένα δεδομένα αξιολόγησης, επιτυγχάνοντας ακρίβεια 49.128%. Ο λόγος για τον οποίο ο επανεκπαιδευμένος ταξινομητής εμφάνισε τόσο μειωμένη απόδοση είναι ο μάλλον προφανής: παρότι μεταξύ κάθε pixel μιας εικόνας υπάρχει κάποιου είδους συσχέτιση, η ανάλυση της εικόνας είναι αδύνατο να αναχθεί στη μελέτη χαρακτηριστικών που ανήκουν σε ένα 2-διάστατο χώρο, τη στιγμή που το πλήρες πρόβλημα περιγράφεται από δεδομένα με χαρακτηριστικά που ανήκουν σε έναν 256-διάστατο χώρο. Το αποτέλεσμα φαίνεται στην Εικόνα 2.2, όπου οι περιοχές απόφασης διακρίνονται καθαρά.



Εικόνα 2.2: Τα δεδομένα και οι περιοχές απόφασης του Ευκλείδειου Ταξινομητή έπειτα από τη μείωση των διαστάσεων των χαρακτηριστικών σε 2.

3 Ταξινομητης Naive Bayes

Το περιεχόμενο αυτής της ενότητας αφορά τα Βήματα 14-16 της εργασίας.

Το επόμενο βήμα της υλοποίησης συστημάτων οπτικής αναγνώρισης ψηφίων αποτέλεσε η γενίκευση του ΕΤ, δηλαδή η κατασκευή ενός Ταξινομητή Naive Bayes (ΤΝΒ). Η αρχή λειτουργίας του ΤΝΒ βασίζεται επίσης στην παραδοχή πως τα χαρακτηριστικά των δειγμάτων (εδώ τα pixels των εικόνων) είναι μεταξύ τους ασυσχέτιστα και υποθέτει πως κάθε ένα εξ αυτών μπορεί να περιγραφεί από μία και μόνο κατανομή πιθανότητας. Η ταξινόμηση των δεδομένων βασίζεται στο ομώνυμο θεώρημα και συγκεκριμένα στη μεγιστοποίηση της a-posteriori πιθανότητας. Χρησιμοποιώντας τον προηγούμενο συμβολισμό, η κλάση ταξινόμησης προκύπτει από την απαίτηση

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, P\left(x_1|y\right) \cdot P\left(x_2|y\right) \cdot \ldots \cdot P\left(x_{256}|y\right) \cdot P(y), \tag{3.1}$$

όπου η $P(y)$ αντιστοιχεί στην a-priori πιθανότητα της κάθε κλάσης και η διάσπαση της $P\left(\vec{x}|y\right)$ είναι εφικτή χάρη στην υπόθεση των ασυσχέτιστων μεταξύ τους χαρακτηριστικών. Με την επιπλέον υπόθεση πως κάθε χαρακτηριστικό ακολουθεί την κανονική κατανομή και δεδομένου πως η λογαρίθμηση του δεξιού μέλους δε μεταβάλλει την ισότητα αφού ο φυσικός λογάριθμος είναι γνησίως αύξουσα συνάρτηση, η Σχέση (3.1) παίρνει τη μορφή

$$\hat{y} = \underset{y}{\operatorname{argmax}}\ \ln\left[P(y)\right] - \sum_{i=1}^{256} \ln\left[\sigma_i\left(y\right)\right] + \frac{1}{2}\left[\frac{x_i - \mu_i\left(y\right)}{\sigma_i\left(y\right)}\right]^2. \tag{3.2}$$

Γίνεται πλέον εμφανής ο λόγος για τον οποίο έγινε αναφορά σε γενίκευση του ΕΤ, αφού αυτός προκύπτει ως το όριο του ΤΝΒ κατά το οποίο οι a-priori πιθανότητες είναι ίσες για όλες τις κλάσεις και η διασπορά κοινή για κάθε χαρακτηριστικό κάθε κλάσης. Όπως και με τον ΕΤ, ο ταξινομητής ΤΝΒ υλοποιήθηκε στα πρότυπα των ταξινομητών της scikit-learn, αφότου πρώτα γράφτηκε μια απλή συνάρτηση υπολογισμού των a-priori πιθανοτήτων του δείγματος εκπαίδευσης (Βήμα 14 - βλ. Πίνακα 3.1).

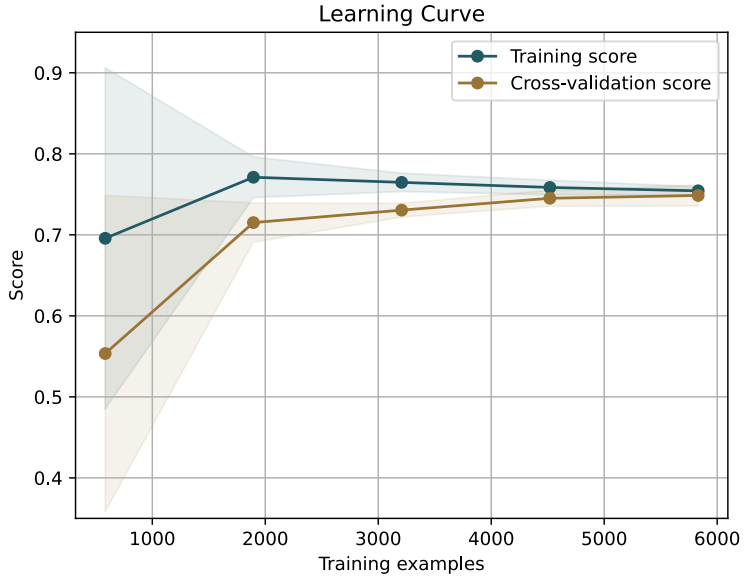
y	0	1	2	3	4	5	6	7	8	9
P(y)	16.38%	13.78%	10.03%	9.02%	8.94%	7.63%	9.11%	8.85%	7.43%	8.83%

Πίνακας 3.1: A-priori πιθανότητες για το δείγμα εκπαίδευσης.

Ένα ιδιαίτερο σημείο κατά την κατασκευή του TNB υπήρξε η περίπτωση των χαρακτηριστικών για τα οποία η διασπορά και κατ' επέκταση η τυπική απόκλιση είναι μηδενική. Στο όριο αυτό, ισχύει

$$\lim_{\sigma \to 0} \mathcal{N}(\mu, \sigma) = \delta(x - \mu), \tag{3.3}$$

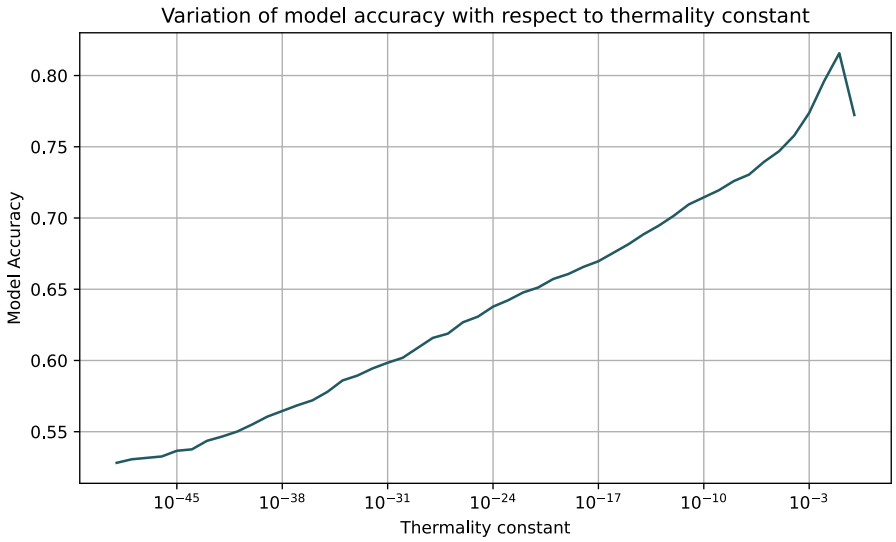
όπου $\delta(x)$ είναι η συνάρτηση (κατανομή) δ του Dirac. Ο τρόπος με τον οποίο αντιμετωπίστηκε το συγκεκριμένο πρόβλημα ήταν μέσω της εισαγωγής μιας σταθεράς `thermality`[1]. Αυτή εισάγεται στην κλήση του TNB και πολλαπλασιάζεται με τη μέγιστη τιμή της διασποράς που προκύπτει για όλα τα δεδομένα. Το γινόμενο αυτό προστίθεται στη συνέχεια σε όλες τις τιμές της διασποράς. Εάν η σταθερά `thermality` είναι επαρκώς μικρή, με τον τρόπο αυτό εξασφαλίζεται πως οι υπολογισμένες διασπορές δε θα επηρεαστούν σημαντικά, ενώ παράλληλα θα αρθούν όλα τα προβλήματα διαίρεσης με μηδενικές τιμές. Θέτοντας την παράμετρο αυτή ίση με 10^{-9}, ο TNB επιτυγχάνει σκορ 71.948% επί των δεδομένων αξιολόγησης. Ενδιαφέρον παρουσιάζει ίσως το γεγονός πως, ενώ το μοντέλο ταξινόμησης έγινε πολυπλοκότερο και λαμβάνει υπ' όψιν περισσότερες παραμέτρους, τελικά αξιολογείται χαμηλότερα σε σχέση με το πιο απλοϊκό μοντέλο. Η κύρια πηγή σφάλματος ενδέχεται να είναι οι πολύ μικρές διασπορές σε πολλά pixels. Στην Εικόνα 3.1 φαίνεται η καμπύλη εκμάθησης του TNB, στην οποία γίνεται εμφανές πως η τυπική απόκλιση επί του συνόλου των δειγμάτων που χρησιμοποιούνται κάθε φορά για εκπαίδευση και αξιολόγηση είναι πολύ υψηλή.



Εικόνα 3.1: Καμπύλη εκμάθησης του TNB για `thermality` $= 10^{-9}$.

[1] Η έμπνευση για την ονομασία «thermality» προέρχεται από τη Στατιστική Φυσική, όπου σε ορισμένες κατανομές δημιουργούνται αντίστοιχοι απειρισμοί σε μηδενική θερμοκρασία, οι οποίοι αίρονται με την αύξησή της σε μια μη μηδενική τιμή.

Είναι ίσως αναμενόμενο πως από τη στιγμή που η παράμετρος `thermality` υπεισέρχεται στο πρόβλημα μεταβάλλοντας τα δεδομένα της Σχέσης (3.2), θα έχει κάποιο ρόλο στην τελική αξιολόγηση του ταξινομητή. Στην Εικόνα 3.2 φαίνεται η ακρίβεια αξιολόγησης του ταξινομητή επί των δεδομένων αξιολόγησης, για διάφορες τιμές της σταθεράς `thermality`. Βλέπουμε πως η μέγιστη ακρίβεια επιτυγχάνεται για `thermality` $= 0.1$ (ίση με 81.565%), ενώ μειώνοντας περαιτέρω την τιμή της παραμέτρου η τελική ακρίβεια συνεχώς μειώνεται.



Εικόνα 3.2: Ακρίβεια του TNB ως συνάρτηση της σταθεράς `thermality`.

Πρέπει να τονιστεί εδώ πως η επιλογή της τιμής `thermality` $= 10^{-9}$ για την αρχική αξιολόγηση του TNB δεν ήταν τυχαία. Ο ταξινομητής `GaussianNB()` της scikit-learn επιτυγχάνει ακριβώς την ίδια ακρίβεια με τον παραπάνω TNB για τη συγκεκριμένη τιμή της παραμέτρου, επομένως για αυτό το λόγο επιλέχθηκε ως η default τιμή της.

Μια άλλη παράμετρος που υπεισέρχεται κατά την κλήση του TNB (Βήμα 16) είναι η λεγόμενη `use_unit_variance`, η οποία είναι τύπου Boolean και σε περίπτωση που η τιμή της είναι αληθής ο TNB καλείται με προκαθορισμένη την τιμή της διασποράς σε όλα τα χαρακτηριστικά και ίση με τη μονάδα. Στην περίπτωση αυτή, η μοναδική διαφορά του TNB από τον ET είναι οι μεταξύ τους άνισες a-priori πιθανότητες. Η ακρίβεια του TNB για ομοιόμορφη διασπορά ίση με τη μονάδα προκύπτει 81.266%, δηλαδή αρκετά κοντά στην ακρίβεια που επετεύχθη για τον ET. Συμπεραίνει κανείς, ξανά, πως η ακρίβεια του TNB εξαρτάται σε πολύ μεγάλο βαθμό από τις υπολογισμένες διασπορές στο δείγμα των δεδομένων, αφού, τόσο η παράμετρος `thermality`, όσο και η παράμετρος `use_unit_variance`, στην ουσία μεταβάλλουν τις διασπορές αυτές. Στον Πίνακα 3.2 φαίνονται συγκεντρωμένες οι τιμές της ακρίβειας που αναφέρθηκαν για τον TNB, καθώς και τα αποτελέσματα αξιολόγησής του μέσω της μεθόδου 5-fold cross-validation.

4 Συλλογες Ταξινομητων

Το περιεχόμενο αυτής της ενότητας αφορά τα Βήματα 17-18 της εργασίας.

Με δεδομένα τα αποτελέσματα για την επίδοση των ET και TNB, στη συνέχεια της μελέτης διερευνήθηκαν οι επιδόσεις άλλων ταξινομητών. Συγκεκριμένα, επιλέχθηκαν οι εξής έτοιμες υλοποιήσεις της βιβλιοθήκης scikit-learn: SVM (linear kernel), SVM (radial basis function

TNB	`thermality` $= 10^{-9}$	`thermality` $= 0.1$	`use_unit_variance=True`
Accuracy	71.948%	81.565%	81.266%
5-fold CV	74.832%	84.542%	84.831%

Πίνακας 3.2: Αξιολόγηση Ταξινομητή Naive Bayes.

kernel), SVM (sigmoid kernel), Decision Tree, k Nearest Neighbours (k = 1). Ειδικά για τον ταξινομητή kNN, η επιλογή k = 1 δεν έγινε τυχαία, αλλά κατόπιν διερεύνησης ως προς το ποιο k μεγιστοποιεί την επίδοσή του[2]. Τα αποτελέσματα για την επίδοση των ταξινομητών αυτών μέσω 5-fold CV συνοψίζονται στον Πίνακα 4.1.

Ταξινομητής	SVM (linear)	SVM (rbf)	SVM (sigmoid)	Decision Tree	kNN (k=1)
5-fold CV	95.254%	97.627%	89.288%	87.080%	96.736%

Πίνακας 4.1: Αξιολόγηση άλλων ταξινομητών.

Συνολικά, φαίνεται πως, με εξαίρεση τον ταξινομητή Decision Tree (και ίσως τον SVM με σιγμοειδή πυρήνα), η επίδοση του οποίου είναι συγκρίσιμη με αυτήν των ET και TNB, οι υπόλοιποι ταξινομητές που επιλέχθηκαν εμφανίζουν πολύ καλύτερα αποτελέσματα.

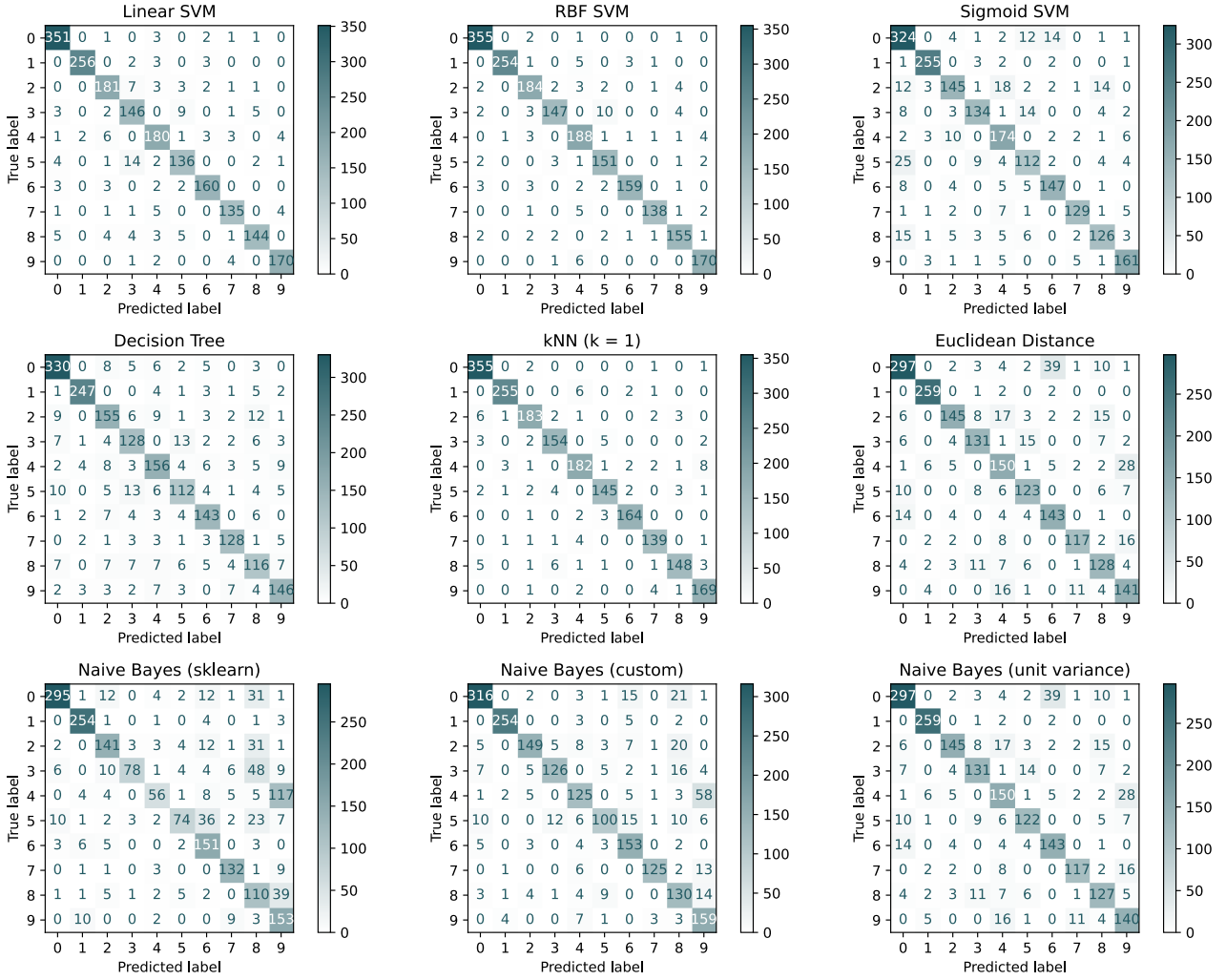
Έχοντας τα αποτελέσματα αυτά για την επίδοση συνολικά 9 ταξινομητών (των 5 του Πίνακα 4.1, των 3 του Πίνακα 3.2 και του ET), το επόμενο βήμα της μελέτης ήταν η δημιουργία συλλογών ταξινομητών, ούτως ώστε, μέσω συνδυασμών τους, να δημιουργηθούν μετα-ταξινομητές οι οποίοι να έχουν ακόμη πιο υψηλή απόδοση. Η αρχή λειτουργίας τέτοιων μετα-ταξινομητών είναι η ακόλουθη: οι ταξινομητές της συλλογής εκπαιδεύονται στο ίδιο σύνολο δεδομένων και μετά την εκπαίδευσή τους, για κάθε νέο δεδομένο, δίνουν ξεχωριστά τις εκτιμήσεις τους ως προς την κλάση ταξινόμησης. Η τελική ταξινόμηση του δεδομένου γίνεται είτε μέσω ψηφοφορίας (hard voting), όπου η κλάση που εκτίμησαν οι περισσότεροι ταξινομητές της συλλογής είναι και η κλάση ταξινόμησης[3], είτε μέσω μέσου όρου (soft voting), όπου για την επιλογή λαμβάνονται υπ' όψιν οι πιθανότητες με τις οποίες ταξινόμησαν οι επί μέρους ταξινομητές το δεδομένο[4]. Η επιλογή των ταξινομητών που συνιστούν τη συλλογή δεν έγινε τυχαία. Αρχικά, υπολογίστηκε ο Πίνακας Σύγχυσης (Confusion Matrix) για κάθε ταξινομητή, ώστε εκτός από την ακρίβεια του κάθε ταξινομητή να υπάρχει και η πληροφορία σχετικά με τα ψηφία τα οποία τείνει να μπερδεύει μεταξύ τους. Η λογική πίσω από αυτό ήταν να συνδυαστούν ταξινομητές οι

[2] Η διαδικασία αυτή καλείται hyperparameter tuning και η βιβλιοθήκη scikit-learn περιέχει την κλάση `GridSearchCV` η οποία επιτελεί το σκοπό αυτό. Στα πλαίσια της παρούσας εργασίας η εύρεση του βέλτιστου k για τον ταξινομητή kNN έγινε με μια απλή επαναληπτική διαδικασία. Ο λόγος για τον οποίο τα υπό συζήτηση μοντέλα δεν πέρασαν από hyperparameter tuning ήταν διότι η επίδοσή τους ήταν ούτως ή άλλως πολύ υψηλή σε σύγκριση με αυτή των ET και TNB.

[3] Για το λόγο αυτό, επιλέγεται μονός αριθμός ταξινομητών σε τέτοιου είδους συλλογές, ώστε να αποφεύγονται όσο το δυνατό περισσότερο οι ισοπαλίες.

[4] Κάτι τέτοιο προϋποθέτει οι ταξινομητές της συλλογής να πραγματοποιούν την ταξινόμηση βάσει της πιθανότητας που υπολογίζουν το δεδομένο να ανήκει στην κάθε κλάση.

οποίοι να εμφανίζουν διαφορές ως προς τα ζεύγη ψηφίων που τείνουν να ταξινομούν λανθασμένα, ώστε η συνολική τους απόδοση να βελτιωθεί (διαφορετικά, όλοι τους θα συμφωνούσαν στη λάθος απόφαση). Οι Πίνακες Σύγχυσης απεικονίζονται στην Εικόνα 4.1. Σημειώνεται πως ο Naive Bayes (Custom) είναι ο TNB με `thermality` $= 0.1$.



Εικόνα 4.1: Πίνακες Σύγχυσης για καθέναν από τους υπό μελέτη ταξινομητές.

Όπως εξάλλου υπέδειξαν και τα αποτελέσματα από την αξιολόγηση μέσω 5-fold CV, οι καλύτεροι ταξινομητές φάνηκε να είναι ο SVM (linear kernel), ο SVM (rbf kernel) και ο kNN με k = 1. Επιπλέον, το γεγονός πως τα κυριότερα λάθη ταξινόμησης καθενός εξ αυτών δεν παρατηρούνται με την ίδια συχνότητα στους υπόλοιπους (ο SVM (linear kernel) τείνει να μπερδεύει το 5 με το 3, ο SVM (rbf kernel) τείνει να μπερδεύει το 3 με το 5 και ο kNN τείνει να μπερδεύει το 4 με το 9) συνηγορεί υπέρ του συνδυασμού τους. Δημιουργώντας, λοιπόν, μια συλλογή από τους τρεις αυτούς ταξινομητές μέσω της κλάσης `VotingClassifier` της scikit-learn, το αποτέλεσμα που προέκυψε από 5-fold CV ήταν 97.764% στην περίπτωση του soft voting και 97.655% στην περίπτωση του hard voting. Τα αποτελέσματα αυτά είναι υψηλότερα από τα αποτελέσματα του κάθε ταξινομητή ξεχωριστά (βλ. Πίνακα 4.1), όμως ούτως ή άλλως το περιθώριο βελτίωσης ήταν σχετικά στενό. Προκειμένου να φανεί η επιτυχία της δημιουργίας μετα-ταξινομητών μέσω συλλογών, η διαδικασία επαναλήφθηκε χρησιμοποιώντας το Decision Tree, τον SVM (sigmoid kernel) και τον TNB. Στην περίπτωση αυτή, το 5-fold CV έδωσε ακρίβεια 90.975% για το soft voting και 90.605% για το hard voting, με τη βελτίωση που επέφερε η επιλογή συλλογής να

είναι αρκετά πιο εμφανής.

Μια άλλη περίπτωση στην οποία αξιοποιήθηκε η έννοια της συλλογής ήταν η λεγόμενη μέθοδος Bagging (Bootstrap aggregating), στην οποία οι ταξινομητές της συλλογής είναι ίδιου τύπου και καθένας εξ αυτών εκπαιδεύεται σε διαφορετικά υποσύνολα των δεδομένων εκπαίδευσης, με τη δυνατότητα να υπάρχουν και επικαλύψεις. Η μέθοδος αυτή εφαρμόστηκε για καθέναν εκ των 9 ταξινομητών μέσω της κλάσης `BaggingClassifier` της scikit-learn, χρησιμοποιώντας συλλογές των 10 ταξινομητών. Τα αποτελέσματα του 5-fold CV παρουσιάζονται στον Πίνακα 4.2, ο οποίος περιλαμβάνει και τα αποτελέσματα του Πίνακα 4.1, προκειμένου να φαίνεται η πιθανή βελτίωση που επέφερε η χρήση της μεθόδου.

Ταξινομητής	5-fold CV (No Bagging)	5-fold CV (Bagging)	Μεταβολή
SVM (linear)	95.254%	96.105%	+0.851%
SVM (rbf)	97.627%	97.600%	-0.027%
SVM (sigmoid)	89.288%	89.494%	+0.206%
Decision Tree	87.080%	91.990%	+4.910%
kNN (k = 1)	96.736%	96.585%	-0.151%
ET	84.858%	84.831%	-0.027%
TNB	74.832%	74.242%	-0.590%
TNB (Custom)	84.542%	84.762%	+0.220%
TNB (unit variance)	84.831%	84.844%	+0.013%

Πίνακας 4.2: Συγκεντρωτικά αποτελέσματα αξιολόγησης όλων των ταξινομητών.

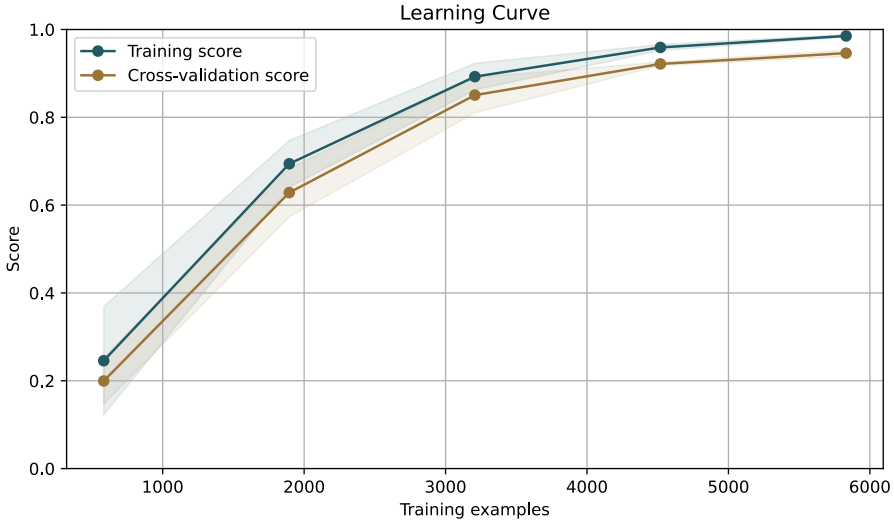
Βάσει των αποτελεσμάτων αυτών, γίνεται εμφανές πως σε ορισμένα μοντέλα η τεχνική αυτή δεν επέφερε ουσιαστική αλλαγή, με τη μεταβολή της επίδοσής τους να είναι σχεδόν μηδενική. Από την άλλη, φαίνεται το Bagging να αύξησε αισθητά την επίδοση του Decision Tree. Ο λόγος για τον οποίο συμβαίνει αυτό, είναι διότι η τεχνική Bagging βελτιώνει σημαντικά μοντέλα τα οποία χαρακτηρίζονται από υψηλή διακύμανση, χαρακτηριστικότερο παράδειγμα των οποίων αποτελεί ο ταξινομητής Decision Tree. Μάλιστα, ο ταξινομητής που προκύπτει εφαρμόζοντας την τεχνική Bagging σε Decision Trees αποτελεί μια ξεχωριστή κατηγορία ταξινομητή, γνωστή ως Random Forest. Χρησιμοποιώντας την υλοποίηση της scikit-learn για τον ταξινομητή Random Forest, η οποία προφανώς είναι βελτιστοποιημένη σε σχέση με την manual εφαρμογή του Bagging σε ένα Decision Tree, το αποτέλεσμα 5-fold CV ήταν 96.201%.

5 Ταξινομητης Νευρωνικου Δικτυου

Το περιεχόμενο αυτής της ενότητας αφορά το Βήμα 19 της εργασίας.

Το τελευταίο βήμα στη μελέτη και υλοποίηση συστημάτων οπτικής αναγνώρισης στοιχείων αποτέλεσε η κατασκευή ενός ταξινομητή νευρωνικού δικτύου, η οποία πραγματοποιήθηκε στα πρότυπα των ταξινομητών της βιβλιοθήκης scikit-learn. Αρχικά, ορίστηκε ένας DataLoader, ο οποίος αναλαμβάνει τη μετατροπή των δεδομένων (τα οποία έχουν αρχικά τη μορφή numpy arrays) σε torch tensors, καθώς και το διαμοιρασμό τους σε batches. Έπειτα, το νευρωνικό δίκτυο υλοποιήθηκε ως μια υποκλάση της `nn.Module`, από την οποία κληρονομεί τα διάφορα χαρακτηριστικά του. Οι παράμετροι που αρχικοποιούν το νευρωνικό δίκτυο κατά την αρχική του κλήση είναι ο αριθμός των hidden layers του, το πλήθος των χαρακτηριστικών των δεδομένων (στην παρούσα περίπτωση τα 256 pixels), το πλήθος των κλάσεων (στην παρούσα περίπτωση τα 10 ψηφία), το πλήθος εποχών εκπαίδευσης, το μέγεθος των batches, καθώς και ο ρυθμός εκμάθησης. Κατά την εκπαίδευση του νευρωνικού δικτύου, εκτός από τα δεδομένα εκμάθησης, υπάρχει η δυνατότητα ορισμού μιας παραμέτρου `split`. Εάν η παράμετρος αυτή είναι μη μηδενική, τότε τα δεδομένα εκπαίδευσης χωρίζονται σε δύο υποσύνολα, εκ των οποίων το ένα χρησιμοποιείται για την εκπαίδευση του δικτύου και το άλλο για την αξιολόγησή του (βάσει της ακρίβειάς του στην ταξινόμηση των δεδομένων αξιολόγησης που προκύπτουν λόγω της `split`).

Αρχικά, εκπαιδεύτηκε ένα νευρωνικό δίκτυο με συνάρτηση ενεργοποίησης την σιγμοειδή, το οποίο είχε 2 σύνολα από hidden layers, το καθένα με 100 layers. Η εκπαίδευσή του διήρκησε για 300 εποχές, με batches των 32 στοιχείων και ρυθμό εκμάθησης ίσο με 0.01. Θέτοντας την παράμετρο `split` ίση με 0.2, η ακρίβεια του δικτύου υπολογίστηκε βάσει των δεδομένων εκπαίδευσης ίση με 91.363%, ενώ με χρήση 5-fold CV η απόδοσή του υπολογίστηκε ίση με 91.222%. Τέλος, το δίκτυο αυτό αξιολογήθηκε στα δεδομένα αξιολόγησης, πετυχαίνοντας ακρίβεια ίση με 86.996%. Στην Εικόνα 5.1 φαίνεται η καμπύλη εκμάθησης του συγκεκριμένου νευρωνικού δικτύου.



Εικόνα 5.1: Καμπύλη εκμάθησης Νευρωνικού Δικτύου.

Η μελέτη ολοκληρώθηκε με την εκπαίδευση ενός άλλου δικτύου, για το οποίο επιλέχθηκαν οι ίδιες παράμετροι, με δύο βασικές διαφορές. Η πρώτη ήταν πως ο αριθμός εποχών αυξήθηκε στις 500 και η δεύτερη ήταν πως η παράμετρος `split` τέθηκε ίση με 0, ώστε το νευρωνικό δίκτυο να εκπαιδευτεί στο σύνολο των δεδομένων εκπαίδευσης. Η ακρίβεια του μοντέλου αυτού στην ταξινόμηση των δεδομένων αξιολόγησης υπολογίστηκε ίση με 91.928%, δηλαδή ελαφρώς υψηλότερη σε σχέση με την αρχική υλοποίηση.