



ΗΜΜΥ · ΕΛΜΕΠΑ · 9ο εξάμηνο

Μηχανική Μάθηση και Εξόρυξη Γνώσης

Διάλεξη 3: Γραμμικά μοντέλα και βελτιστοποίηση

Λευτέρης Κουμάκης

Από την προηγούμενη φορά



Πλαισίωση

Τύπος προβλήματος και μέτρο απόδοσης (RMSE, MAE)



Προετοιμασία

Συμπλήρωση, one-hot, κλιμάκωση, λογάριθμος



Pipelines

fit μόνο στο train, χωρίς διαρροή δεδομένων



Πρώτα μοντέλα

Η γραμμική παλινδρόμηση υποπροσαρμοζόταν

Ο χάρτης της σημερινής διάλεξης



Βιβλιογραφία: Géron Κεφ. 4 · ISL Κεφ. 3, 4.3, 6.2

ΕΝΟΤΗΤΑ 1

Γραμμική παλινδρόμηση

Το μοντέλο, η συνάρτηση κόστους και η κλειστή λύση

Το μοντέλο της γραμμικής παλινδρόμησης

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$

Σύμβολο	Σημασία
$\hat{y}$	Η πρόβλεψη
n	Πλήθος χαρακτηριστικών
x_i	Η τιμή του i -οστού χαρακτηριστικού
θ_0	Σταθερός όρος (bias)
$\theta_1 \dots \theta_n$	Βάρη των χαρακτηριστικών



Παράδειγμα

$$\hat{y} = 20 + 1,5 \cdot (\text{m}^2) + 10 \cdot (\text{δωμάτια})$$

80 m², 3 δωμάτια:
 $20 + 120 + 30 = 170$ χιλ. €

Διανυσματική μορφή

$$\hat{y} = \theta^T x = \theta \cdot x$$

$$\hat{y} = X\theta \text{ (όλο το dataset)}$$

	x_0	m^2	δωμάτια
θ	20	1,5	10
x	1	80	3
$\theta_i \cdot x_i$	20	120	30

Άθροισμα:
 $20 + 120 + 30 = 170$

Η συνάρτηση κόστους

Mean Squared Error

$$\text{MSE}(\theta) = (1/m) \cdot \sum_i (\theta^T \mathbf{x}_i - y_i)^2$$



Στόχος εκπαίδευσης

Βρες το θ που δίνει το ελάχιστο $\text{MSE}(\theta)$ στο training set



Γιατί MSE και όχι RMSE;

Ίδιο ελάχιστο, απλούστερη παράγωγος



Δύο δρόμοι

Κλειστή λύση (κανονική εξίσωση) ή επαναληπτική (gradient descent)

Η κανονική εξίσωση (normal equation)

$$\theta = (X^T X)^{-1} X^T y$$



Πλεονεκτήματα

- Ακριβής λύση, χωρίς επαναλήψεις
- Χωρίς υπερπαραμέτρους (όπως ρυθμός μάθησης)
- Γραμμική πολυπλοκότητα ως προς m



Μειονεκτήματα

- Αντιστροφή πίνακα $(n+1) \times (n+1)$: $\sim O(n^{2.4})$ έως $O(n^3)$
- Αργή με δεκάδες χιλιάδες χαρακτηριστικά
- Πρέπει όλα τα δεδομένα να χωρούν στη μνήμη

Παράδειγμα: η κανονική εξίσωση με το χέρι

x	y
1	2
2	3
3	5

$$X = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \quad y = \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix}$$

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

1 $X^T X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} = \begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix}$

2 $X^T X = (1/6) * \begin{bmatrix} 14 & -6 \\ -6 & 3 \end{bmatrix}$

3 $X^T y = [10, 23]$

4 $\theta = (1/6) \cdot [2, 9] = [0,33 \cdot 1,5]$

Μοντέλο: $\hat{y} = 0,33 + 1,5 \cdot x$

Ερμηνεία συντελεστών: το παράδειγμα Advertising (ISL)

$$\text{πωλήσεις} \approx 7,03 + 0,0475 \times \text{TV}$$



Κλίση $\beta_1 = 0,0475$

+1.000 \$ σε TV $\rightarrow$ περίπου
+47,5 μονάδες πωλήσεων



Σταθερός όρος $\beta_0 =$
7,03

Χωρίς διαφήμιση TV: περίπου
7.030 μονάδες



Στατιστική
σημαντικότητα

Το ISL εξετάζει τυπικά
σφάλματα και p-values για
κάθε συντελεστή

Ο συντελεστής προσδιορισμού R^2

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^m (y_i - \hat{y}_i)^2}{\sum_{i=1}^m (y_i - \bar{y})^2}$$

x	y	$\hat{y} = 0,33 + 1,5x$	$(y - \hat{y})^2$	$(y - \bar{y})^2$
1	2	1,83	0,03	1,78
2	3	3,33	0,11	0,11
3	5	4,83	0,03	2,78
		Σ	0,17	4,67

SSE (Sum of Squared Errors)

SST (Total Sum of Squares)

Διάλεξη 3 · Γραμμικά μοντέλα

$$R^2 \approx 0,96$$

$R^2 = 1$: τέλεια προσαρμογή

$R^2 = 0$: όσο και η μέση τιμή

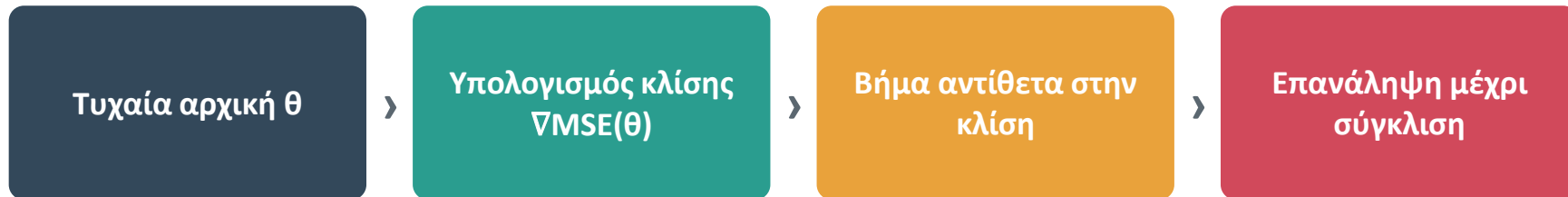
$R^2 < 0$: χειρότερο από τη μέση τιμή

ΕΝΟΤΗΤΑ 2

Gradient descent

Επαναληπτική βελτιστοποίηση: ο αλγόριθμος πίσω από τη βαθιά μάθηση

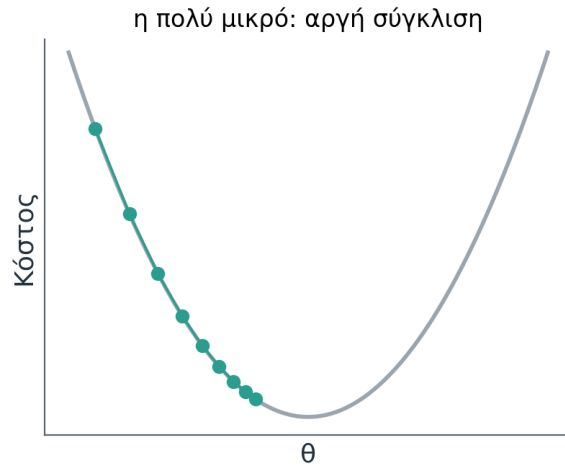
Η διαίσθηση: κατεβαίνοντας βουνό μέσα σε ομίχλη



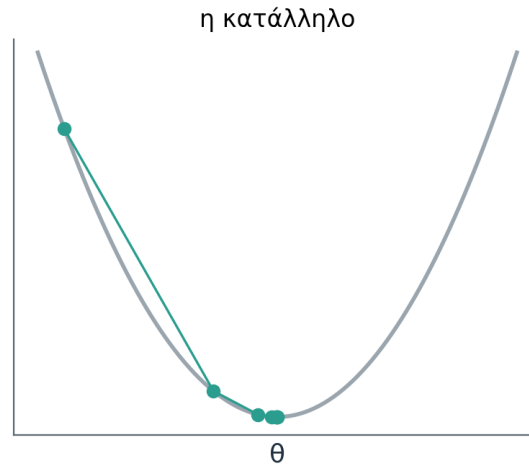
$$\theta \leftarrow \theta - \eta \cdot \nabla \text{MSE}(\theta)$$

η : ρυθμός μάθησης (learning rate), το μέγεθος του βήματος

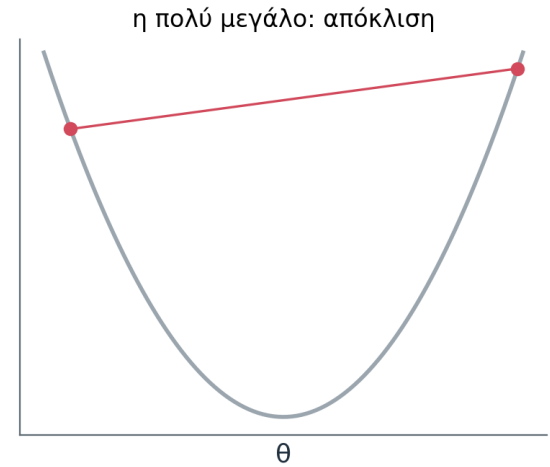
Ο ρυθμός μάθησης η



Φτάνει, αλλά μετά από πολλά βήματα



Λίγα βήματα ως το ελάχιστο



Κάθε βήμα οδηγεί ψηλότερα

Παγίδες του gradient descent



Τοπικά ελάχιστα

Ο αλγόριθμος «κολλά» σε ελάχιστο που δεν είναι το ολικό



Οροπέδια (plateaus)

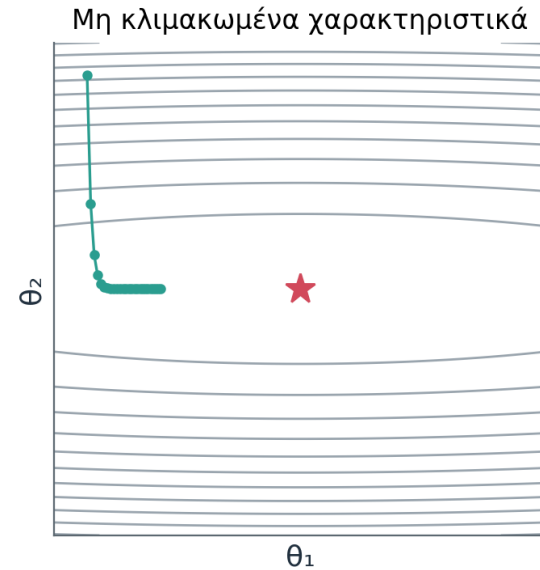
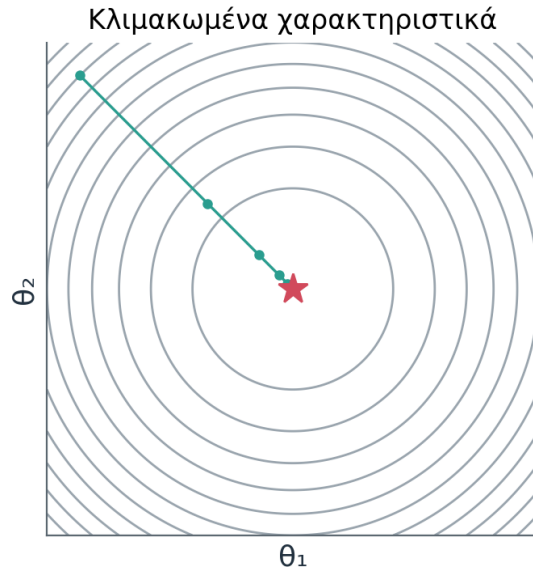
Σχεδόν μηδενική κλίση: η πρόοδος σταματά για πολλές εποχές



Καλά νέα για το MSE

Κυρτή συνάρτηση: ένα μόνο ελάχιστο, χωρίς παγίδες

Κλιμάκωση χαρακτηριστικών και gradient descent



Αριστερά: ευθεία πορεία προς το ελάχιστο · Δεξιά: μακρύς δρόμος κατά μήκος της «κοιλιάς»

Batch gradient descent: η κλίση του MSE

$$\frac{\partial MSE}{\partial \theta_j} = \frac{2}{m} \sum_{i=1}^m (\theta^T x_i - y_i) x_{ij}$$

$$\nabla MSE(\theta) = \frac{2}{m} X^T (X\theta - y)$$

- Κάθε βήμα: όλο το training set (γι' αυτό «batch»)
 - Υπολογίζουμε τις προβλέψεις για όλα τα m δείγματα.
 - Υπολογίζουμε τα σφάλματα όλων των δειγμάτων.
 - Υπολογίζουμε τη μέση κλίση από όλα τα δείγματα.
 - Ενημερώνουμε τους συντελεστές θ
- Ακριβής κλίση, ομαλή πορεία προς το ελάχιστο
- Κλιμακώνεται καλά με πολλά χαρακτηριστικά, όχι με πολλά δείγματα

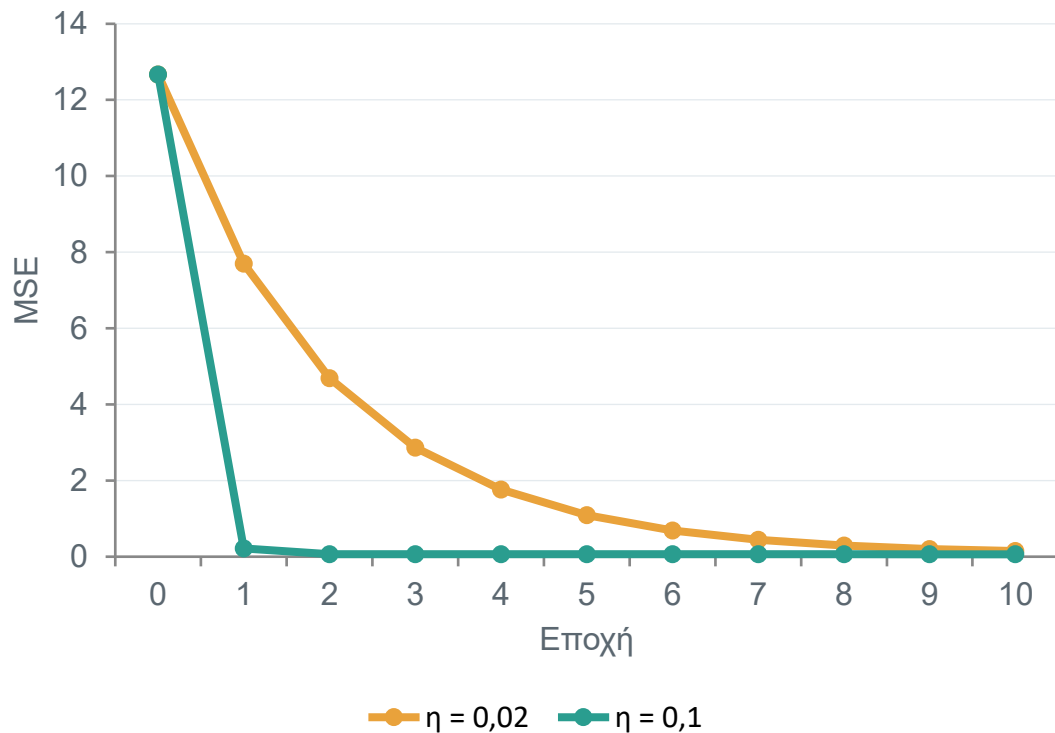
Παράδειγμα: ένα βήμα gradient descent με το χέρι

	θ_0	θ_1	MSE
Αρχή	0	0	12,67
Κλίση	-6,67	-15,33	
Μετά από 1 βήμα ($\eta = 0,1$)	0,67	1,53	0,22
Λύση κανονικής εξίσωσης	0,33	1,50	0,06

Δεδομένα: (1, 2), (2, 3), (3, 5)

- Ένα βήμα μείωσε το MSE από 12,67 σε 0,22
- Τα επόμενα βήματα πλησιάζουν τη λύση (0,33 · 1,50)

Το ίδιο παράδειγμα με τρεις ρυθμούς μάθησης



$\eta = 0,25$

Απόκλιση:

Εποχή 0: 12,7

Εποχή 1: 39,7

Εποχή 2: 124,6

Εποχή 3: 391,8

...

Στοχαστικό gradient descent (SGD)



Πλεονεκτήματα

- Πολύ γρήγορο: ένα δείγμα ανά βήμα
- Εκπαίδευση σε τεράστια datasets (out-of-core)
- Ξεφεύγει από τοπικά ελάχιστα

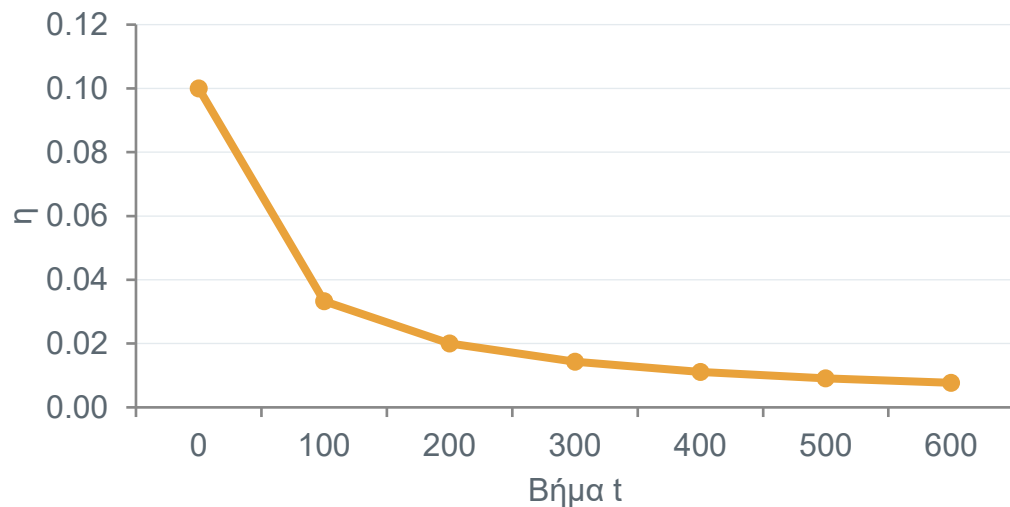


Μειονεκτήματα

- Θορυβώδης πορεία, «χορεύει» γύρω από το ελάχιστο
- Χρειάζεται learning schedule (μείωση του η)
- Τα δεδομένα πρέπει να είναι ανακατεμένα (shuffle)

Παράδειγμα: πρόγραμμα ρυθμού μάθησης

$$\eta(t) = t_0 / (t + t_1), \quad t_0 = 5, \quad t_1 = 50$$



t	$\eta(t)$
0	0,100
50	0,050
450	0,010

Mini-batch gradient descent

Batch: όλα τα m δείγματα

Ακριβής, αργός σε μεγάλα datasets



Mini-batch: 32–256 δείγματα

Ισορροπία ταχύτητας και
σταθερότητας · GPU



Στοχαστικό: 1 δείγμα

Γρήγορος, θορυβώδης

Εποχή (epoch) = ένα πλήρες πέρασμα από όλα τα δεδομένα. Με $m = 10.000$ και mini-batch 100, μία εποχή = 100 βήματα.

Σύγκριση αλγορίθμων για γραμμική παλινδρόμηση

Αλγόριθμος	Μεγάλο m	Out-of-core	Μεγάλο n	Υπερπαράμ.	Κλιμάκωση;	scikit-learn
Κανονική εξίσωση	Γρήγορος	Όχι	Αργός	0	Όχι	—
SVD	Γρήγορος	Όχι	Αργός	0	Όχι	LinearRegression
Batch GD	Αργός	Όχι	Γρήγορος	2	Ναι	—
Στοχαστικό GD	Γρήγορος	Ναι	Γρήγορος	≥ 2	Ναι	SGDRegressor
Mini-batch GD	Γρήγορος	Ναι	Γρήγορος	≥ 2	Ναι	—

m = πλήθος δειγμάτων · n = πλήθος χαρακτηριστικών · Πηγή: Géron, Κεφ. 4



Ερώτηση Slido 1 από 3



Εκπαιδεύετε με gradient descent και το κόστος ΑΥΞΑΝΕΤΑΙ από εποχή σε εποχή. Ποια είναι η πιθανότερη αιτία;

A

Πολύ μικρός ρυθμός μάθησης

B

Πολύ μεγάλος ρυθμός μάθησης

Γ

Πολύ λίγες εποχές

Δ

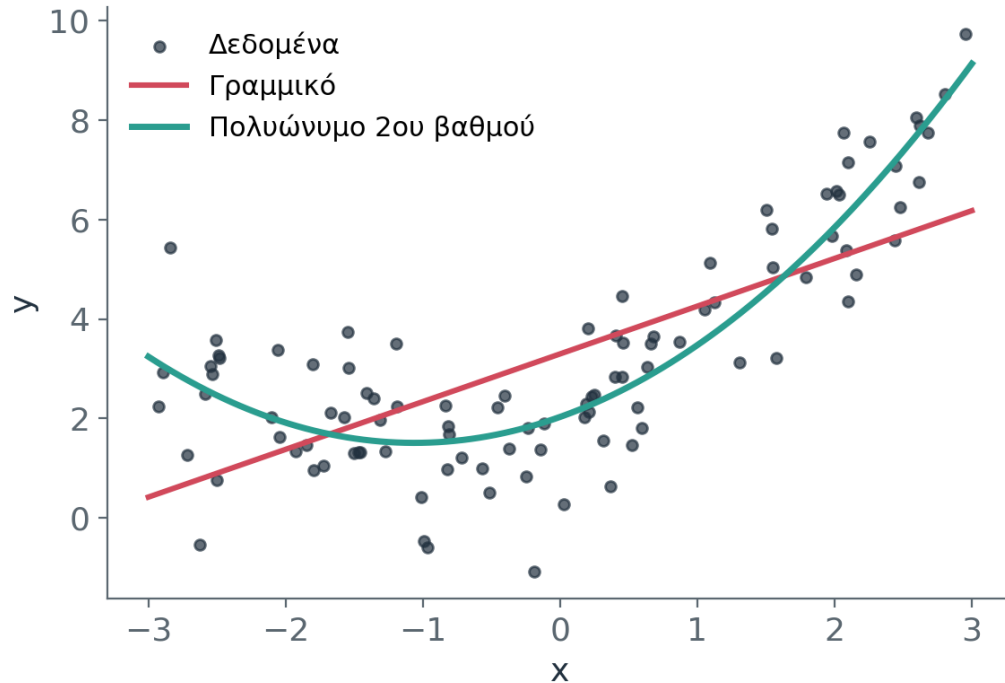
Η συνάρτηση MSE δεν είναι κυρτή

ΕΝΟΤΗΤΑ 3

Πολυώνυμα και ομαλοποίηση

Πολυωνυμική παλινδρόμηση, καμπύλες μάθησης, Ridge, Lasso, early stopping

Πολυωνυμική παλινδρόμηση



- Αληθινή σχέση: $y = 0,5x^2 + x + 2$
- Εκτίμηση: $\hat{y} = 0,46x^2 + 0,98x + 2,03$
- Η ευθεία υποπροσαρμόζεται
- `PolynomialFeatures(degree=2) + LinearRegression`

Προσθέτουμε ως νέα χαρακτηριστικά τις δυνάμεις των αρχικών (x^2 , x^3 , ...) και εκπαιδεύουμε γραμμικό μοντέλο

Προσοχή: η «έκρηξη» των χαρακτηριστικών

$$\text{πλήθος} = (n + d)! / (d! \cdot n!)$$

10

$n = 2, d = 3$

286

$n = 10, d = 3$

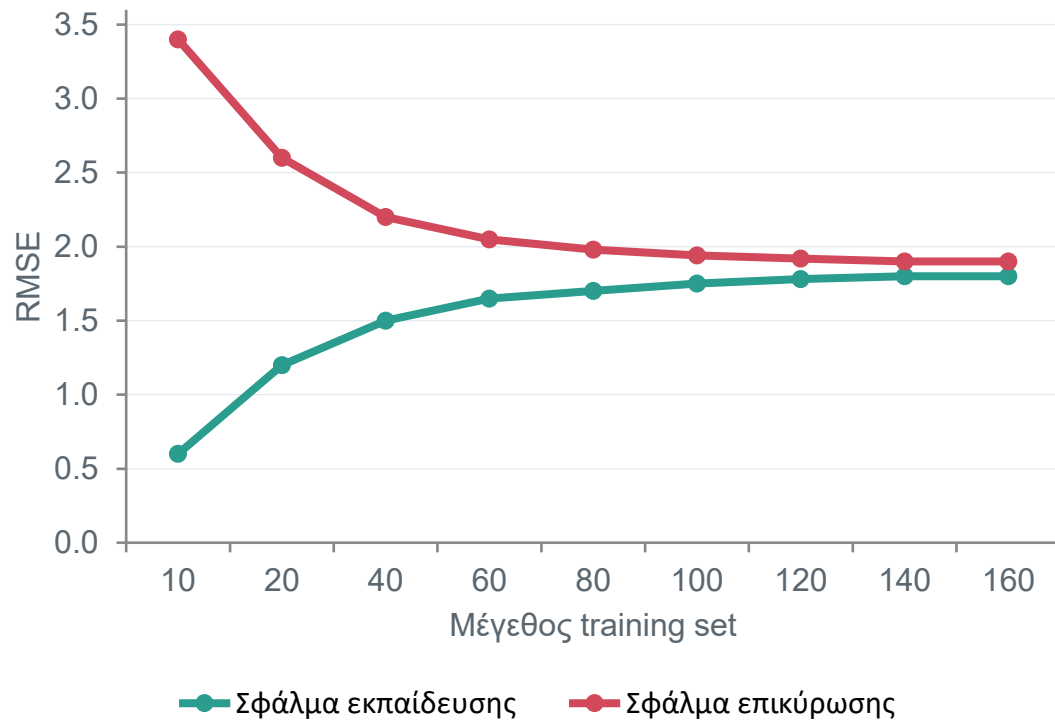
> 176.000

$n = 100, d = 3$

Περιλαμβάνει και τους όρους αλληλεπίδρασης (π.χ. x_1x_2 , $x_1^2x_2$), γι' αυτό το πολυωνυμικό μοντέλο βρίσκει σχέσεις μεταξύ χαρακτηριστικών.

n χαρακτηριστικά με βαθμό d

Καμπύλες μάθησης: διάγνωση υποπροσαρμογής

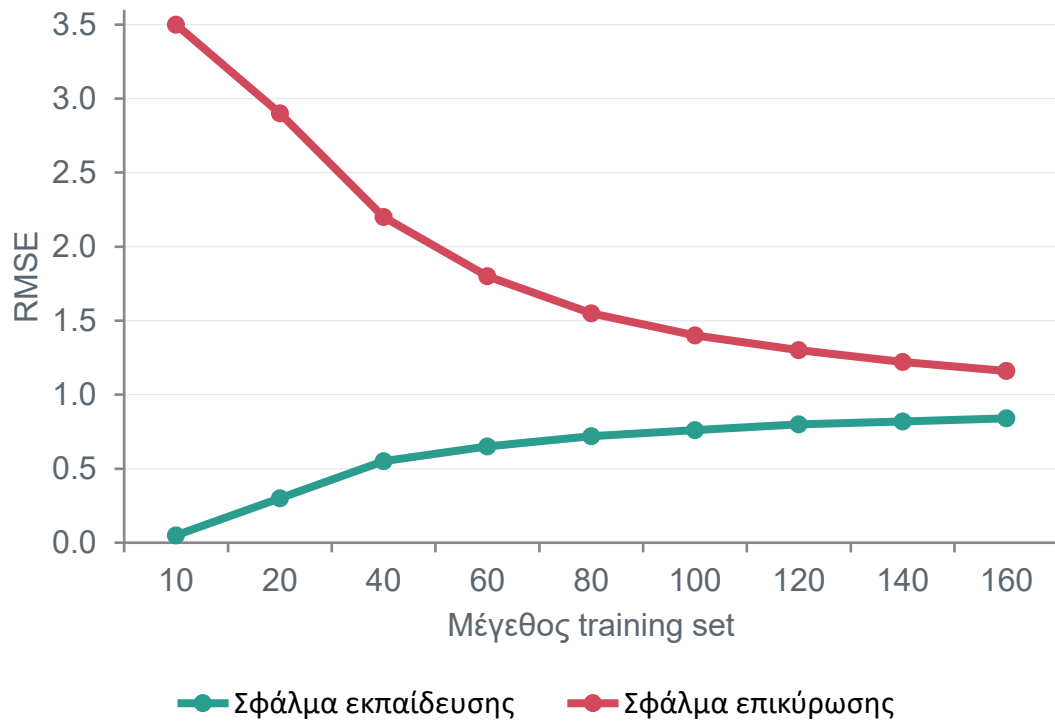


Υποπροσαρμογή

- Υψηλό οροπέδιο και στις δύο καμπύλες
- Μικρό κενό μεταξύ τους

Λύση: πιο σύνθετο μοντέλο, όχι περισσότερα δεδομένα

Καμπύλες μάθησης: διάγνωση υπερπροσαρμογής



Υπερπροσαρμογή

- Χαμηλό σφάλμα εκπαίδευσης
- Μεγάλο κενό από το σφάλμα επικύρωσης

Λύση: περισσότερα δεδομένα, ομαλοποίηση, απλούστερο μοντέλο

Τα τρία συστατικά του σφάλματος γενίκευσης

$$\text{Σφάλμα} = \text{Μεροληψία}^2 + \text{Διακύμανση} + \text{Μη αναγώγιμο σφάλμα}$$



Μεροληψία (bias)

Λάθος υποθέσεις, π.χ.
γραμμικό για τετραγωνική
σχέση → υποπροσαρμογή



Διακύμανση (variance)

Υπερβολική ευαισθησία σε
μικρές αλλαγές των
δεδομένων →
υπερπροσαρμογή



Μη αναγώγιμο

Θόρυβος των δεδομένων·
μειώνεται μόνο με καθαρισμό

Ομαλοποίηση: περιορίζουμε τα βάρη

$$\text{Κόστος} = \text{MSE}(\theta) + \alpha \cdot \text{ποινή}(\theta)$$



Ridge (ℓ_2)

ποινή = $\sum \theta_i^2$

Μικραίνει όλα τα βάρη



Lasso (ℓ_1)

ποινή = $\sum |\theta_i|$

Μηδενίζει μερικά βάρη

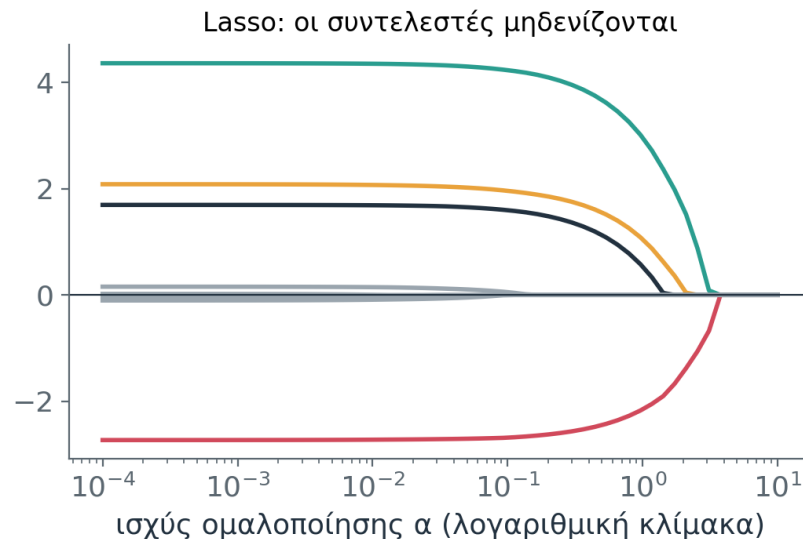
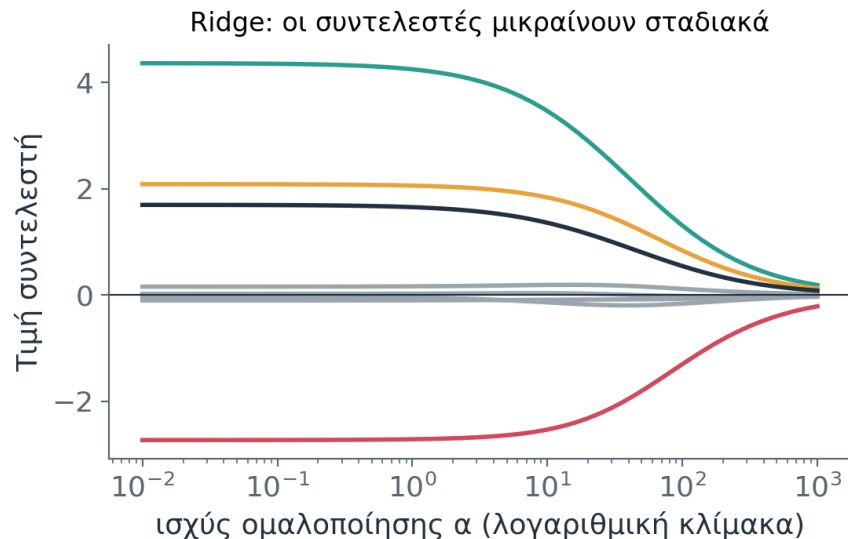


Elastic Net

Συνδυασμός ℓ_1 και ℓ_2 με λόγο r

Ο σταθερός όρος θ_0 δεν ομαλοποιείται · Πριν από την ομαλοποίηση: κλιμάκωση!

Ridge έναντι Lasso: διαδρομές συντελεστών



Ridge: κρατά όλα τα χαρακτηριστικά με μικρότερα βάρη

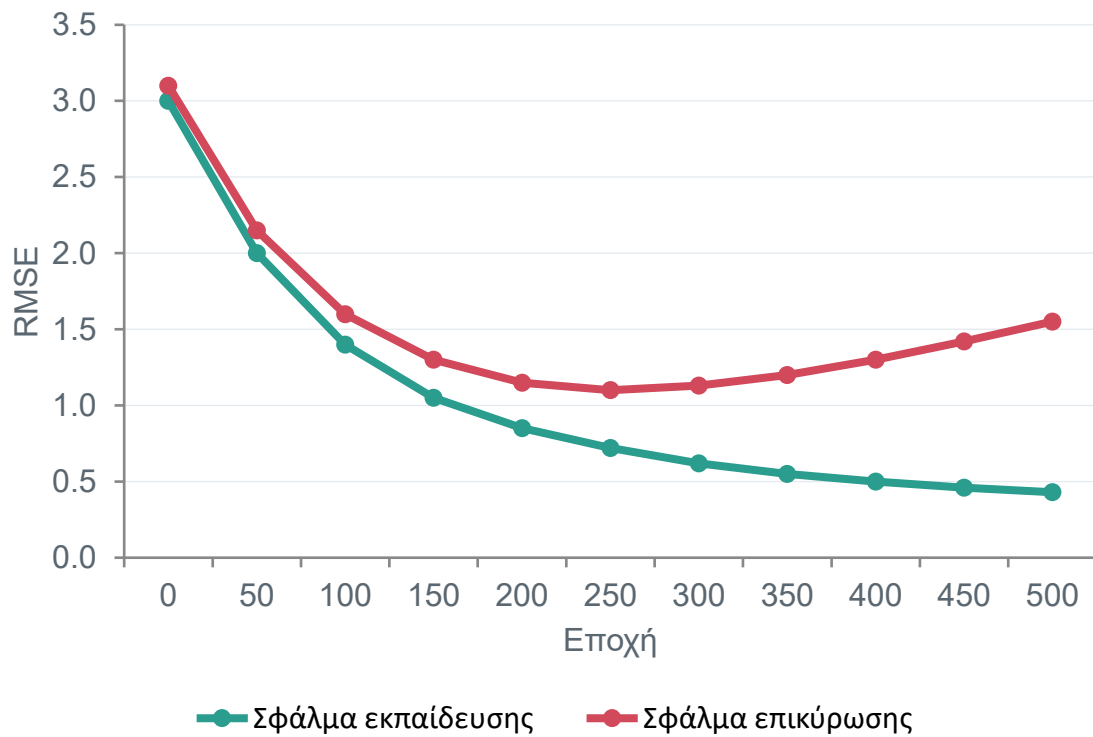
Lasso: παράγει αραιό μοντέλο (feature selection)

Ποια ομαλοποίηση να επιλέξω;

Μέθοδος	Ποινή	Πότε	scikit-learn
Ridge	$\alpha \cdot \sum \theta_i^2$	Προεπιλογή· όλα τα χαρακτηριστικά συνεισφέρουν λίγο	Ridge(alpha=...)
Lasso	$\alpha \cdot \sum \theta_i $	Λίγα χρήσιμα χαρακτηριστικά, θέλουμε επιλογή	Lasso(alpha=...)
Elastic Net	$r \cdot \ell_1 + (1-r) \cdot \ell_2$	Πολλά ή συσχετισμένα χαρακτηριστικά	ElasticNet(alpha, l1_ratio)

Το α επιλέγεται με cross-validation: RidgeCV, LassoCV, ElasticNetCV (ISL 6.2.3).

Early stopping: σταματάμε στο σωστό σημείο



Κανόνες

Παρακολουθούμε το σφάλμα επικύρωσης σε κάθε εποχή.

Κρατάμε το καλύτερο μοντέλο (εδώ: εποχή ~250).



Ερώτηση Slido 2 από 3



Έχετε 200 χαρακτηριστικά και υποψιάζεστε ότι μόνο λίγα είναι πραγματικά χρήσιμα. Θέλετε το μοντέλο να μηδενίσει αυτόματα τα υπόλοιπα. Ποιο επιλέγετε;

A

Γραμμική παλινδρόμηση χωρίς ομαλοποίηση

B

Ridge

Γ

Lasso

Δ

Πολυωνυμική παλινδρόμηση

ΕΝΟΤΗΤΑ 4

Λογιστική και softmax

Από την παλινδρόμηση στην ταξινόμηση

Γιατί όχι γραμμική παλινδρόμηση για κλάσεις;



Γραμμική παλινδρόμηση

$$\hat{y} = \theta^T x$$

- Τιμές από $-\infty$ έως $+\infty$
- «Πιθανότητα» $-0,2$ ή $1,3$ δεν έχει νόημα
- Ευαίσθητη σε ακραίες τιμές

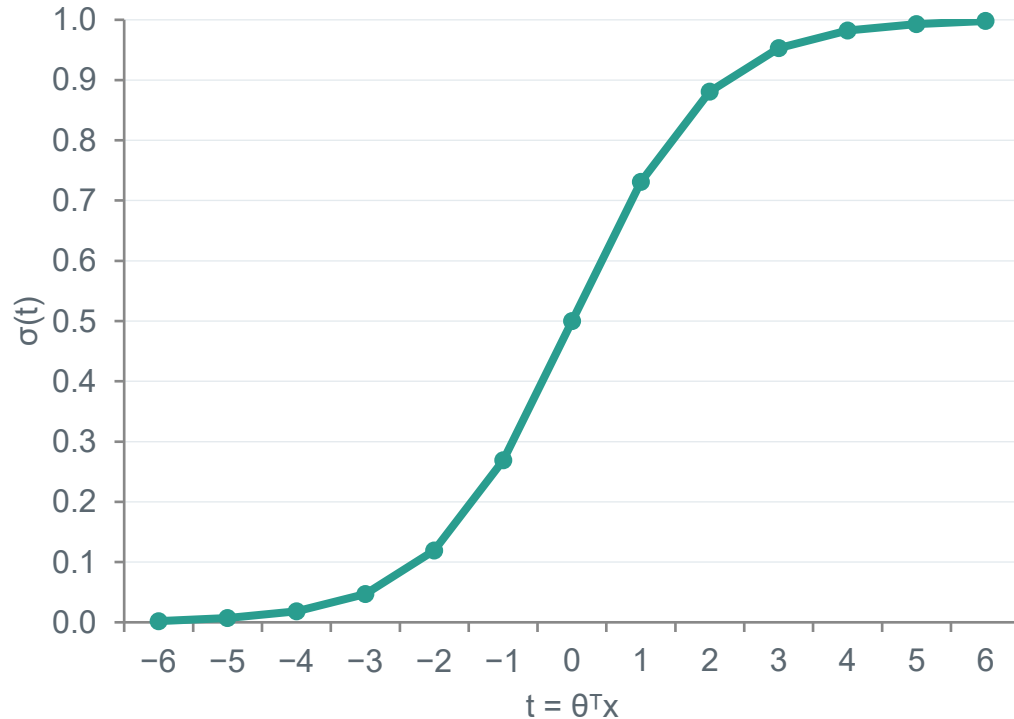


Λογιστική παλινδρόμηση

$$\hat{p} = \sigma(\theta^T x)$$

- Τιμές πάντα στο $(0, 1)$
- Ερμηνεύεται ως πιθανότητα
- Από κάτω, το ίδιο γραμμικό σκορ

Η σιγμοειδής (λογιστική) συνάρτηση



$$\sigma(t) = 1 / (1 + e^{-t})$$

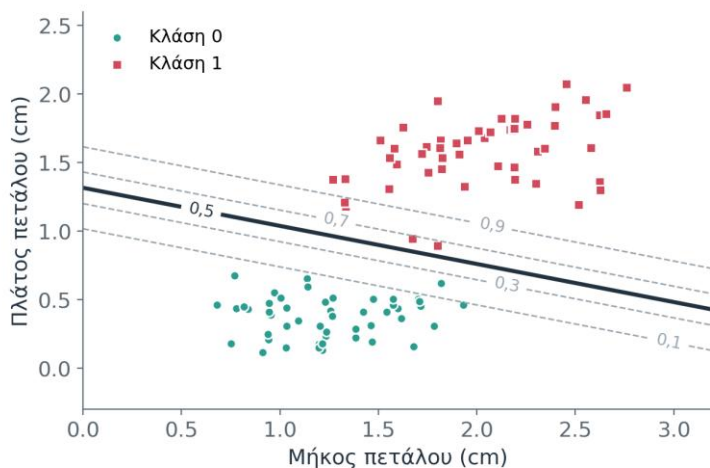
t	$\sigma(t)$
-2	0,12
0	0,50
2	0,88

Το μοντέλο και ο κανόνας απόφασης

$$\hat{p} = \sigma(\theta^T x)$$

$$z = \theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$\hat{y} = 1 \text{ αν } \hat{p} \geq 0,5 \Leftrightarrow \theta^T x \geq 0$$



- Συνεχής γραμμή: όριο απόφασης ($\hat{p} = 0,5$)
- Διακεκομμένες: ισοϋψείς πιθανότητας 0,1 έως 0,9
- Όσο πιο μακριά από το όριο, τόσο πιο «σίγουρη» η πρόβλεψη

Παράδειγμα: πιθανότητα επιτυχίας στις εξετάσεις

$$\hat{p} = \sigma(-4 + 1,5 \cdot \text{ώρες διαβάσματος})$$

Ώρες	$t = -4 + 1,5 \cdot \text{ώρες}$	$\hat{p} = \sigma(t)$	Πρόβλεψη
2	-1	0,27	Αποτυχία
2,67	0	0,50	Όριο απόφασης
4	2	0,88	Επιτυχία

Το όριο απόφασης είναι στις 2,67 ώρες: εκεί ισχύει $\theta^T x = 0$.

Ερμηνεία: odds και log-odds

$$\log(p / (1 - p)) = \theta_0 + \theta_1 x_1 + \dots$$

$$e^{1,5} \approx 4,48$$

πολλαπλασιαστής των odds ανά επιπλέον ώρα

$$\approx 7,3$$

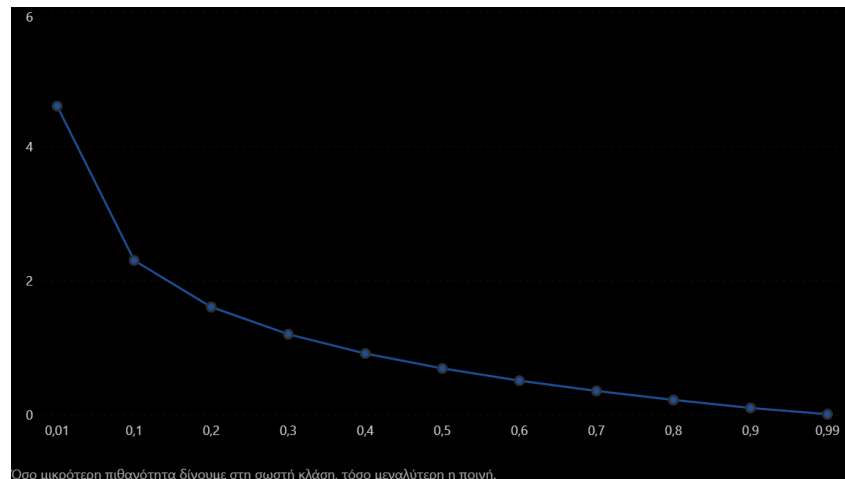
odds επιτυχίας με 4 ώρες (0,88 / 0,12)

Γι' αυτό η λογιστική παλινδρόμηση είναι δημοφιλής στην ιατρική και τις κοινωνικές επιστήμες: οι συντελεστές ερμηνεύονται άμεσα.

Η συνάρτηση κόστους: log loss

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i)]$$

Πραγματική κλάση	$\hat{p}$	Κόστος	Σχόλιο
1	0,9	0,105	Σωστό και σίγουρο
1	0,5	0,693	Αβέβαιο
1	0,1	2,303	Λάθος και σίγουρο
0	0,1	0,105	Σωστό και σίγουρο





Ερώτηση Slido 3 από 3



Σε μοντέλο λογιστικής παλινδρόμησης, για ένα δείγμα ισχύει $\theta^T x = 0,8$. Τι προβλέπει το μοντέλο με κατώφλι 0,5;

A

Κλάση 0, με $\hat{p} \approx 0,31$

B

Κλάση 1, με $\hat{p} \approx 0,69$

Γ

Κλάση 1, με $\hat{p} = 0,8$

Δ

Δεν μπορούμε να ξέρουμε χωρίς το x

Softmax παλινδρόμηση: πολλές κλάσεις

$$\hat{p}_k = \frac{\exp(s_k)}{\sum_j \exp(s_j)}$$

$$\hat{y} = \operatorname{argmax}_k \hat{p}_k$$

Χαρακτηριστικά x

›

Σκορ $s_k = \theta^{(k)\top} x$

›

Softmax

›

Πιθανότητες $\hat{p}_k$

Γενίκευση της λογιστικής για $K > 2$ κλάσεις (*multinomial logistic regression*)

Παράδειγμα: softmax και cross-entropy

Κλάση	Σκορ s_k	$\exp(s_k)$	$\hat{p}_k$
A	2,0	7,389	0,659
B	1,0	2,718	0,242
Γ	0,1	1,105	0,099
Σ		11,212	1,000

$$\text{Cross-entropy} = -\sum y_k \log \hat{p}_k$$

$$-\log(0,659) \approx 0,417$$

Με 2 κλάσεις, η cross-entropy ταυτίζεται με το log loss.

Στον κώδικα

```
from sklearn.linear_model import (LinearRegression, SGDRegressor,  
                                   Ridge, Lasso, ElasticNet, LogisticRegression)  
from sklearn.preprocessing import PolynomialFeatures, StandardScaler  
  
sgd = SGDRegressor(max_iter=1000, tol=1e-5, penalty=None, eta0=0.01)  
poly = make_pipeline(PolynomialFeatures(degree=2), StandardScaler(),  
                     Ridge(alpha=1.0))  
lasso = Lasso(alpha=0.1)  
log_reg = LogisticRegression(C=1.0)           #  $C = 1/\alpha$   
log_reg.fit(X_train, y_train)  
log_reg.predict_proba(X_new)                 # πιθανότητες ανά κλάση
```

Τα μοντέλα της διάλεξης με μια ματιά

Μοντέλο	Έξοδος	Κόστος	Εκπαίδευση
Γραμμική παλινδρόμηση	$\theta^T x$	MSE	Κανονική εξίσωση / GD
Ridge / Lasso / Elastic Net	$\theta^T x$	MSE + ποινή	Κλειστή λύση ή GD / coordinate descent
Λογιστική	$\sigma(\theta^T x)$	Log loss	GD (κυρτό)
Softmax	$\text{softmax}(\theta x)$	Cross-entropy	GD (κυρτό)

Γραμμικό σκορ + μη γραμμική συνάρτηση = ο τεχνητός νευρώνας.

Σύνοψη

1

Γραμμική παλινδρόμηση: $\hat{y} = \theta^T x$, κόστος MSE

2

Κλειστή λύση: $\theta = (X^T X)^{-1} X^T y$

3

Gradient descent: $\theta \leftarrow \theta - \eta \nabla \text{MSE}$, προσοχή στο η και στην κλιμάκωση

4

Καμπύλες μάθησης: διάγνωση υπο- και υπερπροσαρμογής

5

Ridge μικραίνει, Lasso μηδενίζει, early stopping σταματά

6

Λογιστική: $\hat{p} = \sigma(\theta^T x)$, log loss · softmax για K κλάσεις

Για την επόμενη διάλεξη



Διάβασμα

- Géron, Κεφ. 4 (και ασκήσεις)
- ISL, Κεφ. 3, 4.3 και 6.2
- Géron, Κεφ. 3 (για τη Διάλεξη 4)
- ISL, Κεφ. 4.4–4.5 και 5.1



Εξάσκηση

- Υλοποιήστε batch GD σε NumPy για το παράδειγμα (1,2), (2,3), (3,5)
- Δοκιμάστε $\eta = 0,02 \cdot 0,1 \cdot 0,25$

Διάλεξη 4: Ταξινόμηση I – βασικές έννοιες και αξιολόγηση



Ερωτήσεις;

koumakis@hmu.gr · Ώρες γραφείου: Τρίτη 14:00–16:00