# Εργασία Μηχανικής Μάθησης 2022

Tsirmpas Dimitris

16th January 2023

## 1 Λογιστική Παλινδρόμηση

### 1.1 Ερώτημα Δ

Ο ταξινομητής μας έχει υλοποιηθεί στο αρχείο `LogisticRegClassifier.py`. Η πλήρης τεκμηρίωση του μοντέλου, των μεθόδων του και των υπερπαραμέτρων βρίσκεται εκεί με τη μορφή docstrings. Η κανονικοποίηση $L_2$ έχει ήδη υλοποιηθεί σε αυτό το αρχείο, αλλά για τους σκοπούς αυτής της ερώτησης θα θέσουμε την υπερπαράμετρο $\lambda$ ως 0, παρακάμπτοντας την.

Ο κώδικας για την εκτέλεση του μοντέλου βρίσκεται στο αρχείο `run_logistic.py`. Θα τρέξουμε το μοντέλο με υπερπαραμέτρους `iter = 500` και `alpha = 0.2`. Το αποτέλεσμα είναι η ακρίβεια εκπαίδευσης να είναι ίση με $0.982$ και η ακρίβεια ελέγχου ίση με $0.981$. Τα πλήρη αποτελέσματα της εκπαίδευσης και του ελέγχου παρουσιάζονται στην εικόνα 1.

### 1.2 Ερώτημα Ε

Επιλέγουμε το διάστημα των $\lambda$ τιμών μας λογαριθμικά, εφόσον η βέλτιστη τιμή κανονικοποίησης είναι πολύ πιο πιθανό να βρίσκεται αρκετά κοντά στο 0. Η λογαριθμική κλίμακα μας επιτρέπει να ψάξουμε πιο πολλές τιμές του $\lambda$ όσο πιο κοντά φτάνουμε στο κάτω όριο αναζήτησης μας, το $10^4$.Η προσέγγιση αυτή χρησιμοποιείται και στην πράξη για κανονικοποίηση $L^2$ [1]

Κατά την εκτέλεση του προγράμματος υπάρχει πιθανότητα να εμφανιστούν ειδοποιήσεις για αριθμητική υπερχείλιση. Για αυτό ευθύνεται η επιλογή πολύ μεγάλης τιμής του $\lambda$, κυρίως στο δίαστημα [8, 10]. Σε αυτό το σημείο η κανονικοποίηση είναι τόσο ισχυρή που αποτρέπει το μοντέλο μας από το να μάθει, και έτσι αυτό μαντεύει πάντα την ίδια κατηγορία.

Στην δική μας περίπτωση το μοντέλο, κρατώντας τις υπόλοιπες υπερπαραμέτρους ίσες με το προηγούμενο υποερώτημα, προτιμά την ελάχιστη τιμή κανονικοποίησης $\lambda = 10^4$ με ακρίβεια επαλήθευσης ίση με 0.979 και ακρίβεια ελέγχου ίση με 0.981. Παρατηρούμε ότι η ακρίβεια ελέγχου με την επιλεγμένη τιμή $\lambda$ είναι μικρότερη από την αντίστοιχη στο υποερώτημα Δ. Αυτό μας υποδεικνύει είτε ότι η βέλτιστη τιμή βρίσκεται έξω (και πιο συγκεκριμένα πριν) από το διάστημα αναζήτησης μας, είτε ότι για την διαφορά ευθύνεται το στατιστικό σφάλμα.

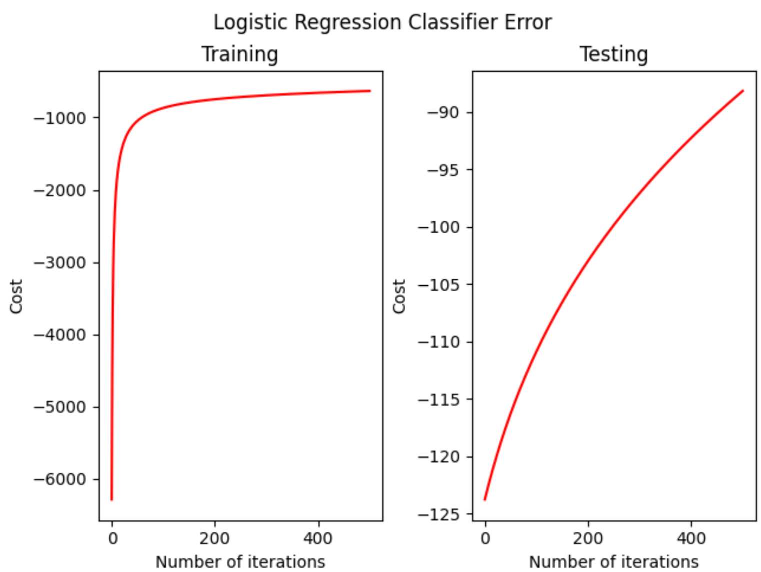


Figure 1: Τα αποτελέσματα της εκπαίδευσης και του ελέγχου στον ταξινομητή μας. Αριστερά: Το κόστος εκπαίδευσης ως συνάρτηση των επαναλήψεων του αλγορίθμου gradient ascent. Δεξία: Το αντίστοιχο κόστος ελέγχου. Υπενθυμίζουμε ότι οι κλίμακες των γραφημάτων δεν είναι ίσες, καθώς ο ήδη εκπαιδευμένος ταξινομητής αρχίζει με πολύ μικρότερο κόστος.

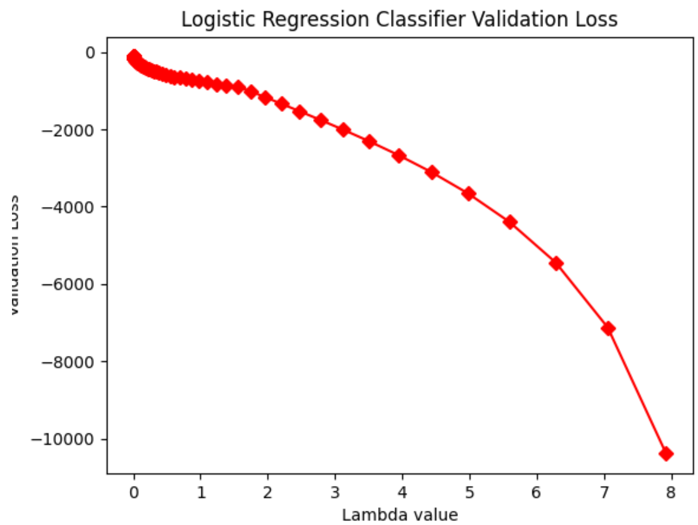


Figure 2: Τα αποτελέσματα της αναζήτησης για το βέλτιστο $\lambda$. Οι ρόμβοι αντιπροσωπεύουν τις τιμές που εξετάσαμε. Παρατηρείστε το πλήθος των τιμών που εξετάστηκαν στην αρχή συγκριτικά με το τέλος του πεδίου αναζήτησής μας.

Τα πλήρη αποτελέσματα αναζήτησης της υπερπαραμέτρου παρουσιάζονται στην εικόνα 2.

# 2 ΜΕΡΟΣ Γ

## 2.1 Ερώτημα ΣΤ

Το νευρωνικό μας δίκτυο έχει υλοποιηθεί στο αρχείο `mlp.py`. Αποτελείται από δύο πίνακες βαρών (h_w για hidden weights, o_w για output weights) και δύο πίνακες bias (h_b για hidden bias, o_b για output bias). Ο πίνακας h_w έχει μέγεθος IxH, ο o_w HxO, ο h_b 1xH και ο o_b 1xO, όπου I=inputSize, H=hiddenNeurons, O=outputSize. Ο κώδικας για την εκτέλεση του δικτύου βρίσκεται στο αρχείο `run_mlp.py`, το οποίο εκτελεί κώδικα και για τα ερωτήματα Η, Θ.

Θα τρέξουμε το μοντέλο με υπερπαραμέτρους m=2, $\eta$=0.2 και `tolerance =0.001`. Το `tolerance` είναι μια υπερ-παράμετρος απαραίτητη για το early stopping, και η οποία καθορίζει πόσο το κόστος πρέπει να έχει μειωθεί για να θεωρείται η τρέχουσα εποχή ”βελτίωση”. Το αποτέλεσμα είναι η ακρίβεια εκπαίδευσης να είναι ίση με 0.978, το τελικό κόστος εκπαίδευσης ίσο με 0.0671, η ακρίβεια ελέγχου ίση με 0.974 και το μέσο κόστος ελέγχου ίσο με 0.0734. Τα πλήρη αποτελέσματα της εκπαίδευσης παρουσιάζονται στην εικόνα 3.

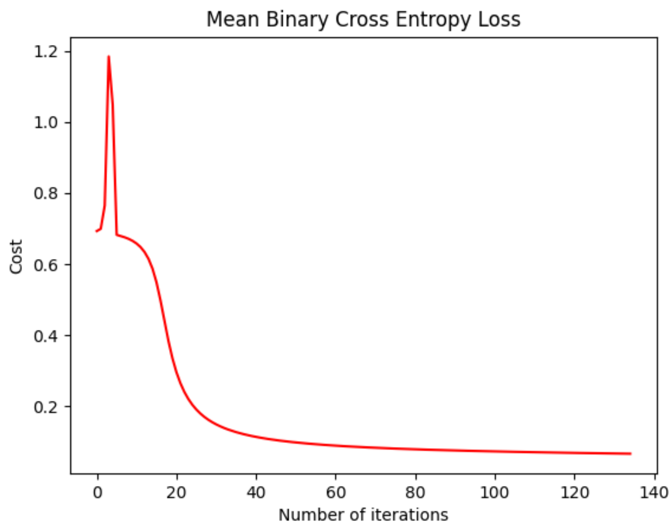


Figure 3: Το κόστος εκπαίδευσης ως συνάρτηση των επαναλήψεων του αλγορίθμου gradient descent.

## 2.2 Ερώτημα Ζ

Ο τύπος της Δυαδικής Διασταυρούμενης Εντροπίας είναι:

$$E_b = -(t \ln \hat{y} + (1-t)\ln(1-\hat{y})) \tag{1}$$

όπου $t$ το διάνυσμα των ετικετών δεδομένων και $\hat{y}$ το της εκτιμώμενης πιθανότητας.

Στην παραπάνω εξίσωση παρατηρούμε ότι η τιμή $t$ είναι πάντοτε γνωστή, αντίθετα με τη $\hat{y}$, επομένως θα παραγωγίσουμε με βάση το $\hat{y}$. Εφόσον $y = f(x) + g(x) \iff y' = f'(x) + g'(x)$ ισχύει ότι:

$$\frac{\partial E_b}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}}(t \ln \hat{y}) + \frac{\partial}{\partial \hat{y}}(1-t)\ln(1-\hat{y}) \tag{2}$$

Αναλύουμε τις επιμέρους συναρτήσεις του αθροίσματος της παραγώγου:

$$\frac{\partial}{\partial \hat{y}}(t \ln \hat{y}) = \frac{\partial}{\hat{y}}(t \ln \hat{y}) + \ln \hat{y} \frac{\partial}{\partial \hat{y}} t = \frac{t}{\hat{y}} + \ln \hat{y} \cdot 0 = \frac{t}{\hat{y}} \tag{3}$$

$$\frac{\partial}{\partial \hat{y}}((t-1)\ln(1-\hat{y})) = (1-t)\frac{\partial}{\partial \hat{y}}(\ln(1-\hat{y})) + \ln(1-\hat{y})\frac{\partial}{\partial \hat{y}}(1-t) = \frac{1-t}{1-\hat{y}} + \ln(1-\hat{y})\cdot 0 = \frac{1-t}{1-\hat{y}} \tag{4}$$

Επομένως, αθροίζοντας τις 3, 4, η 2 γίνεται:

$$\frac{\partial E_b}{\partial \hat{y}} = -(\frac{t}{\hat{y}} - \frac{1-t}{1-\hat{y}}) = (\frac{t}{\hat{y}} + \frac{1-t}{1-\hat{y}}) \tag{5}$$

Ο κώδικας επαλήθευσης του αναλυτικού τύπου βρίσκεται στο αρχείο `test_mlp.py`.

### 2.3 Ερώτημα Η

Εκτελούμε grid search για τις παραμέτρους $m, \eta$. Επιλέγουμε το διάστημα των $\eta$ τιμών μας στο χώρο αναζήτησης $[0.5, 10^{-5}]$, εκθετικά κοντά στο $0.5$ εφόσον εμπειρικά αναμένουμε ότι η βέλτιστη τιμή της θα βρίσκεται κοντά στις τιμές $0.5, 0.01, 0.01$. Το πρόγραμμα χρειάζεται περίπου 5 λεπτά για να βγάλει το βέλτιστο συνδυασμό υπερπαραμέτρων, το οποίο για το συγκεκριμένο δίκτυο είναι το $(\eta = 0.5, m = 0.5, E = 172)$ με κόστος επικύρωσης ίσο με $0.0423$.

### 2.4 Ερώτημα Θ

Χρησιμοποιούμε την μέθοδο `predict` του μοντέλου μας για να αποκτήσουμε τις προβλεπόμενες ετικέτες του. Η μέθοδος αυτή χρησιμοποιεί μεθόδους της numpy, ισοδύναμες της κατηγοριοποιήσης με βρόγχο. Στο σύνολο ελέγχου το μοντέλο με τις βελτιστοποιημένες παραμέτρους επιτυγχάνει ακρίβεια ελέγχου ίση με $0.979$ και κόστος ελέγχου ίσο με .