

Μέρος Β΄

Σε αυτό το μέρος θα συγκρίνουμε την απόδοση των δικών μας υλοποιήσεων των **Naïve Bayes Bernoulli**, **Random Forest** και **AdaBoost** με τις έτοιμες υλοποιήσεις των αντίστοιχων αλγορίθμων στη βιβλιοθήκη **Scikit-learn**. Η σύγκριση γίνεται βάσει:

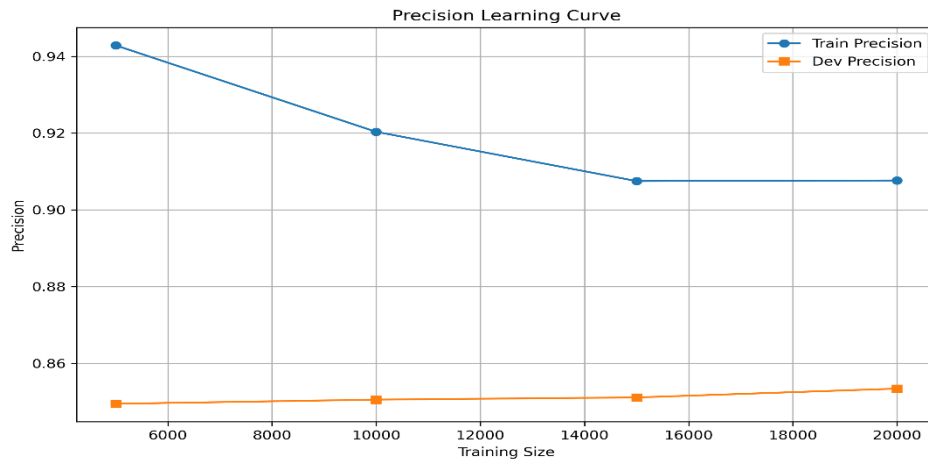
- **Καμπυλών μάθησης** (Precision, Recall, F1-score) για διαφορετικά μεγέθη δεδομένων εκπαίδευσης.
- **Τελικών επιδόσεων στο test set** (μέσω πινάκων αξιολόγησης).

Για να είναι η σύγκριση δίκαιη:

Χρησιμοποιούμε **το ίδιο λεξιλόγιο και την ίδια προεπεξεργασία** δεδομένων (IMDBPreprocessor.py). Επίσης, το μέγεθος του τελικού λεξιλογίου (καλύτερες m λέξεις βάσει του κέρδους πληροφορίας) καθώς και οι τιμές των n πιο συχνών και των k πιο σπάνιων λέξεων θα είναι ίδιες μεταξύ των δικών μας υλοποιήσεων και των υλοποιήσεων από το Scikit-learn. Επίσης οι τιμές των υπερπαραμέτρων των **AdaBoost** και **Random Forest** θα είναι οι ίδιες. Τέλος, η δομή των Main αρχείων μεταξύ των δύο υλοποιήσεων και οι βοηθητικές κλάσεις μετρικών, αποθήκευσης και οπτικοποίησης αποτελεσμάτων είναι τα ίδια ακριβώς

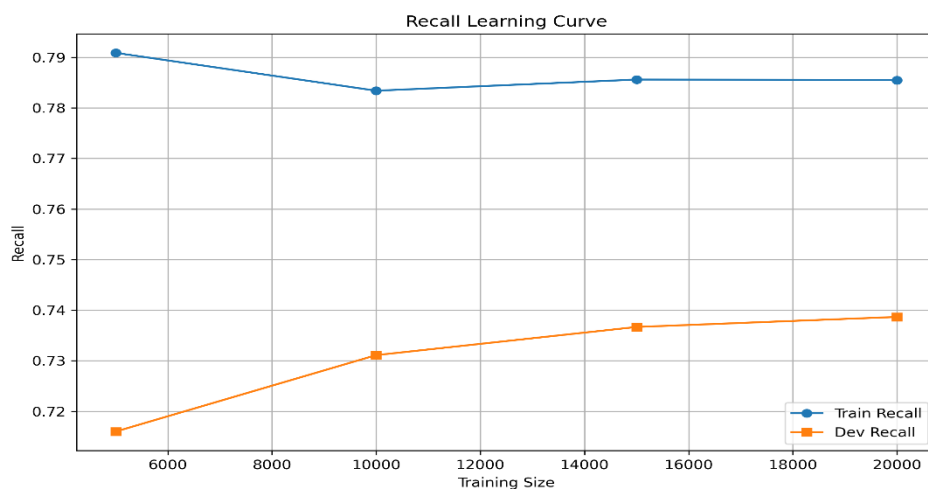
- Αποτελέσματα NaiveBayesBernoulli Sklearn:

1. Καμπύλες Μάθησης συναρτήσει πλήθους παραδειγμάτων εκπαίδευσης (Θετικές κριτικές)

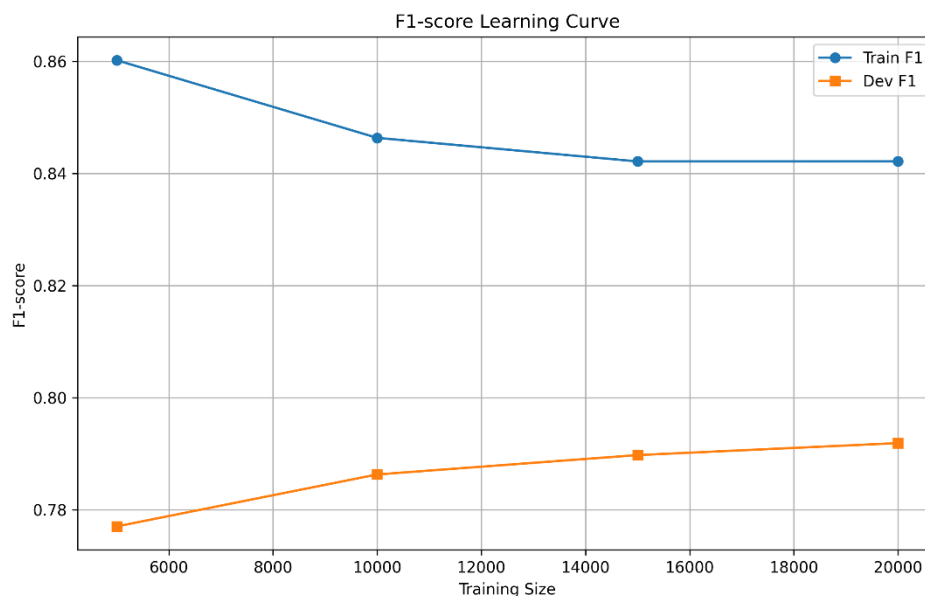


Στην δική μας υλοποίηση, το train precision ξεκινά από 0.92 και σταδιακά μειώνεται κοντά στο 0.90, ενώ το dev precision παραμένει σχετικά σταθερό γύρω στο 0.86. Στην **Scikit-learn** υλοποίηση, το train precision ξεκινά από 0.94, αλλά μειώνεται πιο απότομα, σταθεροποιούμενο γύρω στο 0.90. Το dev precision είναι ελαφρώς χαμηλότερο (0.86), αλλά ακολουθεί παρόμοια τάση.

Συμπερασματικά: Η Scikit-learn υλοποίηση ξεκινά με υψηλότερο precision, αλλά καθώς τα δεδομένα εκπαίδευσης αυξάνονται, η διαφορά μειώνεται και τελικά συγκλίνουν. Αυτό δείχνει ότι το δικό μας μοντέλο είναι πιο συντηρητικό στις προβλέψεις του.



Στην **δική μας** υλοποίηση, το train recall κυμαίνεται μεταξύ 0.77 και 0.78, ενώ το dev recall αυξάνεται σταδιακά από 0.72 σε 0.74. Στην **Scikit-learn** υλοποίηση, το train recall ξεκινά ελαφρώς υψηλότερα (0.79), αλλά μειώνεται πιο έντονα. Το dev recall αυξάνεται με πιο σταθερό ρυθμό και τελικά φτάνει πιο ψηλά (0.74-0.75). **Συμπερασματικά**, η Scikit-learn υλοποίηση έχει υψηλότερο recall στο training set, αλλά μειώνεται πιο απότομα. Το τελικό dev recall είναι πολύ κοντά και στις δύο υλοποιήσεις.



Στην **δική μας** υλοποίηση, το train F1-score ξεκινά από 0.85 και μειώνεται σταδιακά, ενώ το dev F1-score αυξάνεται σταθερά, πλησιάζοντας το 0.80. Στην **Scikit-learn** υλοποίηση, το train F1-score ξεκινά υψηλότερα (0.86), αλλά πέφτει πιο γρήγορα. Το dev F1-score αυξάνεται με πιο σταθερό ρυθμό και τελικά φτάνει πολύ κοντά σε αυτό της δικής μας. **Συμπερασματικά**, η Scikit-learn υλοποίηση μαθαίνει πιο γρήγορα, αλλά παρουσιάζει πιο απότομη μείωση του train F1-score. Η δική μας υλοποίηση έχει πιο σταθερή συμπεριφορά και λιγότερη απότομη μείωση, γεγονός που δείχνει ίσως καλύτερη γενίκευση.

2. Πίνακας γενικής αξιολόγησης:

Μετρική	Custom Naïve Bayes	Scikit-learn Naïve Bayes
Test Class 0 Precision	0.7676	0.7195
Test Class 0 Recall	0.8718	0.8714
Test Class 0 F1-score	0.8164	0.7882
Test Class 1 Precision	0.8517	0.8370
Test Class 1 Recall	0.7360	0.6604
Test Class 1 F1-score	0.7896	0.7380
Test Macro Precision	0.8096	0.7783
Test Macro Recall	0.8039	0.7659
Test Macro F1-score	0.8030	0.7632
Test Micro Precision / Recall / F1-score	0.8039	0.7659

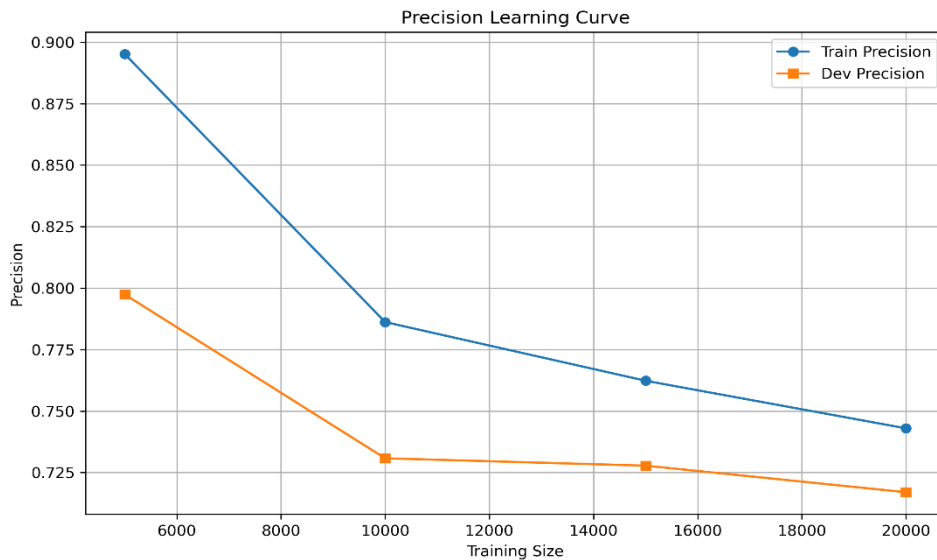
Η δική μας υλοποίηση έχει υψηλότερο precision και recall για την κλάση 1 (θετικές κριτικές), με διαφορά 7.56% στο recall. Στην κλάση 0 (αρνητικές κριτικές), οι επιδόσεις είναι παρόμοιες, αλλά το precision είναι υψηλότερο στη δική μας υλοποίηση. Οι **macro μετρικές** (macro precision, macro recall, macro F1) είναι **αρκετά υψηλότερες στην δική μας υλοποίηση**. Στις **micro μετρικές**, η συνολική ακρίβεια (micro precision, recall, F1-score) είναι 80.39% στην δική μας και 76.59% στην Scikit-learn.

Γενικά Συμπεράσματα:

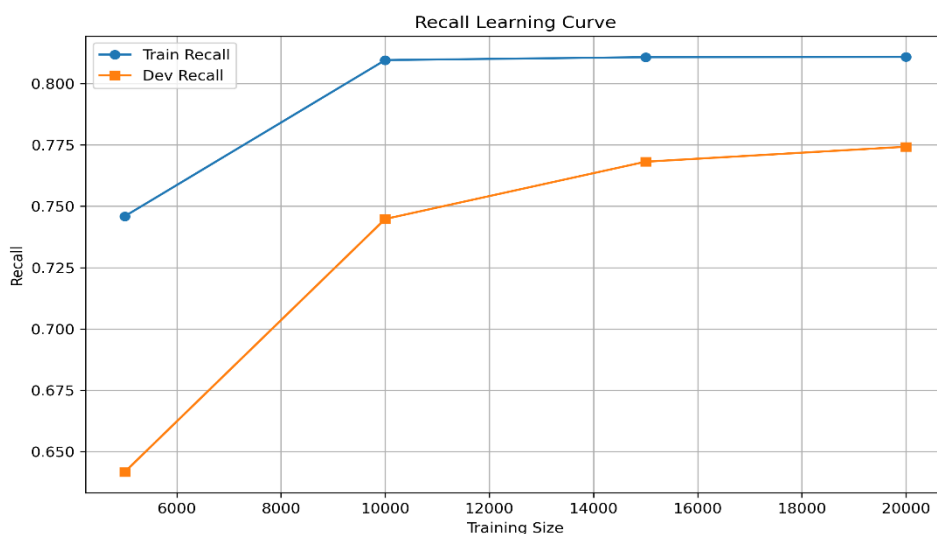
Η δική μας υλοποίηση του Naïve Bayes έχει ανώτερη απόδοση σε όλες τις μετρικές, ειδικά στο recall της κλάσης 1, γεγονός που δείχνει ότι μπορεί να αναγνωρίζει περισσότερες θετικές κριτικές. Η **Scikit-learn** υλοποίηση ήταν πολύ πιο γρήγορη, αλλά παρουσιάζει χαμηλότερη γενική απόδοση, ιδιαίτερα στο Recall.

- Αποτελέσματα Random Forest Sklearn:

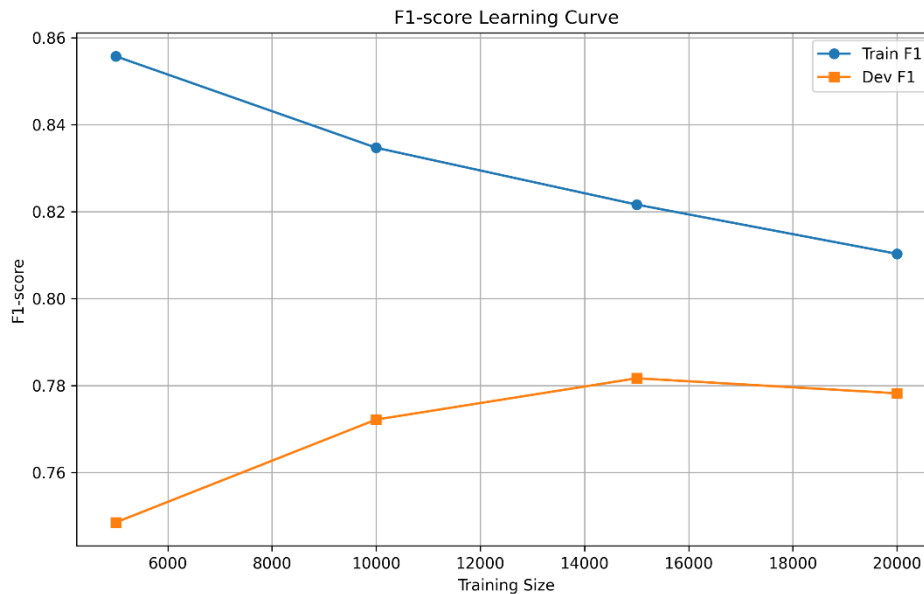
1. Καμπύλες Μάθησης συναρτήσεσι πλήθους παραδειγμάτων εκπαίδευσης (Θετικές κριτικές)



Η υλοποίηση του **Scikit-learn** επιτυγχάνει υψηλότερο Precision στο σύνολο εκπαίδευσης, υπάρχει αρκετή μείωση στο σύνολο ανάπτυξης, γεγονός που δείχνει ότι ενδέχεται να υπάρχει υπερπροσαρμογή (overfitting). Η δική μας υλοποίηση έχει αντίστοιχη μείωση του Precision καθώς αυξάνεται το μέγεθος των δεδομένων εκπαίδευσης.



Παρατηρούμε ότι στη **δική μας** υλοποίηση το **Recall** αυξάνεται σταδιακά τόσο στο train όσο και στο dev set, δείχνοντας ότι η ταξινόμηση καταφέρνει να ανιχνεύσει περισσότερες θετικές κριτικές με κάποια μείωση της ακρίβειας. Στην υλοποίηση του **Scikit-learn**, το recall αυξάνεται πιο απότομα, αλλά παρουσιάζει μεγαλύτερη διακύμανση στις υψηλότερες τιμές εκπαίδευσης.



Παρατηρούμε ότι στη **δική μας** υλοποίηση το F1-score είναι αρκετά σταθερό και έχει μικρότερο χάσμα μεταξύ εκπαίδευσης και ανάπτυξης, γεγονός που δείχνει καλύτερη γενίκευση. Στην υλοποίηση του **Scikit-learn**, το F1-score του συνόλου εκπαίδευσης είναι υψηλότερο από το αντίστοιχο του συνόλου ανάπτυξης.

2. Πίνακας γενικής αξιολόγησης:

Μετρική	Custom Random Forest	Scikit-learn Random Forest
Test Class 0 Precision	0.7354	0.8081
Test Class 0 Recall	0.5939	0.6527
Test Class 0 F1-score	0.6571	0.7221

Μετρική	Custom Random Forest	Scikit-learn Random Forest
Test Class 1 Precision	0.6449	0.7087
Test Class 1 Recall	0.7689	0.8450
Test Class 1 F1-score	0.7013	0.7709
Test Macro Precision	0.6901	0.7584
Test Macro Recall	0.6815	0.7488
Test Macro F1-score	0.6793	0.7465
Test Micro Precision / Recall / F1-score	0.6815	0.7488

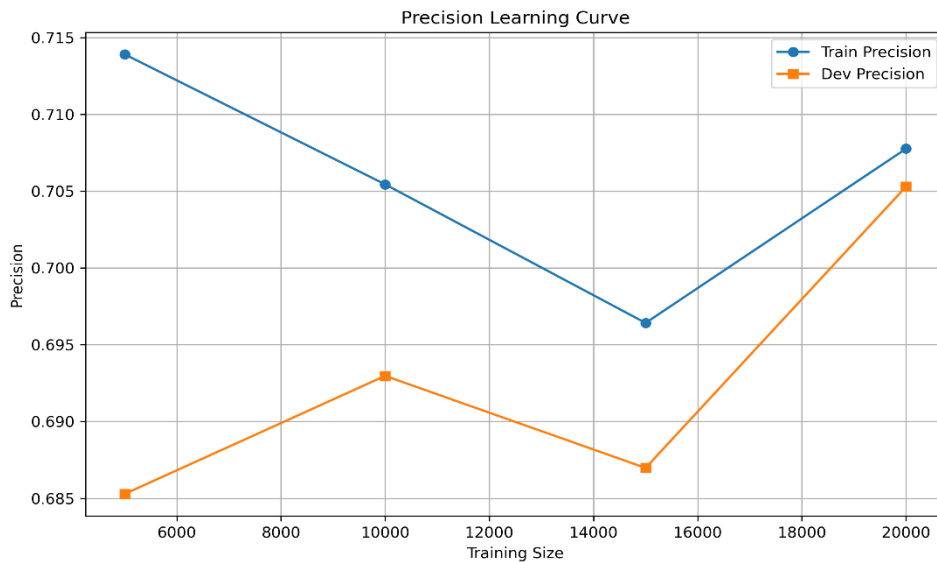
Η υλοποίηση του **Scikit-learn** έχει καλύτερο **Precision** σε όλες τις κατηγορίες, πράγμα που σημαίνει ότι όταν κάνει μια θετική ή αρνητική πρόβλεψη, έχει υψηλότερη ακρίβεια. Επίσης, το **Scikit-learn** έχει καλύτερο **Recall** και στις δύο κατηγορίες (0 και 1), υποδεικνύοντας ότι εντοπίζει περισσότερα σωστά παραδείγματα κάθε κλάσης. Όλες οι τιμές F1-score είναι υψηλότερες στην υλοποίηση του Scikit-learn, γεγονός που δείχνει ότι συνολικά έχει καλύτερη ισορροπία μεταξύ Precision και Recall. Οι μακρο-μετρικές δείχνουν ότι η υλοποίηση του Scikit-learn αποδίδει καλύτερα συνολικά.

Γενικά Συμπεράσματα:

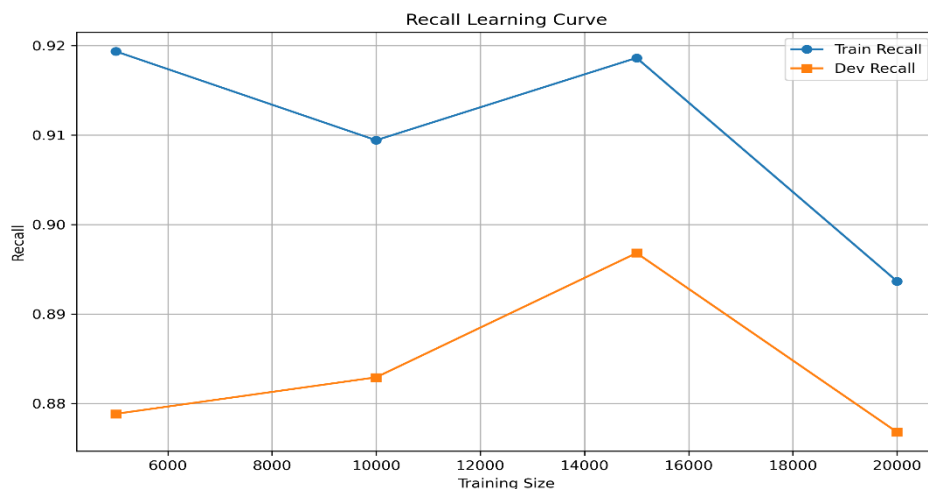
Η υλοποίηση του Scikit-learn εμφανίζει καλύτερη απόδοση σε όλους τους δείκτες, αλλά είναι πιθανό να έχει μεγαλύτερη υπερπροσαρμογή στο training set. Από την άλλη, η δική μας υλοποίηση μπορεί να είναι πιο σταθερή, αν και έχει χαμηλότερη ακρίβεια.

- Αποτελέσματα AdaBoost Sklearn:

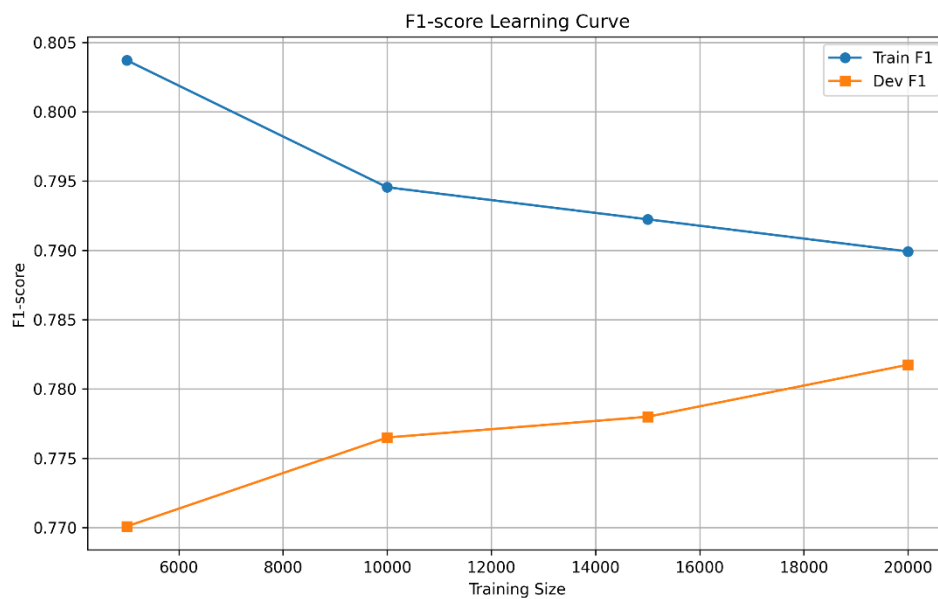
1. Καμπύλες Μάθησης συναρτήσεως πλήθους παραδειγμάτων εκπαίδευσης (Θετικές κριτικές)



Στην υλοποίηση του **Scikit-learn** η καμπύλη είναι πιο σταθερή και δείχνει ότι η απόδοση βελτιώνεται σταδιακά, κάτι που σημαίνει ότι το μοντέλο μαθαίνει καλύτερα καθώς αυξάνονται τα δεδομένα. Στην **δική μας** υλοποίηση παρατηρείται σημαντική μείωση του precision στο training set καθώς αυξάνονται τα δεδομένα, κάτι που υποδηλώνει ότι το μοντέλο γίνεται πιο γενικό και λιγότερο προσαρμοσμένο στα δεδομένα εκπαίδευσης.



Στην **δική μας** υλοποίηση, το recall στο training set παραμένει σχετικά υψηλό, αλλά δεν παρατηρείται μεγάλη αύξηση στο dev set, κάτι που υποδηλώνει ότι το μοντέλο δεν εκμεταλλεύεται πλήρως την αύξηση των δεδομένων. Στην υλοποίηση του **Scikit-learn**, το recall τόσο στο training όσο και στο dev set είναι σταθερά υψηλότερο από την **δική μας** υλοποίηση. Η βελτίωση στο dev set δείχνει ότι το μοντέλο της **Scikit-learn** είναι ικανότερο να εντοπίζει θετικά δείγματα.



Στην **δική μας** υλοποίηση, το **F1-score στο training set** ξεκινά από υψηλές τιμές, αλλά μειώνεται σταδιακά καθώς αυξάνεται το μέγεθος των δεδομένων. Η απόδοση στο dev set βελτιώνεται, αλλά με αργό ρυθμό, δείχνοντας ότι το μοντέλο δεν επωφελείται πλήρως από την αύξηση των δεδομένων. Στην υλοποίηση του **Scikit-learn**, το **F1-score του training set** είναι υψηλότερο από την **δική μας** υλοποίηση, αλλά δεν πέφτει τόσο απότομα. Το F1-score στο dev set αυξάνεται προοδευτικά, δείχνοντας καλύτερη ικανότητα γενίκευσης.

2. Πίνακας γενικής αξιολόγησης:

Μετρική	Custom AdaBoost	Sklearn AdaBoost
Test Class 0 Precision	0.7287	0.8187
Test Class 0 Recall	0.5418	0.6087
Test Class 0 F1	0.6215	0.6983
Test Class 1 Precision	0.6183	0.6947
Test Class 1 Recall	0.7768	0.8729
Test Class 1 F1	0.6886	0.7737
Test Macro Precision	0.6735	0.7561
Test Macro Recall	0.6593	0.7408
Test Macro F1	0.6550	0.7359
Test Micro Precision / Recall / F1-score	0.6593	0.7408

Η υλοποίηση του Scikit-learn έχει καλύτερο **Precision** και **Recall** σε όλες τις κατηγορίες, γεγονός που σημαίνει ότι αναγνωρίζει περισσότερα σωστά παραδείγματα και κάνει λιγότερα λάθη σε θετικές και αρνητικές προβλέψεις. Το **F1-score** είναι υψηλότερο, δείχνοντας καλύτερη ισορροπία μεταξύ ακρίβειας και ανάκλησης. Οι **macro** και **micro μετρικές** επιβεβαιώνουν ότι η sklearn υλοποίηση αποδίδει καλύτερα συνολικά, εξασφαλίζοντας πιο αξιόπιστα αποτελέσματα.

Γενικά Συμπεράσματα:

Η υλοποίηση του Scikit-learn προσφέρει **υψηλότερη ακρίβεια** σε όλες τις μετρικές, αλλά μπορεί να εμφανίζει **μεγαλύτερη υπερπροσαρμογή** στο training set. Αντίθετα, η δική μας υλοποίηση, αν και έχει χαμηλότερη απόδοση, μπορεί να είναι **πιο σταθερή** και να γενικεύει καλύτερα σε άγνωστα δεδομένα.

