

1η Άσκηση: Λογιστική παλινδρόμηση

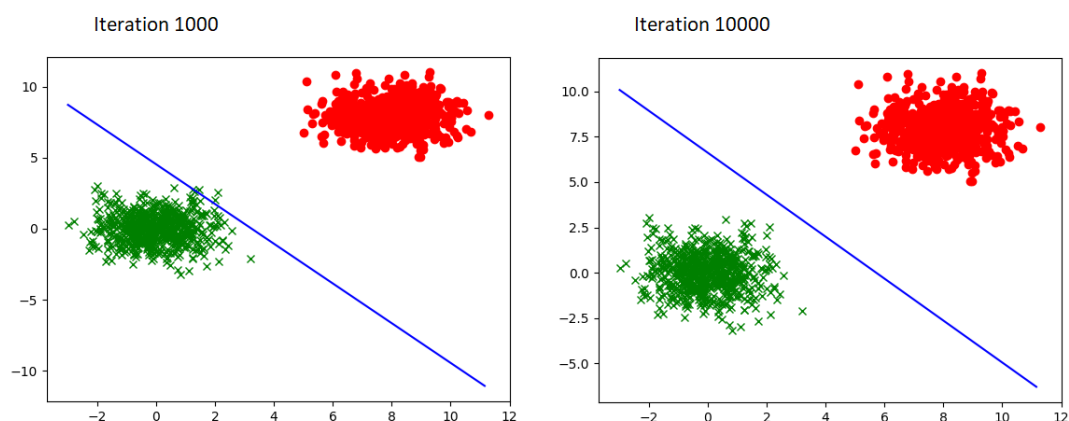
Βασίλειος Κοκκινογένης it2021042

Πείραμα 1: Συνθετικά δεδομένα

Ερώτηση 1: Τι παρατηρείτε για τη γραμμή και για την τιμή της συνάρτησης απώλειας, καθώς προχωράει η εκπαίδευση του μοντέλου; Υπολογίστε και την ευστοχία του μοντέλου μετά την εκπαίδευση.

Απάντηση: Παρατηρώ ότι η γραμμή αρχικά (1000 επαναλήψεις) ακουμπάει ελαφρώς τα πράσινα σημεία και στη συνέχεια, καθώς προχωράει η εκπαίδευση του μοντέλου, μετατοπίζεται προς τα δεξιά, μέχρι να μην ακουμπάει κανένα σημείο (10000 επαναλήψεις), με αποτέλεσμα να διαχωρίζει πιο σωστά τα πράσινα από τα κόκκινα σημεία σε σχέση με την αρχική κατάσταση. Η βελτιστοποίηση αυτή φαίνεται και στην τιμή της συνάρτησης απώλειας, όπου αρχικά είχαμε $\text{loss} = 0.1150$ ενώ στο τέλος έχουμε $\text{loss} = 0.0145$. Συνεπώς το μοντέλο κάνει πιο σίγουρες προβλέψεις μετά την εκπαίδευση, με ποσοστό ευστοχίας = 1.0 (100%).

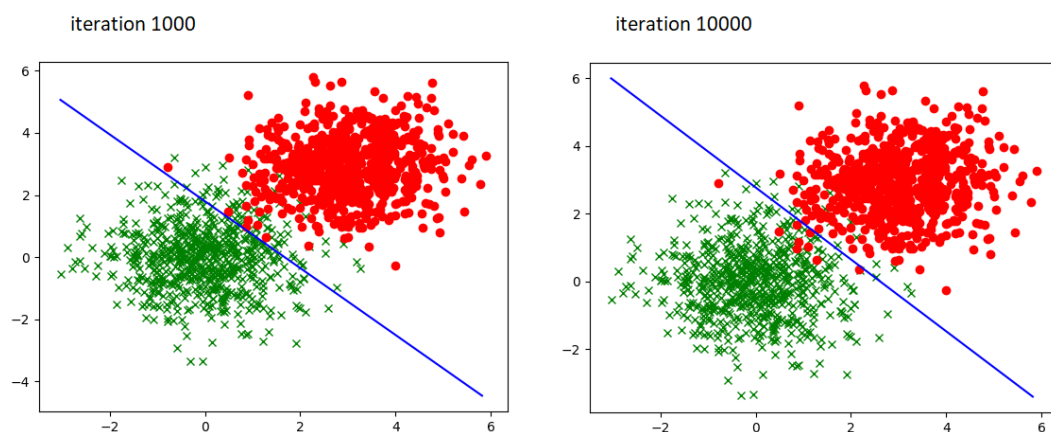
Centers in (0,0) and (8,8)



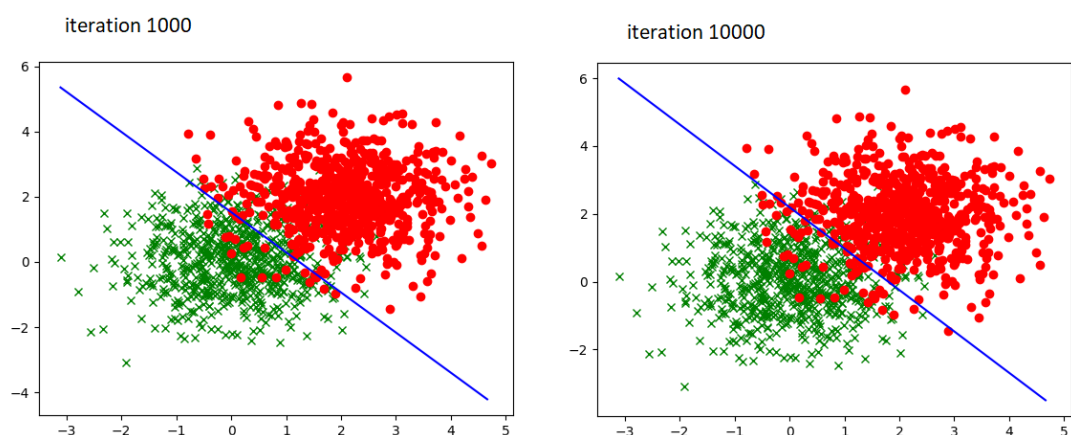
Ερώτηση 2: Δοκιμάστε και γράψτε τις παρατηρήσεις σας και για άλλα, διαφορετικά κέντρα των κλάσεων, που αντιστοιχούν σε πιο δύσκολα διαχωρίσιμα προβλήματα.

Απάντηση: Δοκιμάζοντας για διαφορετικά κέντρα των κλάσεων με κέντρα στο (0, 0) και (3, 3), ή ακόμη μικρότερα (2, 2) ή (1, 1) αντί για (8, 8), παρατηρούμε ότι είναι δύσκολο να διαχωριστούν πλήρως τα σημεία άσχετα με τον αριθμό των επαναλήψεων (γιατί πολλά σημεία της μιας κλάσης πέφτουν δίπλα σε άλλα σημεία της άλλης κλάσης). Ωστόσο πάλι παρατηρούμε μια βελτιστοποίηση μετά την εκπαίδευση του μοντέλου (όσο πιο δύσκολα γραμμικά διαχωρίσιμο είναι ένα πρόβλημα, τόσο μικρότερη βελτιστοποίηση έχουμε). Για κέντρα (3,3) παρατηρούμε ότι η γραμμή αρχικά (1000 επαναλήψεις) τείνει προς τη μεριά των πράσινων σημείων και στη συνέχεια, καθώς προχωράει η εκπαίδευση του μοντέλου, μετατοπίζεται προς τα δεξιά, μέχρι να έχει περίπου την ίδια απόσταση και από τις 2 κλάσεις (10000 επαναλήψεις), με αποτέλεσμα να διαχωρίζει πιο σωστά τα πράσινα από τα κόκκινα σημεία σε σχέση με την αρχική κατάσταση. Η βελτιστοποίηση φαίνεται και στην τιμή της συνάρτησης απώλειας, όπου αρχικά είχαμε $\text{loss} = 0.2003$ ενώ στο τέλος έχουμε $\text{loss} = 0.0697$. Συνεπώς το μοντέλο κάνει πιο σίγουρες προβλέψεις μετά την εκπαίδευση, με ποσοστό ευστοχίας = 0.9866666666666667 (98.6%). Παρόμοια αποτελέσματα βλέπουμε και για τα κέντρα (2,2) και (1,1) για την 2η κλάση, όμως με μεγαλύτερο loss και μικρότερη ευστοχία (και μικρότερη βελτιστοποίηση).

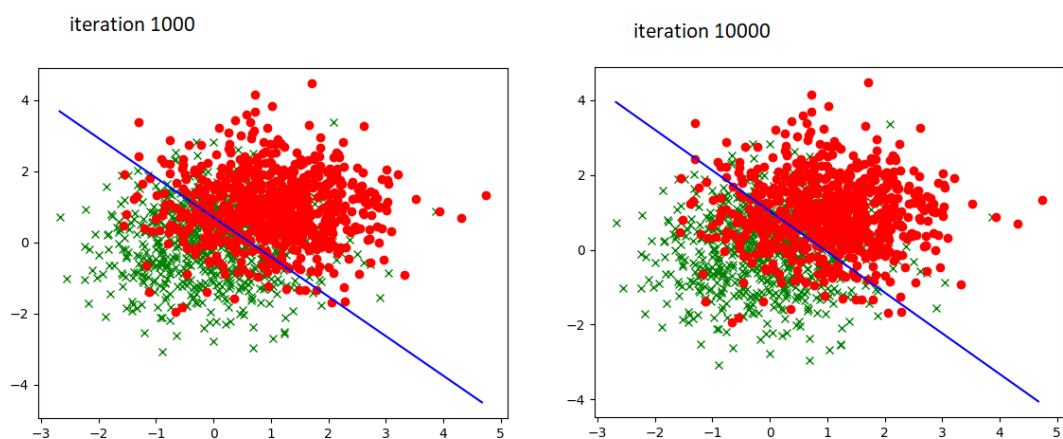
Centers in (0,0) and (3,3)



Centers in (0,0) and (2,2)



Centers in (0,0) and (1,1)



Πείραμα 2: Breast cancer dataset

Ερώτηση 3: Υπολογίστε και σημειώστε το μέσο όρο και την τυπική απόκλιση της ευστοχίας του μοντέλου για τις 20 επαναλήψεις. Πόσο χρόνο κάνει να τρέξει αυτά τα πειράματα στον υπολογιστή σας; Αναφέρετε τον επεξεργαστή, τη μνήμη του υπολογιστή σας, καθώς και το χρόνο εκτέλεσης σε δευτερόλεπτα.

Απάντηση: Για τις 20 επαναλήψεις έχουμε:

```
Mean Accuracy: 0.9084795321637428
Standard Deviation of Accuracy: 0.022894942229068356
Execution Time: 6.50 seconds
Processor: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz
Total RAM: 15.79 GB

Process finished with exit code 0
```

Πείραμα 3: Breast cancer dataset (scikit-learn)

Ερώτηση 4: Κάντε ακριβώς ό,τι κάνατε στο προηγούμενο πείραμα, αλλά χρησιμοποιώντας την κλάση LogisticRegression του scikit-learn, με τις προεπιλεγμένες παραμέτρους. Αναφέρετε και πάλι το χρόνο εκτέλεσης. Πώς συγκρίνονται τα δύο μοντέλα, σε ό,τι αφορά την επίδοση και την αποτελεσματικότητά τους;

Απάντηση: Για τις 20 επαναλήψεις έχουμε:

```
Mean Accuracy: 0.9581871345029238
Standard Deviation of Accuracy: 0.017276223680377144
Execution Time: 0.14 seconds
Processor: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz
Total RAM: 15.79 GB

Process finished with exit code 0
```

Συμπεραίνουμε λοιπόν, ότι το μοντέλο του scikit-learn έχει καλύτερη επίδοση και αποτελεσματικότητα, εφόσον όχι μόνο έχει μεγαλύτερο mean accuracy ($0.9581 > 0.9084$) και μικρότερο standard deviation ($0.017 < 0.022$) αλλά και πολύ μικρότερο χρόνο εκτέλεσης ($0.14s < 6.5s$). Επιπροσθέτως η διαφορά στον χρόνο εκτέλεσης θα μεγάλωνε ακόμα περισσότερο αν στο πείραμα 2 χρησιμοποιούσαμε το δικό μας μοντέλο με batch size = None, όπου σε αυτήν την περίπτωση θα είχαμε σχεδόν πενταπλάσιο χρόνο εκτέλεσης (30.45s) από το μοντέλο με batch size = 64 (6.5s).