

Πανεπιστήμιο Κύπρου - Τεχνητή Νοημοσύνη

MAI612 - ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

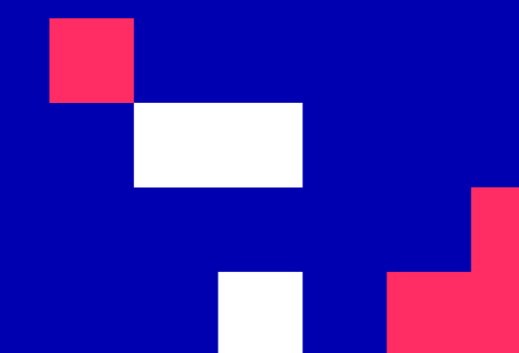
Διάλεξη 5: Αξιολόγηση και βελτίωση μοντέλου

Βασίλης Βασιλειάδης, PhD

Χειμερινό Εξάμηνο 2022/23



CYENS
CENTRE OF EXCELLENCE



MAI4CAREU

Master programmes in Artificial
Intelligence 4 Careers in Europe



Επανάληψη





Παλινδρόμηση

- Η παλινδρόμηση είναι το εποπτευόμενο μαθησιακό πρόβλημα της πρόβλεψης μιας συνεχούς τιμής
- Παλινδρόμηση k -κοντινότερου γείτονα
 - Μη παραμετρικό μοντέλο
 - Πρόβλεψη είναι ο μέσος όρος των πλησιέστερων γειτόνων K
 - $K=1$: θορυβώδης, $1 < K < m$ καταγράφει καλύτερα την τάση, $K=m$ παράγει πάντα το μέσο όρο όλων
 - Τα δυνατά σημεία: δεν υπάρχει χρόνος προπόνησης, χειρίζεται μη γραμμικά προβλήματα,...
 - Αδυναμίες: η πρόβλεψη γίνεται πιο αργή καθώς το σύνολο δεδομένων γίνεται μεγαλύτερο,...
- Γραμμική παλινδρόμηση
 - Παραμετρικό μοντέλο: 1 παράμετρος ανά διάσταση + όρος τομής
 - Πρόβλεψη είναι το εσωτερικό γινόμενο του διανύσματος παραμέτρων και του διανύσματος εισόδου
 - Η μάθηση μπορεί να γίνει είτε με τη χρήση μεθόδων που βασίζονται στο gradient είτε με τον υπολογισμό της αναλυτικής λύσης.





Παλινδρόμηση

- Η γραμμική παλινδρόμηση βελτιστοποιεί το ΜΤΣ, το οποίο έχει κυρτό σχήμα (μονό βέλτιστο)
- Το Gradient Descent αφορά τον υπολογισμό των μερικών παραγώγων μιας συνάρτησης σε ένα σημείο και την αλλαγή του σημείου με την προσθήκη μιας τιμής ανάλογης με την αρνητική κλίση.
- Ένα μικρό ποσοστό μάθησης οδηγεί σε βραδύτερη σύγκλιση, ένα μεγάλο ποσοστό μάθησης μπορεί να οδηγήσει σε απόκλιση
- Διάνυσμα κλίσης του ΜΤΣ για γραμμική παλινδρόμηση
- Ανάγκη χρήσης κλιμάκωσης χαρακτηριστικών με κάθοδο κλίσης
- Γραμμική οπισθοδρόμηση
 - Τα δυνατά σημεία: σταθερός χρόνος πρόβλεψης, αναλυτική λύση εύκολη στην εφαρμογή,...
 - Αδυναμίες: δεν μπορεί να διαμορφώσει μη γραμμικές σχέσεις,...
- Αδυναμία μπορεί να αντιμετωπιστεί χρησιμοποιώντας πολυωνυμικά χαρακτηριστικά





Ταξινόμηση

- Η ταξινόμηση είναι το εποπτευόμενο μαθησιακό πρόβλημα της πρόβλεψης μιας διακριτής τιμής
- Ταξινόμηση k-κοντινότερου γείτονα
 - Πρόβλεψη είναι η πλειοψηφία των πλησιέστερων γειτόνων
 - $K=1$: τεριάζει με το θόρυβο, $K=m$ πάντα προβλέπουν την τάξη πλειοψηφίας
- Logistic Regression
 - Απλή μέθοδος για τη δυαδική ταξινόμηση
 - Τροφοδοτεί την έξοδο της γραμμικής παλινδρόμησης μέσω της σιγμοειδούς συνάρτησης που την κάνει σε $[0,1]$
 - Το αποτέλεσμα θεωρείται ως η εκτιμώμενη πιθανότητα πρόβλεψης της θετικής τάξης
 - Χρησιμοποιεί προκαθορισμένος κατώφλι (πιθανότητας) 0,5 για την τοποθέτηση του ορίου απόφασης
 - Το όριο απόφασης του logistic regression είναι γραμμικό, μπορεί να είναι μη γραμμικό εάν χρησιμοποιούμε πολυωνυμικά χαρακτηριστικά





Ταξινόμηση

- Το logistic regression βελτιστοποιεί το σφάλμα Διασταυρούμενης Εντροπίας που έχει κυρτό σχήμα
- Χρησιμοποιούμε μεθόδους που βασίζονται στο gradient για να βρούμε το ελάχιστο
- Το σφάλμα ΔΕ τιμωρεί το μοντέλο πολύ αν η προβλεπόμενη πιθανότητα του είναι πολύ μακριά από την πραγματική
- Διάνυσμα κλίσης του σφάλματος ΔΕ για logistic regression (παρόμοιο με το ΜΤΣ για τη γραμμική παλινδρόμηση)
- Ανάλυση σφαλμάτων για δυαδικούς ταξινομητές
 - Αληθινά θετικά, αληθινά αρνητικά, ψευδώς θετικά και ψευδώς αρνητικά
 - Μετρήσεις: Ακρίβεια, ακρίβεια, αναλογία ΠΘ, αναλογία ΨΑ, F1-score
 - Καμπύλη ROC: χαράσσει το αναλογία ΨΘ και το αναλογία ΠΘ για όλα τα κατώφλια ταξινόμησης
 - Βαθμολογία ΠΚΚ: μια ενιαία μέτρηση με βάση το ROC που μπορεί να χρησιμοποιηθεί για τη σύγκριση των ταξινομητών
- Ταξινόμηση πολλαπλών κλάσεων
 - One-vs-rest: εκμάθηση K δυαδικών ταξινομητών· η πρόβλεψη είναι η κλάση του πιο σίγουρου ταξινομητή
 - Softmax: χρησιμοποιεί one-hot encoding κλάσεων· πρόβλεψη είναι μια κατανομή πιθανοτήτων σε κλάσεις K





Διάλεξη 5: Αξιολόγηση και βελτίωση μοντέλου

Μαθησιακά αποτελέσματα

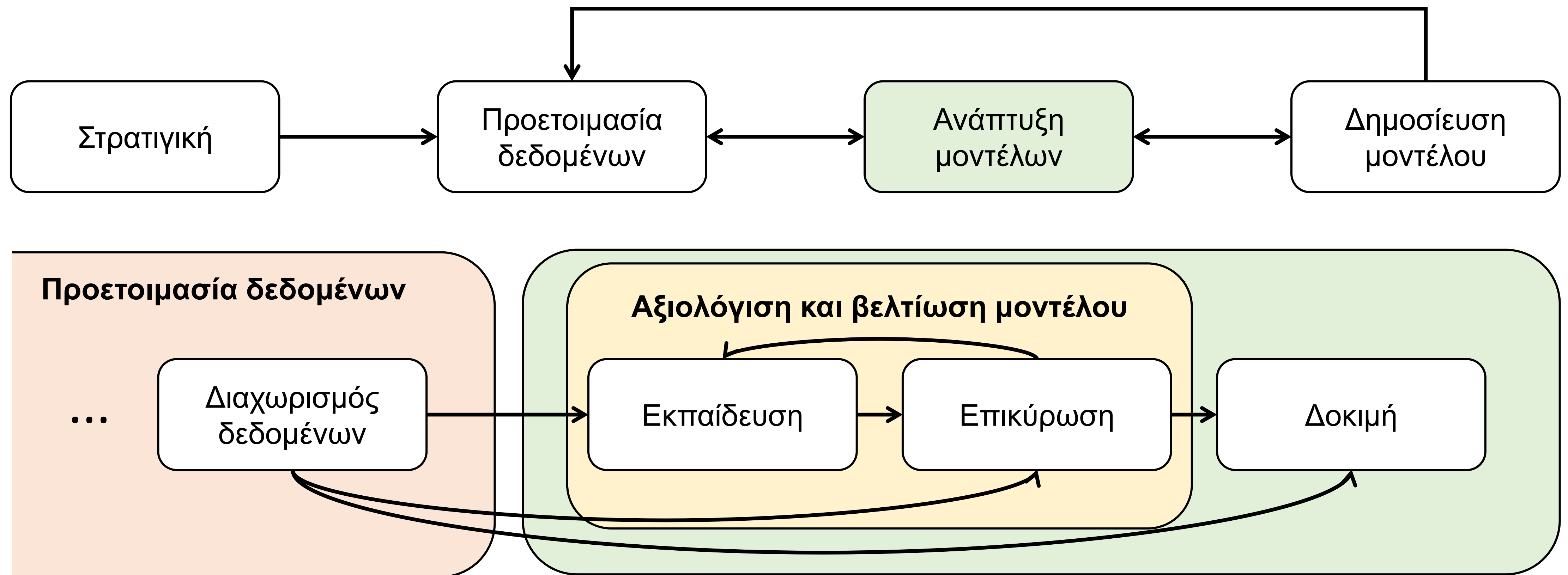
Θα κατανοήσετε:

1. το ζήτημα της γενίκευσης στη μηχανική μάθηση και πώς σχετίζεται με την αξιολόγηση μοντέλων
2. οι έννοιες της υπερπροσαρμογής, της μη προσαρμογής και της αντιστάθμισης της προκατάληψης-διακύμανσης
3. η σημασία της κατάρτισης, της επικύρωσης και των συνόλων δοκιμών
4. τι είναι η διασταυρούμενη επικύρωση k-fold και πότε να το χρησιμοποιήσετε
5. πότε να αποκτήσετε περισσότερα δεδομένα για τη βελτίωση των μοντέλων ML
6. πώς να βελτιώσετε τα μοντέλα ML χρησιμοποιώντας την τακτοποίηση L1 και L2
7. πώς να βελτιώσετε τα μοντέλα ML με το συντονισμό των υπερπαραμέτρων τους





Ανάπτυξη μοντέλων





Το ζήτημα της γενίκευσης

Τι είναι η γενίκευση;

- Δεν απομνημονεύουμε τις λεπτομέρειες των δεδομένων εκπαίδευσης μας
- Η ικανότητα του μοντέλου να έχει «καλές επιδόσεις» σε προηγουμένως άορατα δεδομένα, που προέρχονται από την ίδια κατανομή με εκείνη που χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου.

Γιατί θέλουμε γενίκευση;

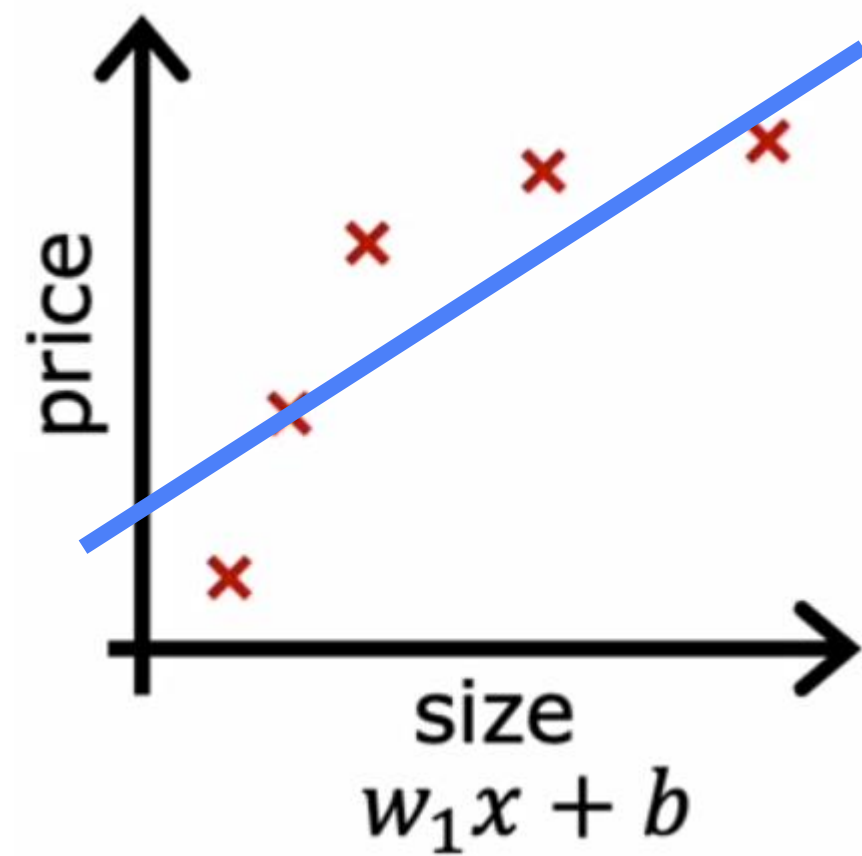
- Το σύνολο δεδομένων μας είναι ένα **δείγμα** από μια **ευρύτερη κατανομή**
- Θέλουμε να εκπαιδύσουμε ένα μοντέλο με τρόπο που να μην εστιάζει στον «θόρυβο» στα δεδομένα
- Η αγνόηση του θορύβου θα επιτρέψει στο μοντέλο να έχει προγνωστική ισχύ πάνω στην ευρύτερη κατανομή



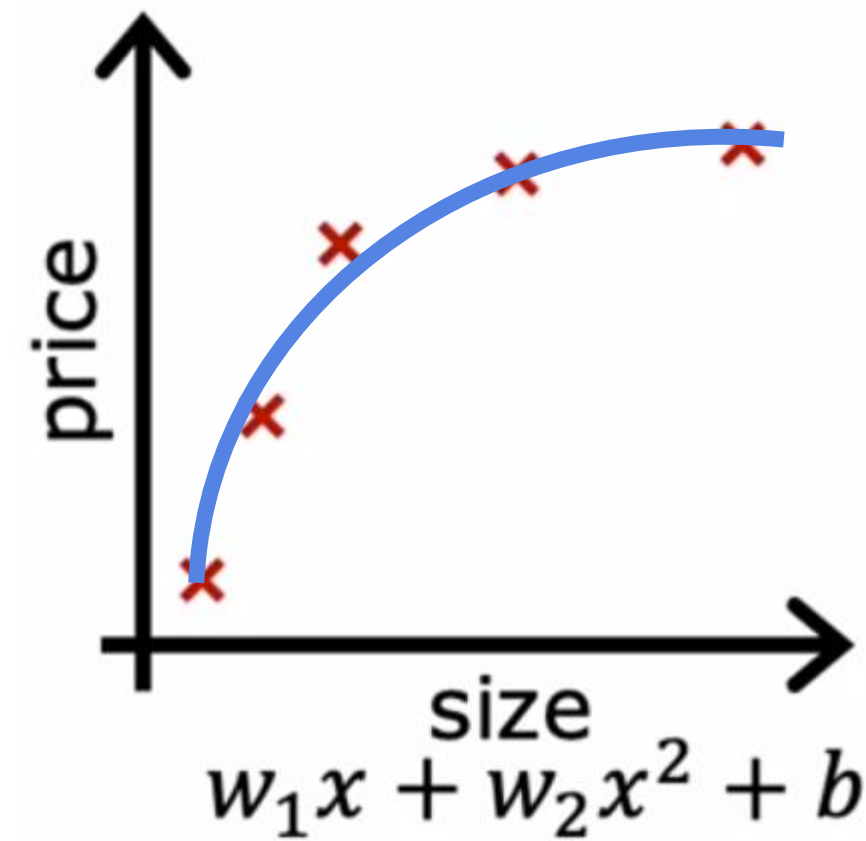


Παράδειγμα παλινδρόμησης

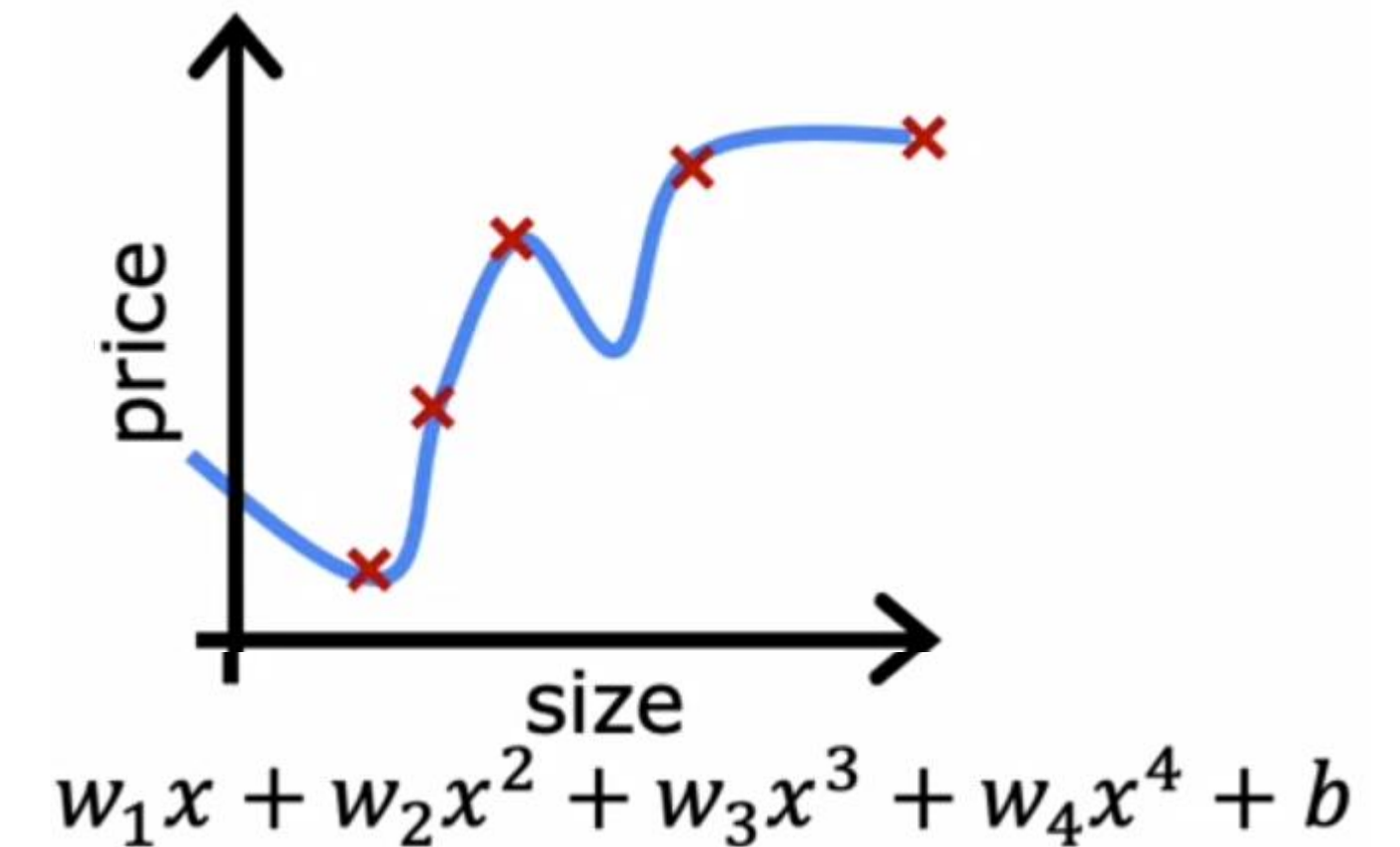
Πηγή: Andrew Ng, Μηχανική Μάθηση — Coursera



(α)



(β)



(γ)

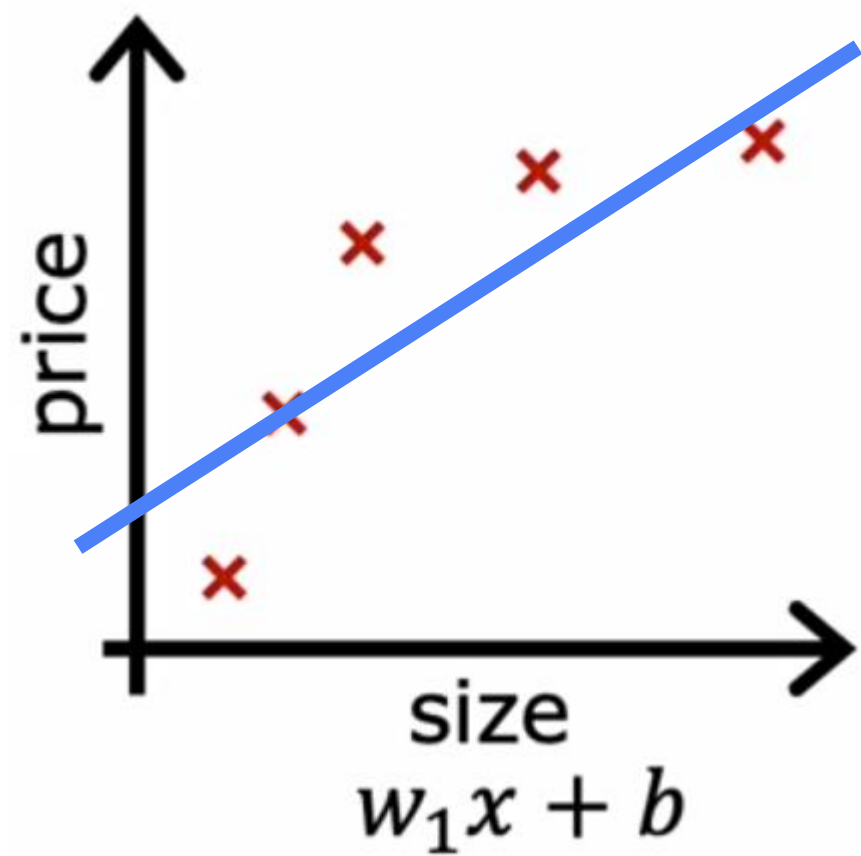
Ποιο μοντέλο είναι καλύτερο, δηλαδή παρέχει καλύτερη γενίκευση;





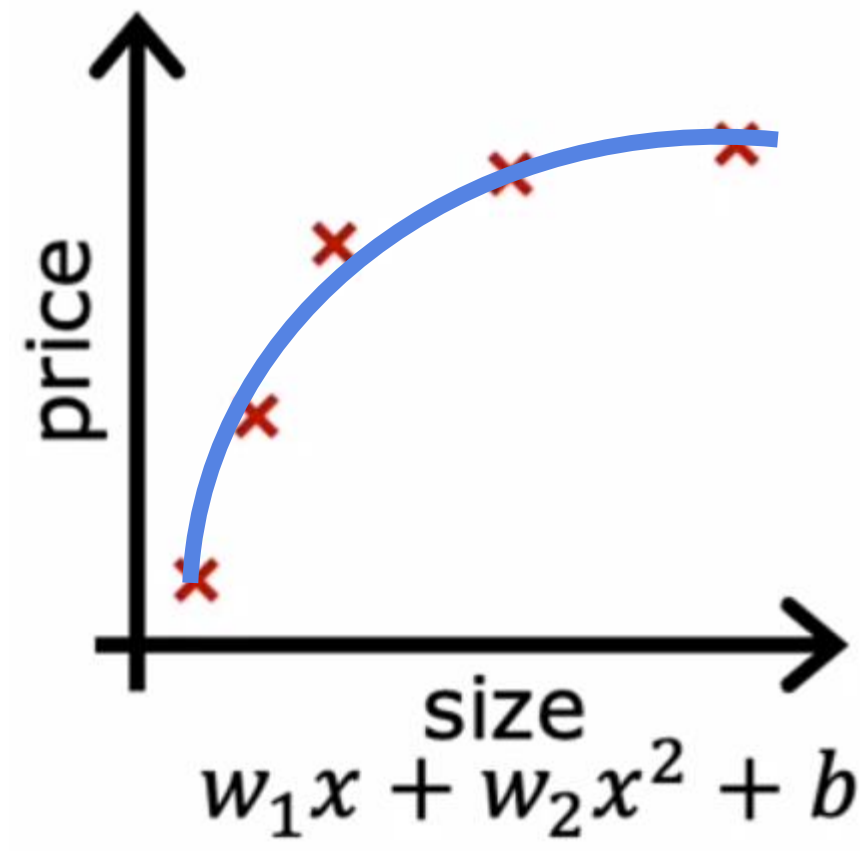
Παράδειγμα παλινδρόμησης: υποπροσαρμογή vs υπερπροσαρμογή

Πηγή: Andrew Ng, Μηχανική Μάθηση — Coursera



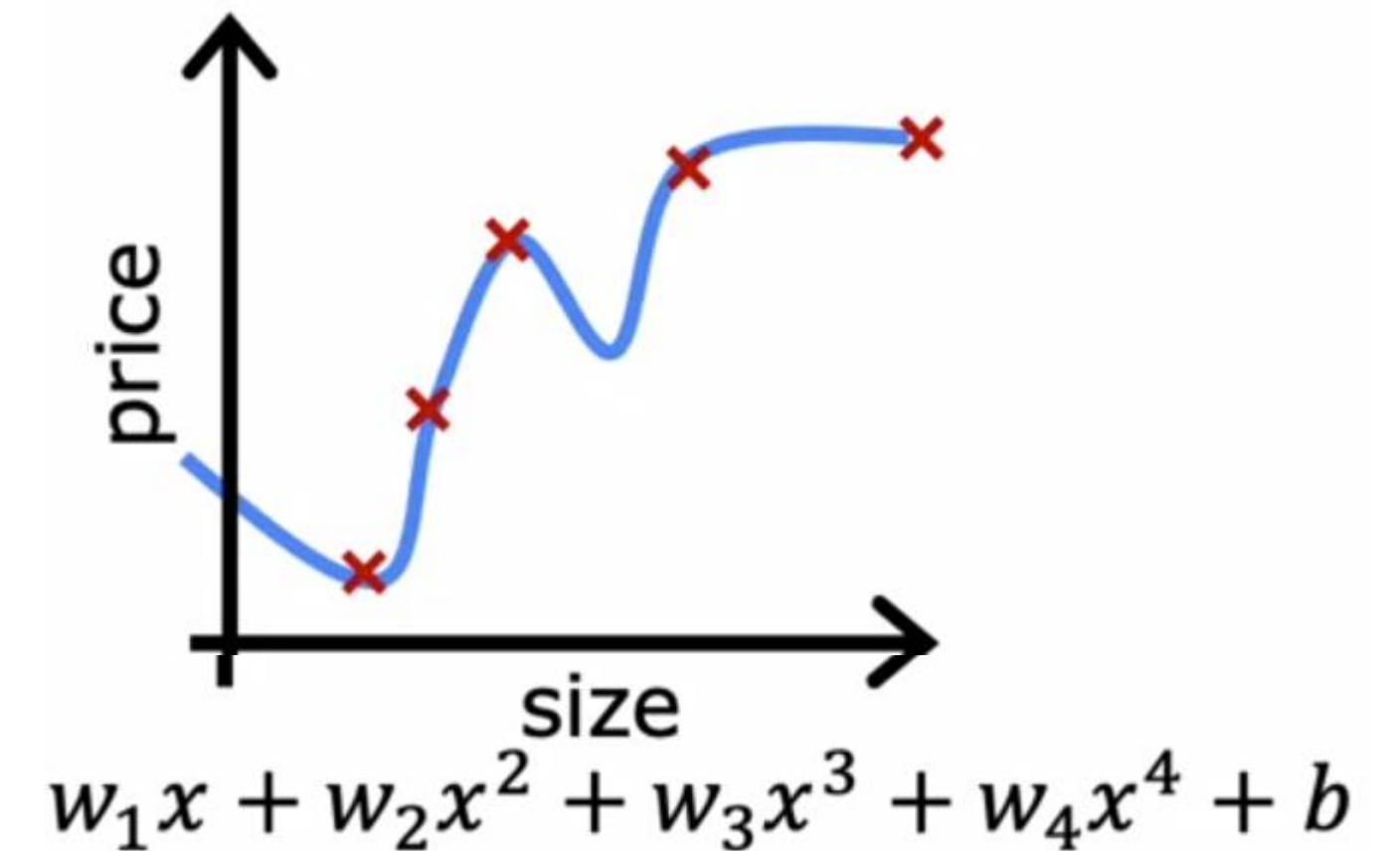
Μοντέλο πάρα πολύ απλό

Δεν ταιριάζει με τα δεδομένα εκμάθησης καλά



Μοντέλο ακριβώς σωστό

Ταιριάζει με τα δεδομένα εκμάθησης καλά



Μοντέλο πάρα πολύ περίπλοκο

Ταιριάζει με τα δεδομένα εκμάθησης υπερβολικά καλά

Υποπροσαρμογή

Γενίκευση

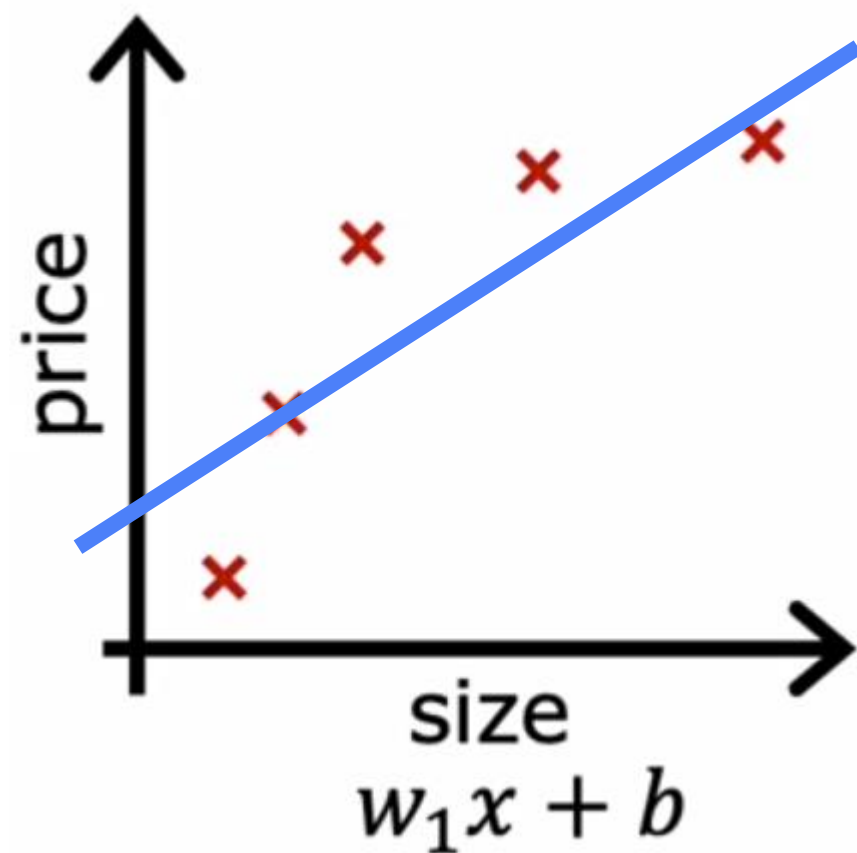
Υπερπροσαρμογή





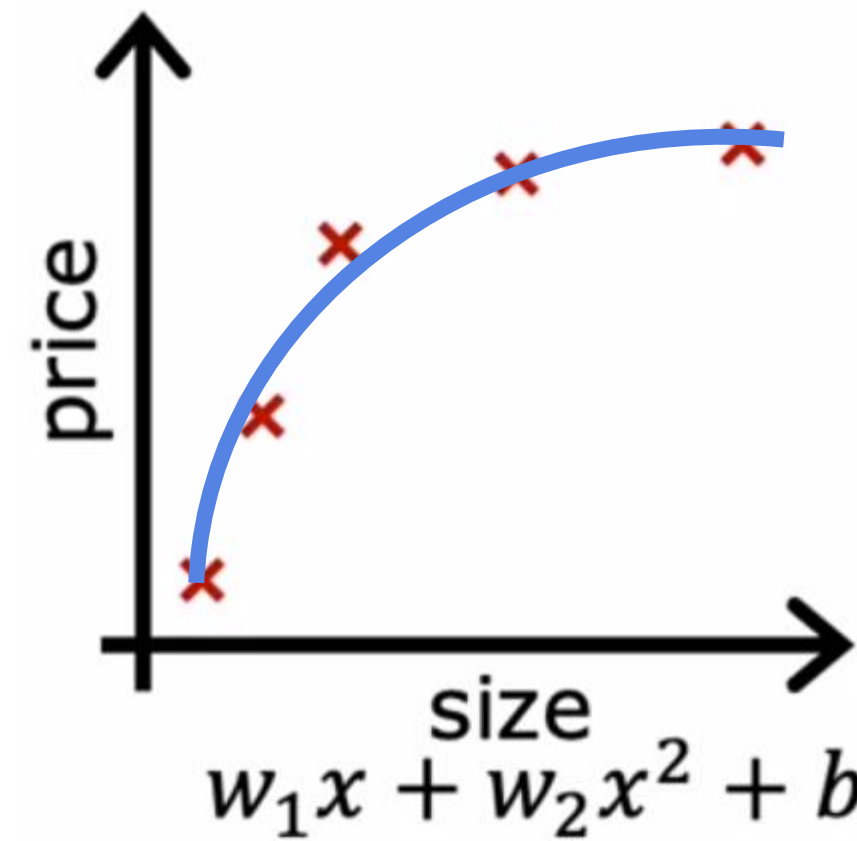
Παράδειγμα παλινδρόμησης: υποπροσαρμογή vs υπερπροσαρμογή

Πηγή: Andrew Ng, Μηχανική Μάθηση — Coursera



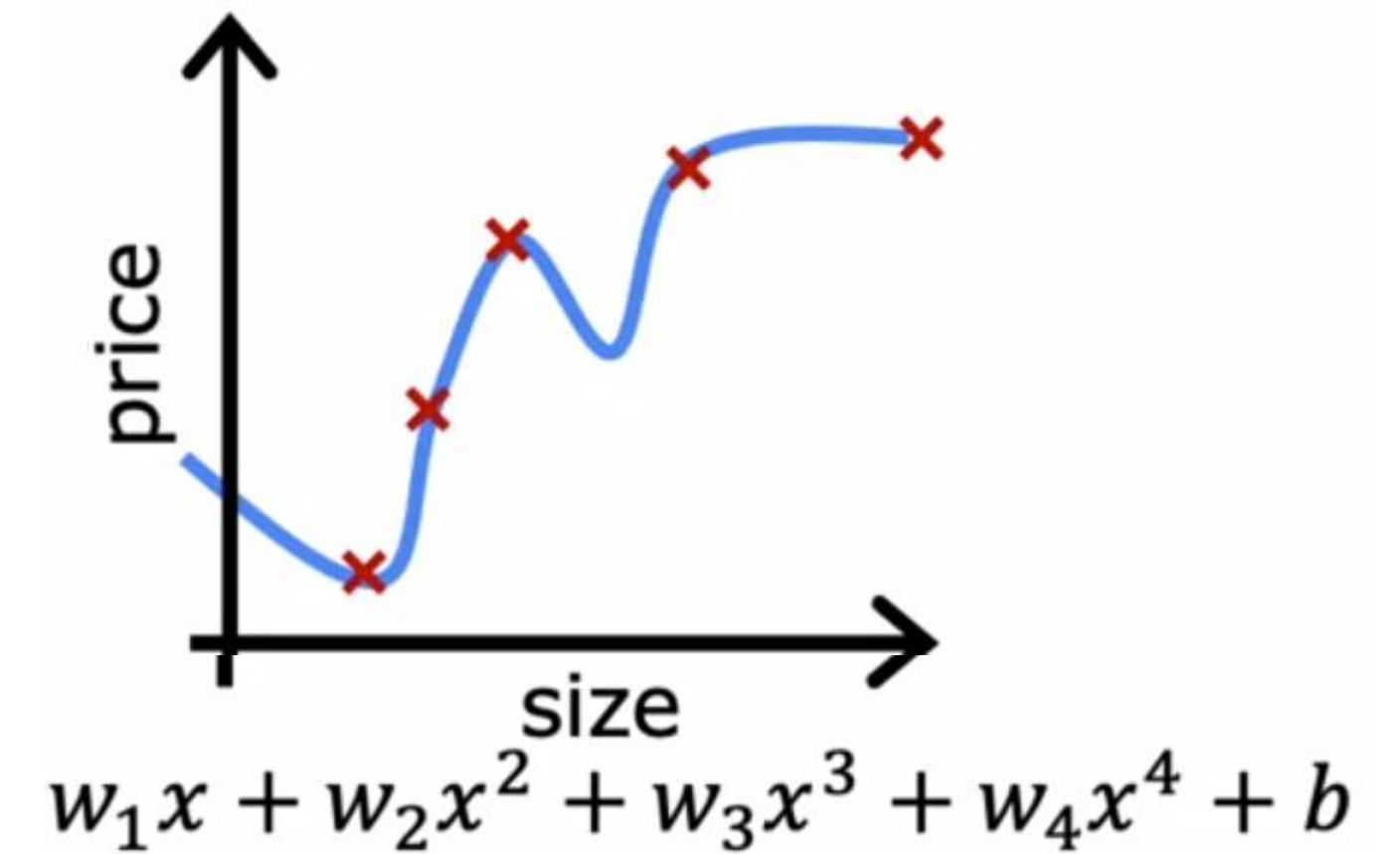
Μοντέλο πάρα πολύ απλό

Δεν ταιριάζει με τα δεδομένα εκμάθησης καλά



Μοντέλο ακριβώς δεξιά

Ταιριάζει με τα δεδομένα εκμάθησης καλά



Μοντέλο πάρα πολύ περίπλοκο

Ταιριάζει με τα δεδομένα εκμάθησης υπερβολικά καλά

Υψηλή προκατάληψη

Γενίκευση

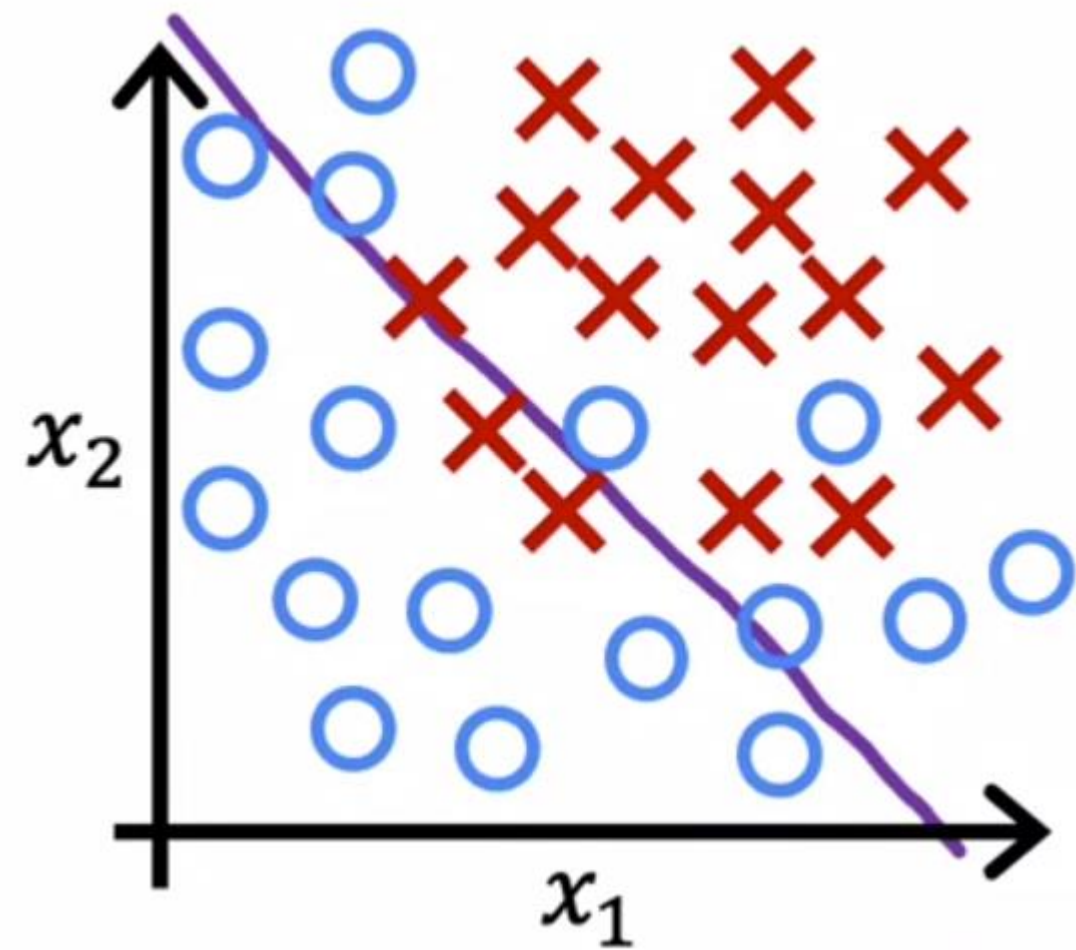
Υψηλή διακύμανση





Παράδειγμα ταξινόμησης

Πηγή: Andrew Ng, Μηχανική Μάθηση — Coursera

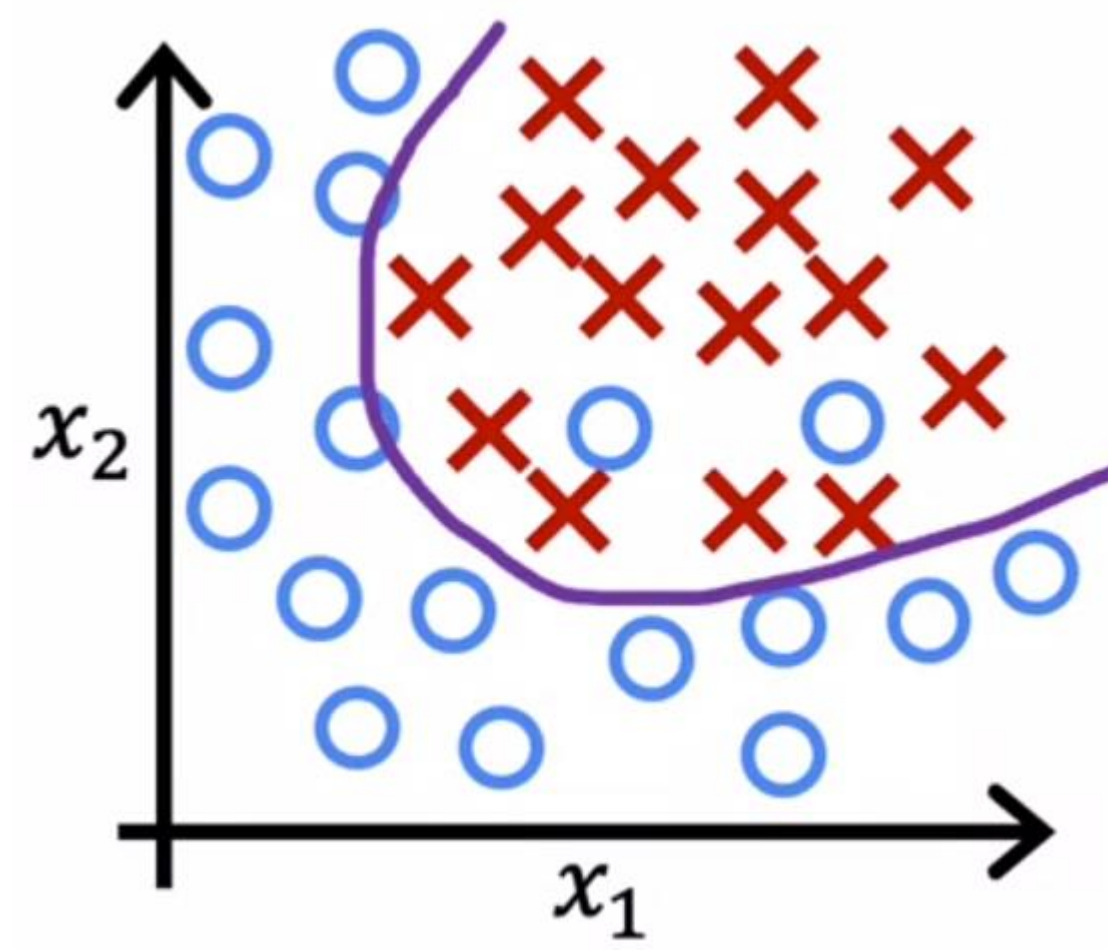


$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$f_{\theta}(x) = g(z)$$

$$g \text{ είναι η σιγμοειδής συνάρτηση}$$

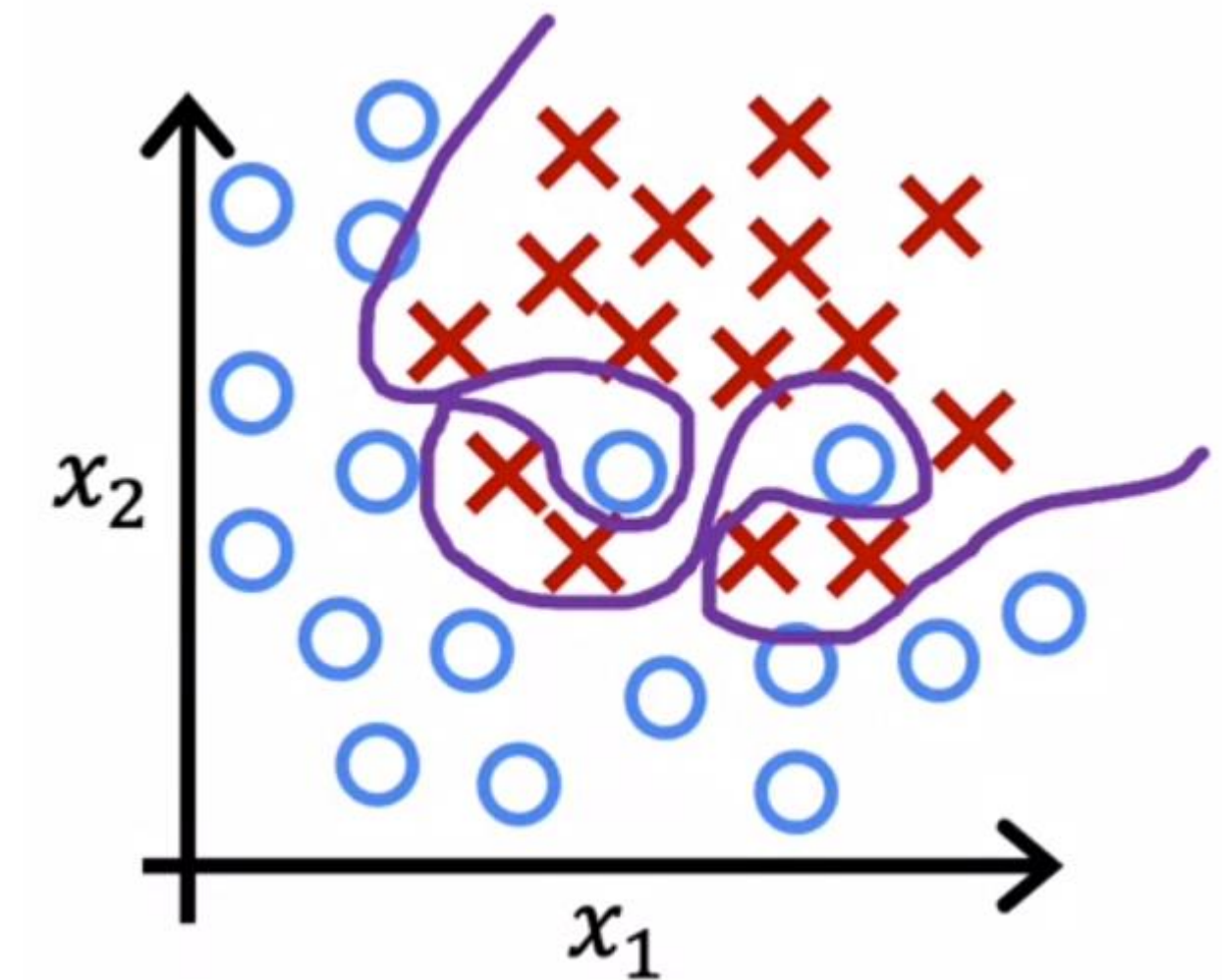
Ακατάλληλη προσαρμογή
Υψηλή προκατάληψη



$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$+ \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2$$

Γενίκευση



$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$+ \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2$$

$$+ \theta_5 x_1^2 x_2^3 + \dots$$

Υπερπροσαρμογή
Υψηλή διακύμανση





Κουίζ

Πηγή: Andrew Ng, Μηχανική Μάθηση — Coursera

Στόχος μας κατά τη δημιουργία ενός μοντέλου είναι να μπορούμε να χρησιμοποιούμε το μοντέλο για να προβλέψουμε σωστά τα αποτελέσματα για **νέα παραδείγματα**. Ένα μοντέλο που το κάνει αυτό λέγεται ότι **γενικεύει** καλά.

Όταν ένα μοντέλο ταιριάζει καλά στα δεδομένα κατάρτισης αλλά δεν λειτουργεί καλά με νέα παραδείγματα που δεν περιλαμβάνονται στο σύνολο κατάρτισης, αυτό είναι ένα παράδειγμα:

1. Υποπροσαρμογή (υψηλή προκατάληψη)
2. Υπερπροσαρμογή (υψηλή διακύμανση)
3. Ένα μοντέλο που γενικεύει καλά (ούτε υψηλή διακύμανση ούτε υψηλή προκατάληψη)





Εξισορρόπηση Προκατάληψης-διακύμανσης

Προκατάληψη: σφάλμα από λανθασμένες (απλουστεύσεις) παραδοχές στον αλγόριθμο μάθησης.

- Η υψηλή προκατάληψη μπορεί να προκαλέσει έναν αλγόριθμο να χάσει τις σχετικές σχέσεις μεταξύ των χαρακτηριστικών και των εξόδων στόχων (υποπροσαρμογή).

Διακύμανση: σφάλμα από ευαισθησία σε μικρές διακυμάνσεις στο σύνολο εκπαίδευσης.

- Υψηλή διακύμανση μπορεί να προκύψει από έναν αλγόριθμο που μοντελοποιεί τον τυχαίο θόρυβο στα δεδομένα κατάρτισης (υπερπροσαρμογή).

Εξισορρόπηση Προκατάληψη-διακύμανση: συγκρούονται στην προσπάθεια να ελαχιστοποιήσουν ταυτόχρονα αυτές τις δύο πηγές σφάλματος που εμποδίζουν τους εποπτευόμενους αλγόριθμους μάθησης να γενικεύσουν πέρα από το σύνολο της εκπαίδευσής τους.





Αποσύνθεση της Προκατάληψης-διακύμανσης του ΜΤΣ

Ας υποθέσουμε ότι υπάρχει μια συνάρτηση με θόρυβο: $y = f(x) + \varepsilon$, που $\varepsilon \sim N(0, \sigma^2)$

Θέλουμε να βρούμε μια συνάρτηση $\hat{f}_\theta(x; D)$ που προσεγγίζει την πραγματική συνάρτηση $f(x)$ όσο το δυνατόν καλύτερα χρησιμοποιώντας έναν αλγόριθμο μάθησης και ένα σύνολο εκπαίδευσης $D = \{(x^{(1)}, y^{(1)}), \dots, (x^{(m)}, y^{(m)})\}$, όπου $D \sim P(x, y)$

Μετράμε το ΜΤΣ μεταξύ y και $\hat{f}_\theta(x; D)$: θέλουμε να ελαχιστοποιήσουμε $(y - \hat{f}_\theta(x; D))^2$ για όλα τα σημεία εντός D και εκτός

Δεν μπορούμε να το κάνουμε τέλεια, διότι $y^{(i)}$ περιέχει θόρυβο, ε οπότε πρέπει να δεχτούμε κάποιο *αμετάκλητο λάθος*.

Μπορούμε να αποσυνθέσουμε το αναμενόμενο σφάλμα ενός μοντέλου \hat{f}_θ σε ένα αόρατο δείγμα x ως εξής:

$$E_{D, \varepsilon} \left[(y - \hat{f}_\theta(x; D))^2 \right] = (Bias_D[\hat{f}_\theta(x; D)])^2 + Var_D[\hat{f}_\theta(x; D)] + \sigma^2$$

που

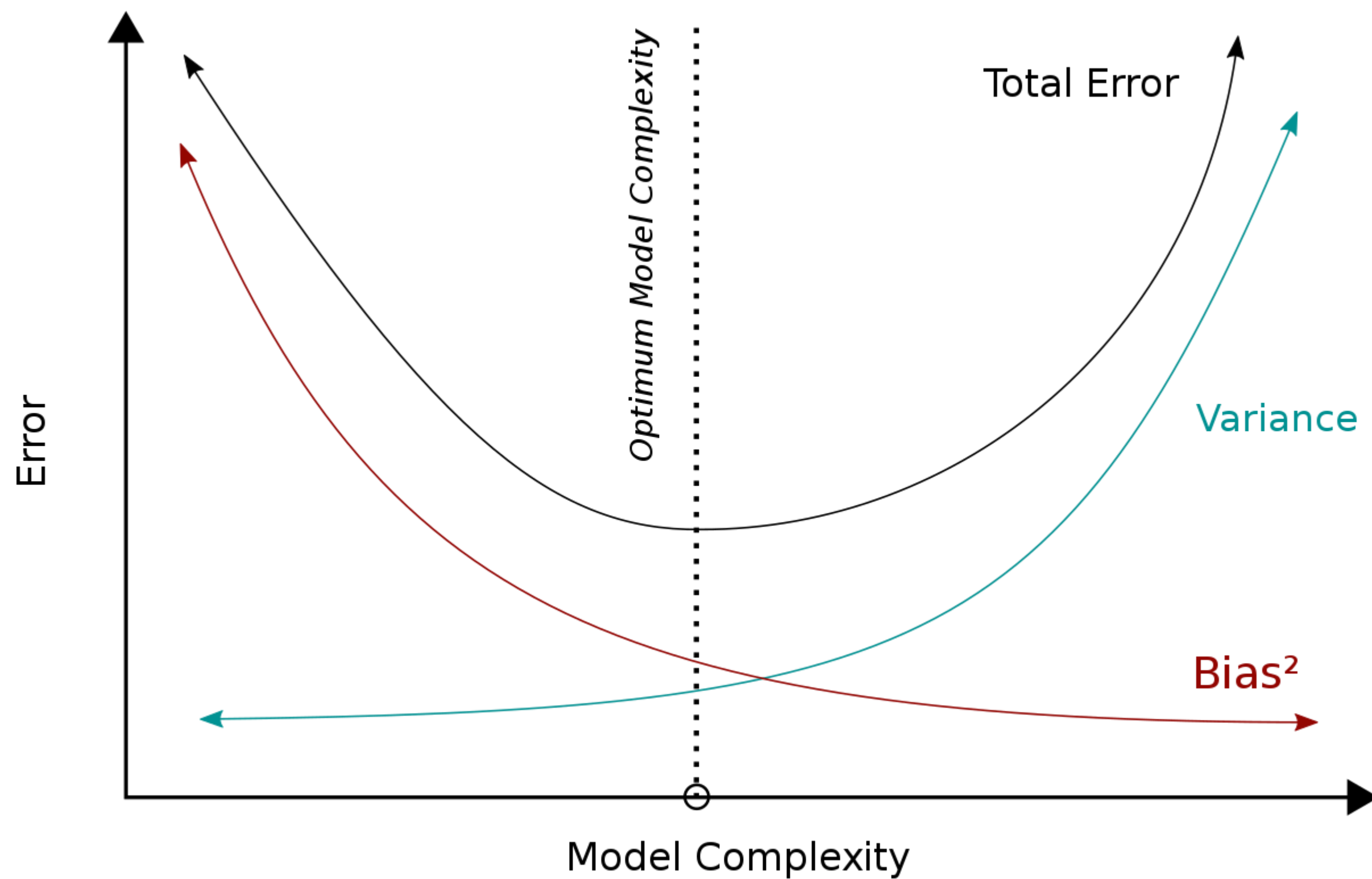
$$Bias_D[\hat{f}_\theta(x; D)] = E_D[\hat{f}_\theta(x; D)] - f(x)$$

$$Var_D[\hat{f}_\theta(x; D)] = E_D \left[(E_D[\hat{f}_\theta(x; D)] - \hat{f}_\theta(x; D))^2 \right]$$





Εξισορρόπηση Προκατάληψης-διακύμανσης



[ΠΗΓΗ](#)

Καθώς αυξάνουμε την πολυπλοκότητα του μοντέλου:

- η τετραγωνική προκατάληψη μειώνεται
- η διακύμανση αυξάνεται
- το συνολικό σφάλμα ($\text{Προκατάληψη}^2 + \text{Διακύμανση} + \epsilon$) μειώνεται μέχρι το σημείο που αρχίζει να αυξάνεται και πάλι.

Αυτό θεωρείται η βέλτιστη πολυπλοκότητα του μοντέλου σε σχέση με το μοντέλο και το σύνολο δεδομένων που έχουμε.





Εξισορρόπηση Προκατάληψης-διακύμανσης

Στην παλινδρόμηση K-κοντινότερου γείτονα:

- $K=1$: χαμηλή προκατάληψη, υψηλή διακύμανση
- $K=m$: υψηλή προκατάληψη, χαμηλή διακύμανση

Σε γραμμική παλινδρόμηση:

- λιγότερα χαρακτηριστικά: υψηλή προκατάληψη, χαμηλή διακύμανση
- πολλά χαρακτηριστικά: χαμηλή προκατάληψη, υψηλή διακύμανση





Κουίζ

1. Ας υποθέσουμε ότι έχουμε ένα σύνολο δεδομένων που περιέχει πολλά χαρακτηριστικά (στήλες). Χρησιμοποιούμε ένα μοντέλο γραμμικής παλινδρόμησης για την πρόβλεψή μας, ωστόσο, φαίνεται ότι πάσχει από *μεγάλη διακύμανση*. **Ποιες μέθοδοι προετοιμασίας δεδομένων μπορούμε να χρησιμοποιήσουμε για να μειώσουμε τη διακύμανση;**
2. Αυτό που αναφέραμε σχετικά με την αντιστάθμιση προκατάληψης-διακύμανσης του ΠΜΕ είναι *θεωρητικό*. Ο ορισμός της προκατάληψης περιλαμβάνει έναν όρο για την *πραγματική λειτουργία*, στην οποία δεν έχουμε πρόσβαση. **Τι μπορούμε να κάνουμε στην πράξη για να αντλήσουμε τη βέλτιστη πολυπλοκότητα ενός μοντέλου;**





Αξιολόγηση γενίκευσης σχετικά με το σφάλμα εκμάθησης-δοκιμής

Τι είναι η γενίκευση;

- Η ικανότητα του μοντέλου να έχει «καλές επιδόσεις» σε προηγουμένως **αόρατα δεδομένα**, που προέρχονται από την ίδια κατανομή

Υπάρχει τρόπος να μετρηθεί η γενίκευση;

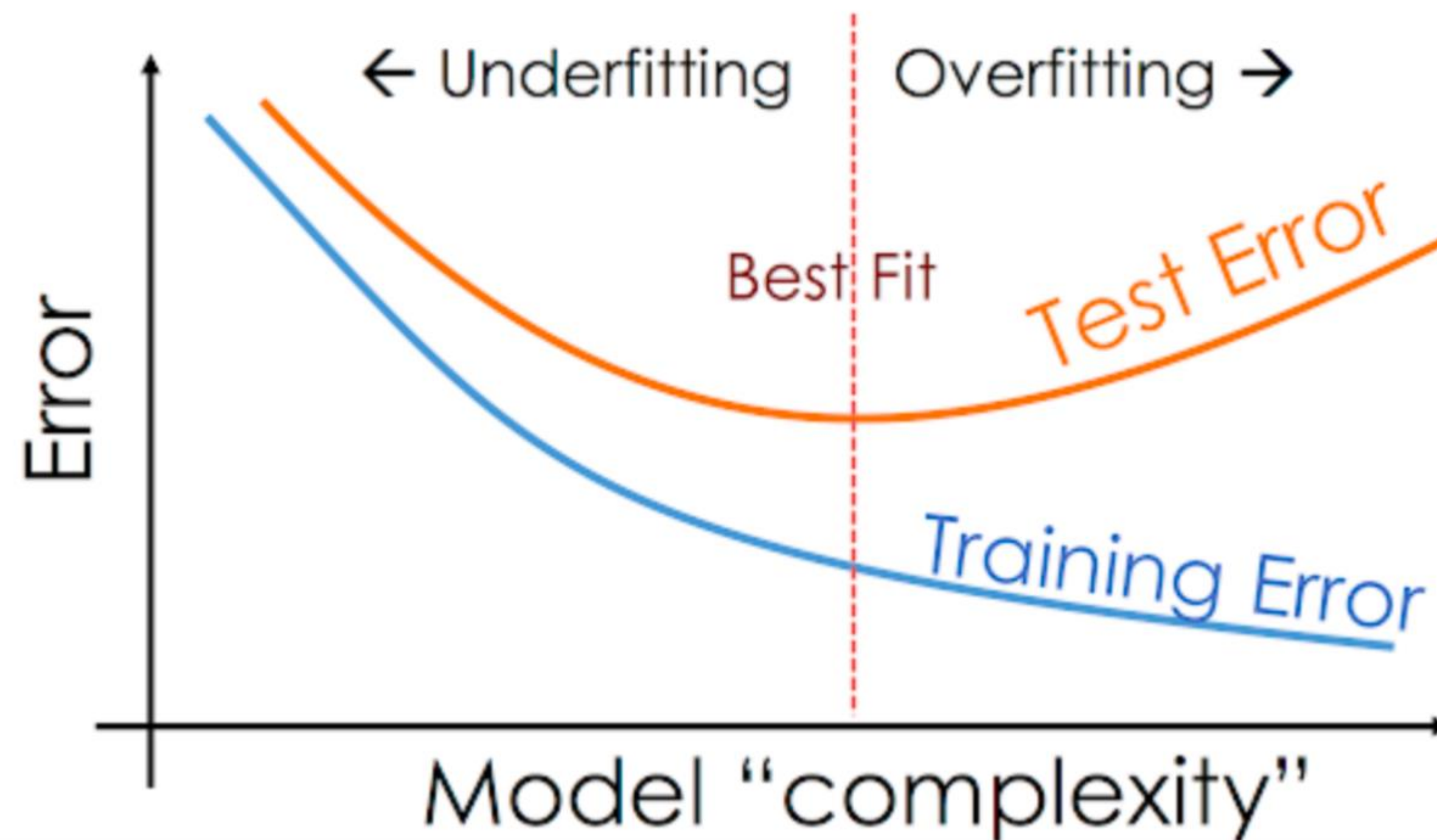
Ας το απεικονίσουμε σχεδιάζοντας τα σφάλματα εκμάθησης και δοκιμής στο **ίδιο** γράφημα!

Διαχωρισμός συνόλου δεδομένων





Αξιολόγηση γενίκευσης σχετικά με το σφάλμα εκμάθησης-δοκιμής



Αύξηση της πολυπλοκότητας του μοντέλου:

- μειώνει το σφάλμα εκμάθησης
- μειώνει το σφάλμα δοκιμής μέχρι ένα σημείο όπου αρχίζει να αυξάνεται και πάλι.

Η καλύτερη προσαρμογή είναι για την πολυπλοκότητα του μοντέλου που αντιστοιχεί στο **χαμηλότερο σφάλμα δοκιμής**

- **Επιλέγουμε το μοντέλο που έχει το χαμηλότερο σφάλμα δοκιμής**

[ΠΗΓΗ](#)





Επανεξέταση του διαχωρισμού του συνόλου δεδομένων

Μέχρι στιγμής: Διαχωρισμός συνόλου δεδομένων => σύνολο κατάρτισης + σύνολο δοκιμής

- Εκτελούμε την **επιλογή μοντέλου** στο **αόρατο** σύνολο δοκιμών
- Ωστόσο, είδαμε τα δεδομένα δοκιμής για να βρούμε τη βέλτιστη πολυπλοκότητα του μοντέλου!

Πώς γνωρίζουμε ότι το επιλεγμένο μοντέλο μας κάνει καλά σε αόρατα δείγματα;

- Το επιλεγμένο μοντέλο μπορεί να είναι το βέλτιστο για τη συγκεκριμένη επιλογή ενός σετ δοκιμών
- Πρέπει να κρύψουμε το σετ δοκιμών: μην εκτελείτε επιλογή μοντέλου σε αυτό
 - Σε διαγωνισμούς MM (π.χ. Kaggle) το σύνολο δοκιμής δεν παρέχεται





Επανεξέταση του διαχωρισμού του συνόλου δεδομένων

Αντ' αυτού, Διαχωρισμός δεδομένων:

- **Σύνολο εκμάθησης:** εκπαίδευση το μοντέλο
- **Σύνολο επικύρωσης:** χρήση για την επιλογή μοντέλου (υπερπαραμέτροι, πολυπλοκότητα, χρήση διαφορετικών μοντέλων)
- **Σύνολο δοκιμής:** απλά για να δοκιμαστεί η απόδοση — **μην** χρησιμοποιείτε για οποιεσδήποτε αποφάσεις

Στην πράξη (εκτέλεση με την ακόλουθη σειρά):

1. Διαχωρισμός συνόλου δεδομένων σε σύνολα εκμάθησης και δοκιμών
2. Σετ κατάρτισης διαίρεσης σε σύνολα εκμάθησης και επικύρωσης (ή δοκιμή σε σύνολα επικύρωσης και δοκιμής)





Γενίκευση μοντέλων



Διαχωρισμός της εκπαίδευσης-δοκιμής: π.χ. 70 %-30 %

Μην αγγίζετε το σύνολο δοκιμής κατά τη διάρκεια της εκμάθησης!

ΔΙΑΔΙΚΑΣΙΑ

1. Εκπαίδευση σε σετ κατάρτισης: $L_{\text{εκμάθησης}}(\theta)$
2. Πρόβλεψη σχετικά με το σύνολο δοκιμών: $L_{\text{δοκιμή}}(\theta)$

Επιλογή μοντέλου σε θ

Επιλογή μοντέλου σε υπερπαραμέτρους



Διαχωρισμός της εκπαίδευσης-επικύρωσης-δοκιμής: π.χ. 60 %-20 %-20 %

Μην αγγίζετε το σύνολο δοκιμής κατά τη διάρκεια της εκπαίδευσης-επικύρωσης!

ΔΙΑΔΙΚΑΣΙΑ

1. Για κάθε μοντέλο/υπερπαραμέτρο:
 - 1.1. Εκπαίδευση σε σετ κατάρτισης: $L_{\text{εκμάθησης}}(\theta)$
 - 1.2. Πρόβλεψη σχετικά με το σύνολο επικύρωσης: $L_{\text{επικύρωση}}(\theta)$

2. Επιλέξτε μοντέλα/υπερπαραμέτρα που έχουν τη χαμηλότερη $L_{\text{επικύρωση}}(\theta)$

3. Πρόβλεψη σχετικά με το σύνολο δοκιμών: $L_{\text{δοκιμή}}(\theta)$





Παράδειγμα

```
In [1]: import numpy as np
...: from sklearn.model_selection import train_test_split
...:
...: X, y = np.arange(20).reshape((10, 2)), range(10)
...:
...: train_ratio=0.6
...: val_ratio=0.2
...: test_ratio=0.2
...:
...: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=1-train_ratio, shuffle=False)
...: X_train
Out[1]:
array([[ 0,  1],
       [ 2,  3],
       [ 4,  5],
       [ 6,  7],
       [ 8,  9],
       [10, 11]])

In [2]: np.shape(X_test)
Out[2]: (4, 2)

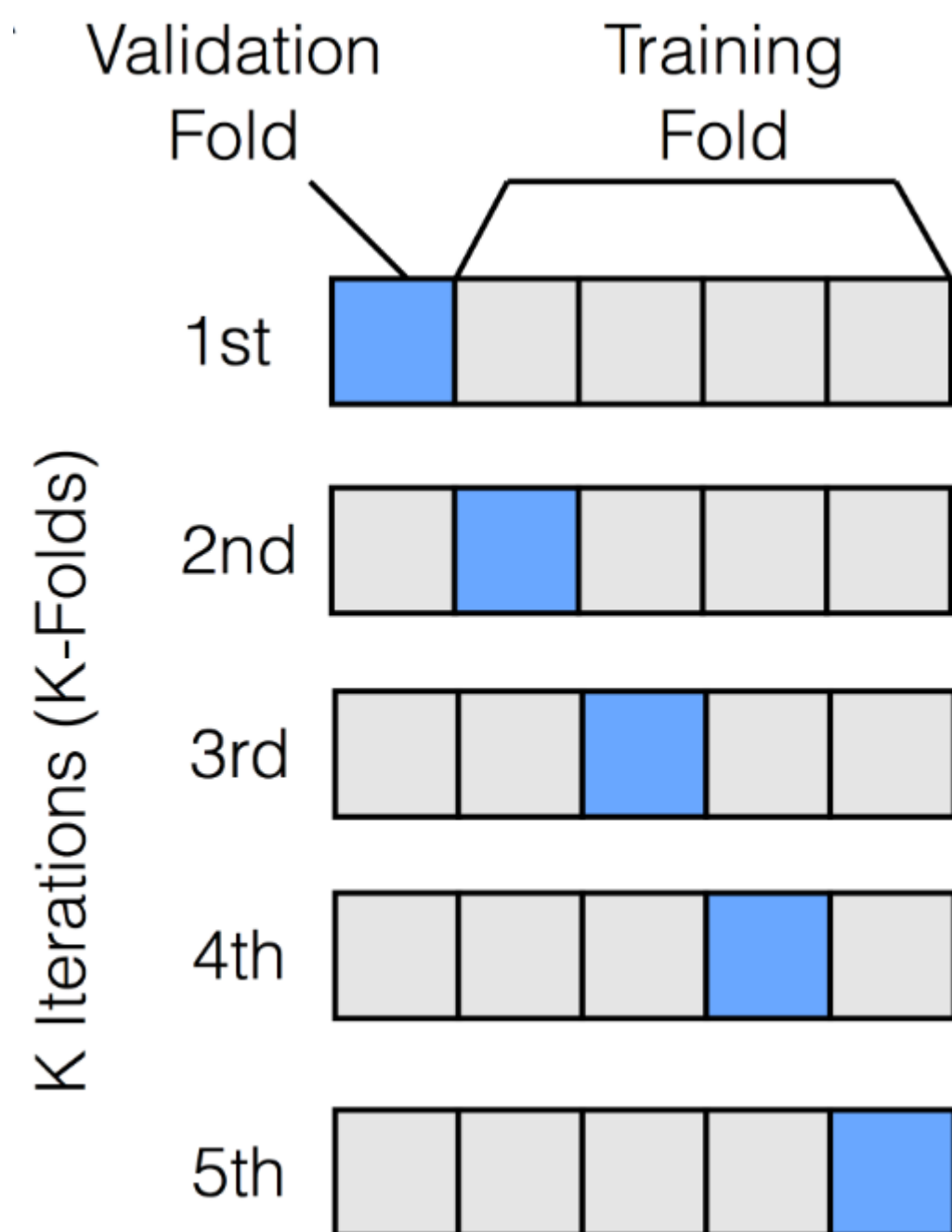
In [3]: X_val, X_test, y_val, y_test = train_test_split(X_test, y_test, test_size=test_ratio/(test_ratio + val_ratio), shuffle=False)
...: np.shape(X_test)
Out[3]: (2, 2)
```





K-fold Cross-Validation

Χρησιμοποιείται όταν δεν έχουμε πολλά σημεία δεδομένων στα σύνολα εκπαίδευσης + επικύρωσης



ΔΙΑΔΙΚΑΣΙΑ:

1. Διαχωρισμός συνόλων εκμάθησης+επικύρωσης σε k folds
2. για $i = 1$ έως k
 - 2.1. Χρησιμοποιήστε το fold i ως σύνολο επικύρωσης, παραμένοντας ως σύνολο εκπαίδευσης
 - 2.2. Εκμάθηση σε σύνολο εκμάθησης, επικύρωση σε σύνολο επικύρωσης
3. Αναφορά του μέσου σφάλματος επικύρωσης στις πτυχές επικύρωσης k

[ΠΗΓΗ](#)





K-fold Cross-Validation

Ποιο είναι το μειονέκτημα αυτής της προσέγγισης;

- Υπολογιστικά ακριβά

Ποια είναι η ακραία/πλήρης μορφή του K-fold Cross-Validation;

- Leave-one-out cross-validation (μοντέλα m που έχουν εκπαιδευτεί και επικυρωθεί)

Επιλογή K:

- Ανάγκη εξισορρόπησης του μεγέθους του συνόλου δεδομένων, της υπολογιστικής ικανότητας, του χρόνου...
- Τυπικά: $K=5$, $K=10$





Παράδειγμα

```
In [1]: import numpy as np
....: from sklearn.model_selection import KFold
....:
....: X, y = np.arange(40).reshape((20, 2)), range(20)
....:
....: K = 5
....:
....: kf = KFold(n_splits=K)
....: kf.get_n_splits(X)
Out[1]: 5

In [2]: for train_index, val_index in kf.split(X):
....:     print("TRAIN:", train_index, "VAL:", val_index)
....:
TRAIN: [ 4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19] VAL: [0 1 2 3]
TRAIN: [ 0  1  2  3  8  9 10 11 12 13 14 15 16 17 18 19] VAL: [4 5 6 7]
TRAIN: [ 0  1  2  3  4  5  6  7 12 13 14 15 16 17 18 19] VAL: [ 8  9 10 11]
TRAIN: [ 0  1  2  3  4  5  6  7  8  9 10 11 16 17 18 19] VAL: [12 13 14 15]
TRAIN: [ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15] VAL: [16 17 18 19]
```





Πότε να αποκτήσετε περισσότερα δεδομένα

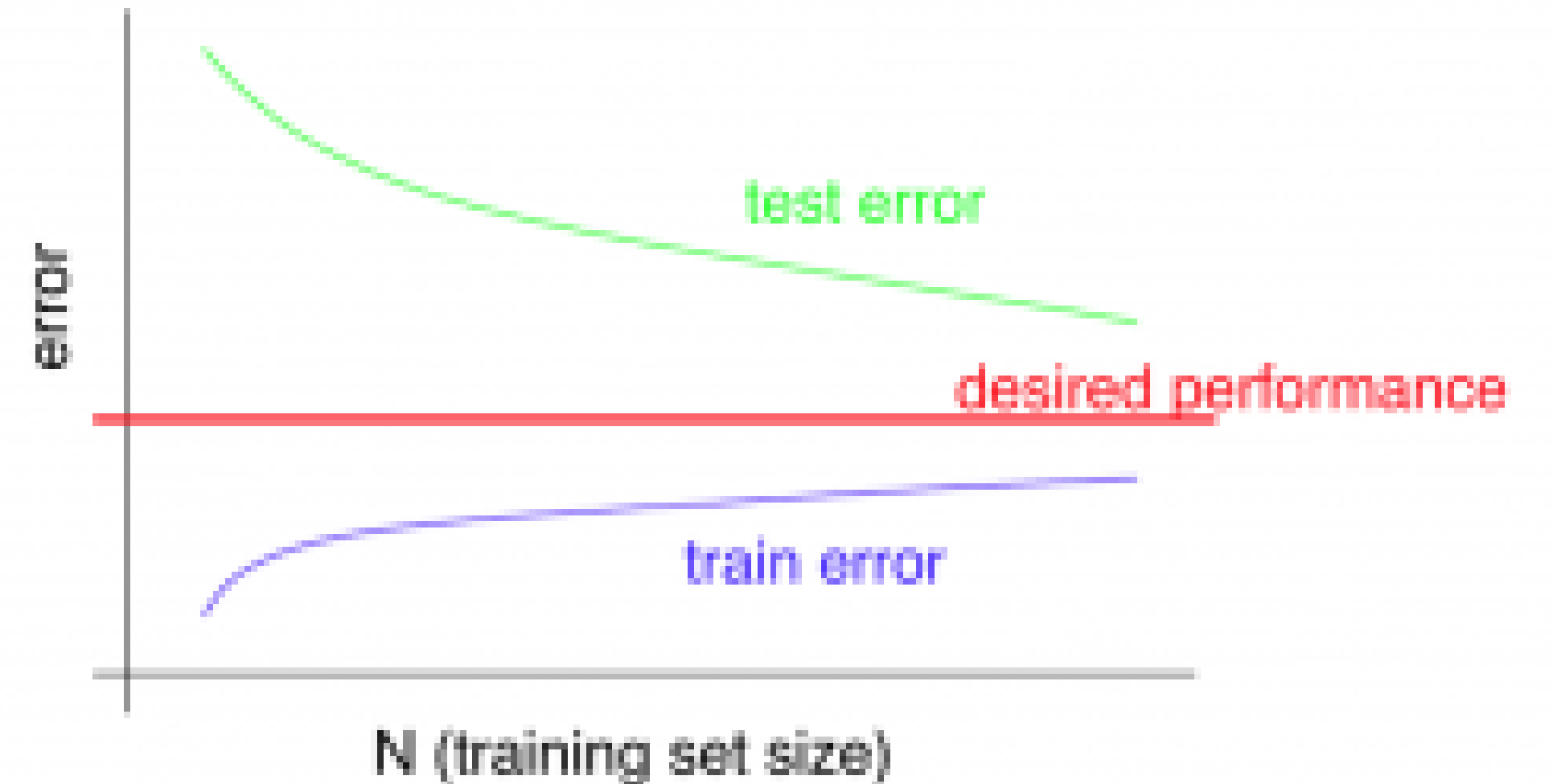
Πηγή: Andrew Ng, Μηχανική Μάθηση — Coursera



Μοντέλα Υψηλής προκατάληψης

π.χ. απλή γραμμική παλινδρόμηση

Η απόκτηση περισσότερων δεδομένων πιθανότατα δεν θα βοηθήσει.



Μοντέλα υψηλής διακύμανσης

π.χ. πολυωνυμικά χαρακτηριστικά, νευρωνικό δίκτυο

Η απόκτηση περισσότερων δεδομένων πιθανότατα θα βοηθήσει.





Κουίζ

1. Η ύπαρξη περισσότερων δεδομένων βοηθά όλα τα μοντέλα μηχανικής μάθησης εξίσου.
Σωστό ή Λάθος;

2. Τα μοντέλα με υψηλή πολυπλοκότητα υποφέρουν από υψηλή διακύμανση. Η απόκτηση περισσότερων δεδομένων είναι μια μέθοδος για τη μείωση της διακύμανσης σε τέτοια μοντέλα. Σωστό ή Λάθος;





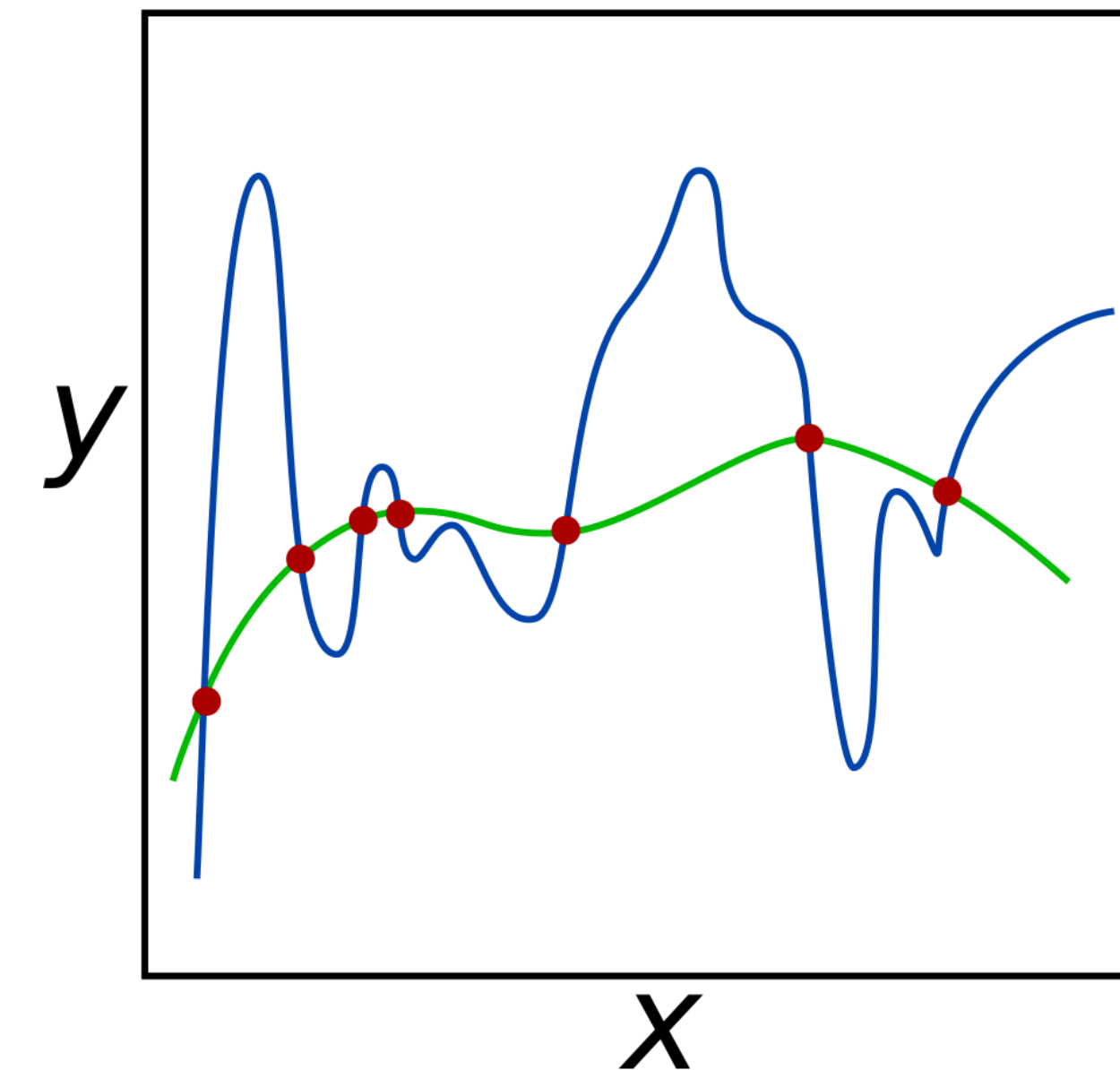
Βελτίωση μοντέλων μέσω της κανονικοποίησης

Τα σύνθετα μοντέλα έχουν το **πρόβλημα της υπερπροσαρμογής**, δηλ., υψηλή διακύμανση (προσαρμόζεται στο θόρυβο)

- Υπάρχουν πολλά πολύπλοκα μοντέλα
- **Πώς να επιλέξετε μεταξύ τους;**

Η λύση:

- Τα απλούστερα μοντέλα είναι πιο πιθανό να γενικεύσουν («το ξυράφι του Όκαμ»)
- Μειώστε τη διακύμανση απλοποιώντας το μοντέλο



Τόσο οι **μπλε** όσο και οι **πράσινες** γραμμές έχουν 0 σφάλμα κατάρτισης





Βελτίωση μοντέλων μέσω της κανονικοποίησης

Κανονικοποίηση:

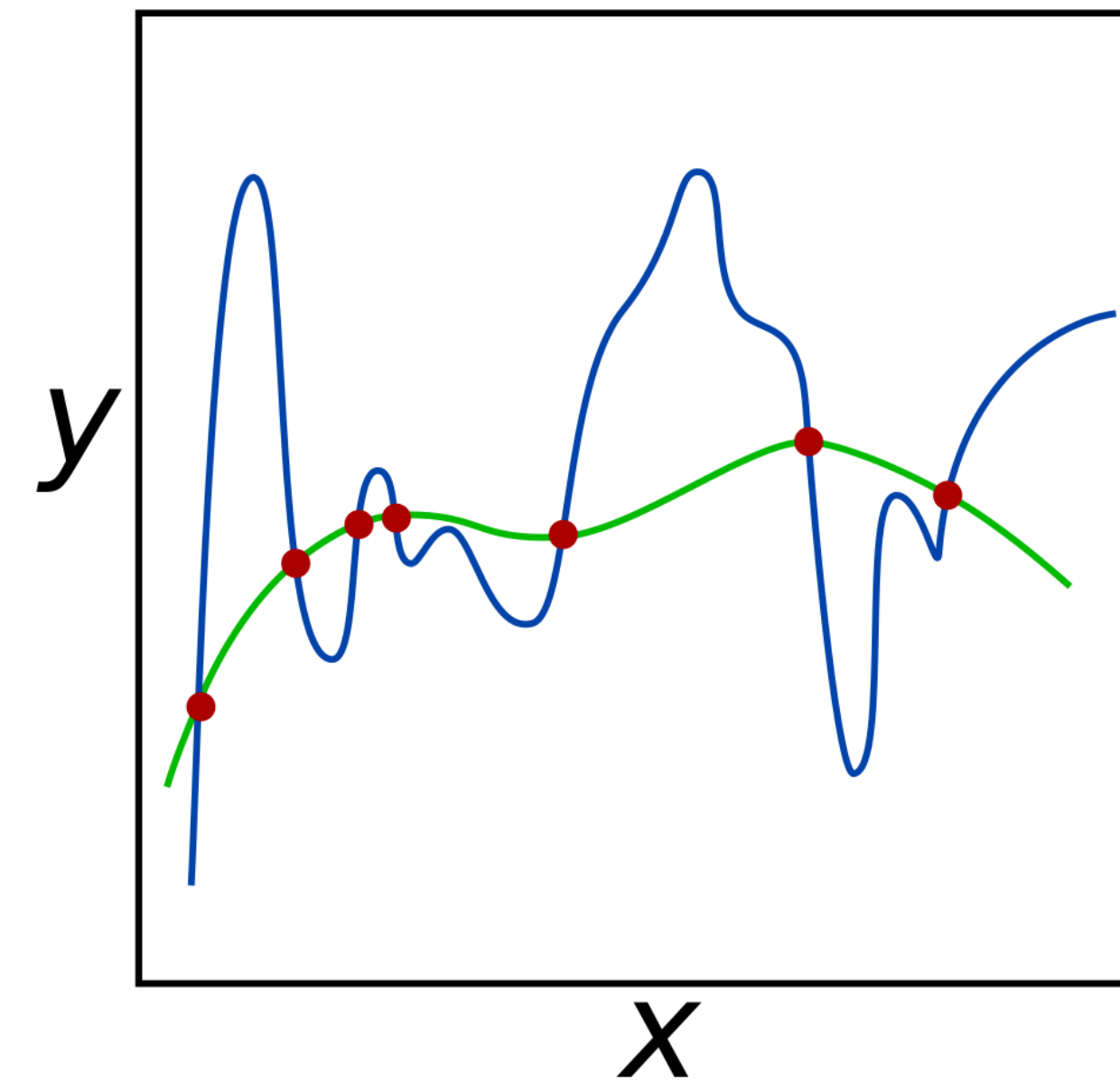
- τρόπος μείωσης της συμβολής ορισμένων χαρακτηριστικών
- περιλαμβάνει τροποποίηση της συνάρτησης απώλειας $L(\theta)$:

$$\text{νέο } L(\theta) = L(\theta) + \lambda * R(\theta)$$

όπου:

$R(\theta)$: **ποινή** για την **πολυπλοκότητα** του μοντέλου

λ : παράμετρος ελέγχου της σημασίας της τακτοποίησης



Σωστά προσαρμοσμένο λ θα επέλεγε το απλούστερο μοντέλο (**πράσινη γραμμή**)





Κανονικοποίηση L1 και L2

Κανονικοποίηση L1:

— $R(\theta) = \|\theta\|_1 = \sum_{i=1}^n |\theta_i|$

— τείνει να συρρικνώνει τις παραμέτρους στο μηδέν (σοβαρότητα)

— χρήσιμο για την **επιλογή χαρακτηριστικών**

— ονομάζεται **παλινδρόμηση Lasso** κατά την επέκταση της γραμμικής παλινδρόμησης

Κανονικοποίηση L2:

— $R(\theta) = \|\theta\|_2^2 = \sum_{i=1}^n \theta_i^2$

— τείνει να συρρικνώνει τους συντελεστές ομοιόμορφα

— χρήσιμο όταν έχουμε συνεξαρτημένα χαρακτηριστικά (π.χ. «φύλο» και «έγκυος»)

— ονομάζεται **Ridge παλινδρόμηση** κατά την επέκταση της γραμμικής παλινδρόμησης

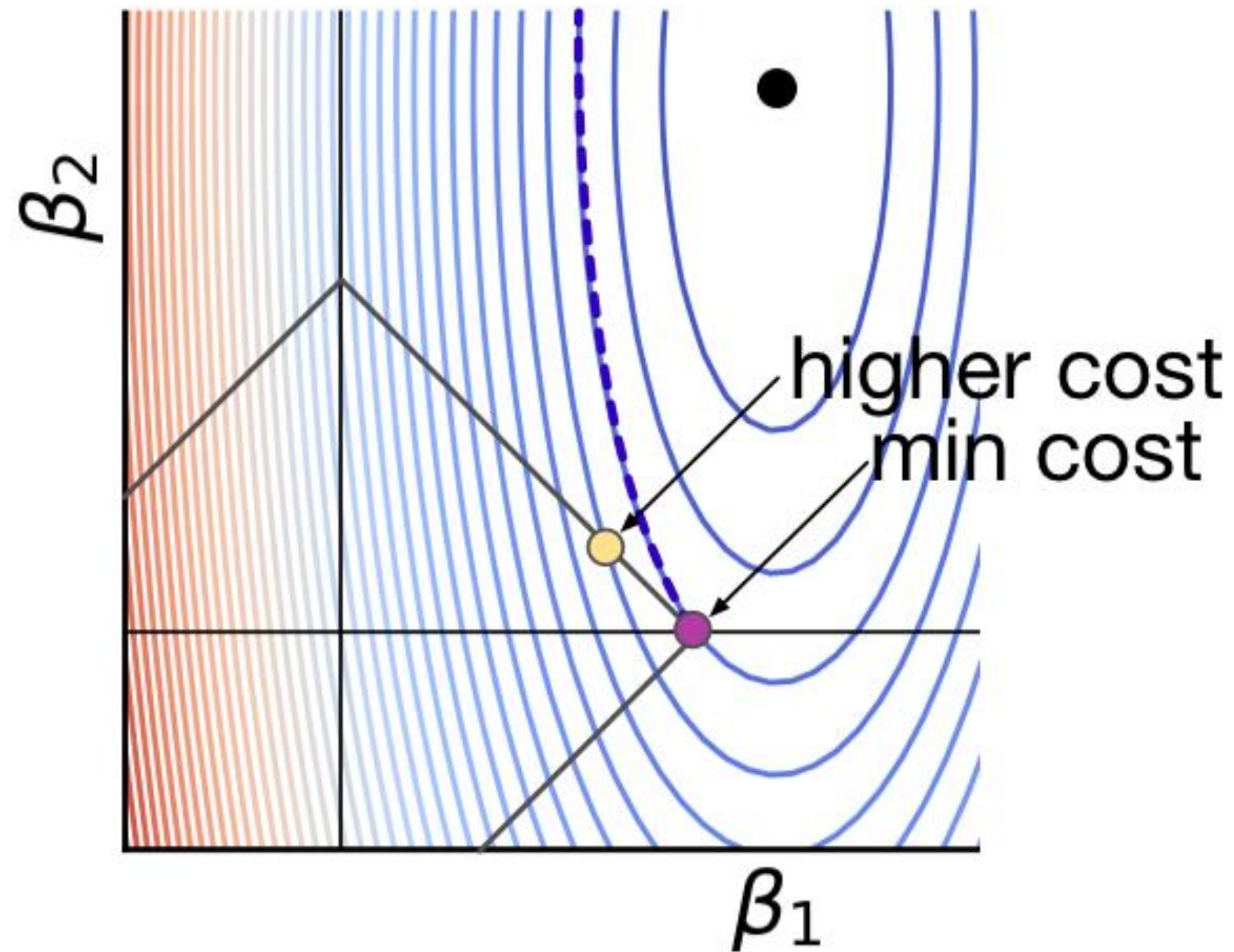
Και τα δύο μπορούν να χρησιμοποιηθούν με οποιοδήποτε μοντέλο



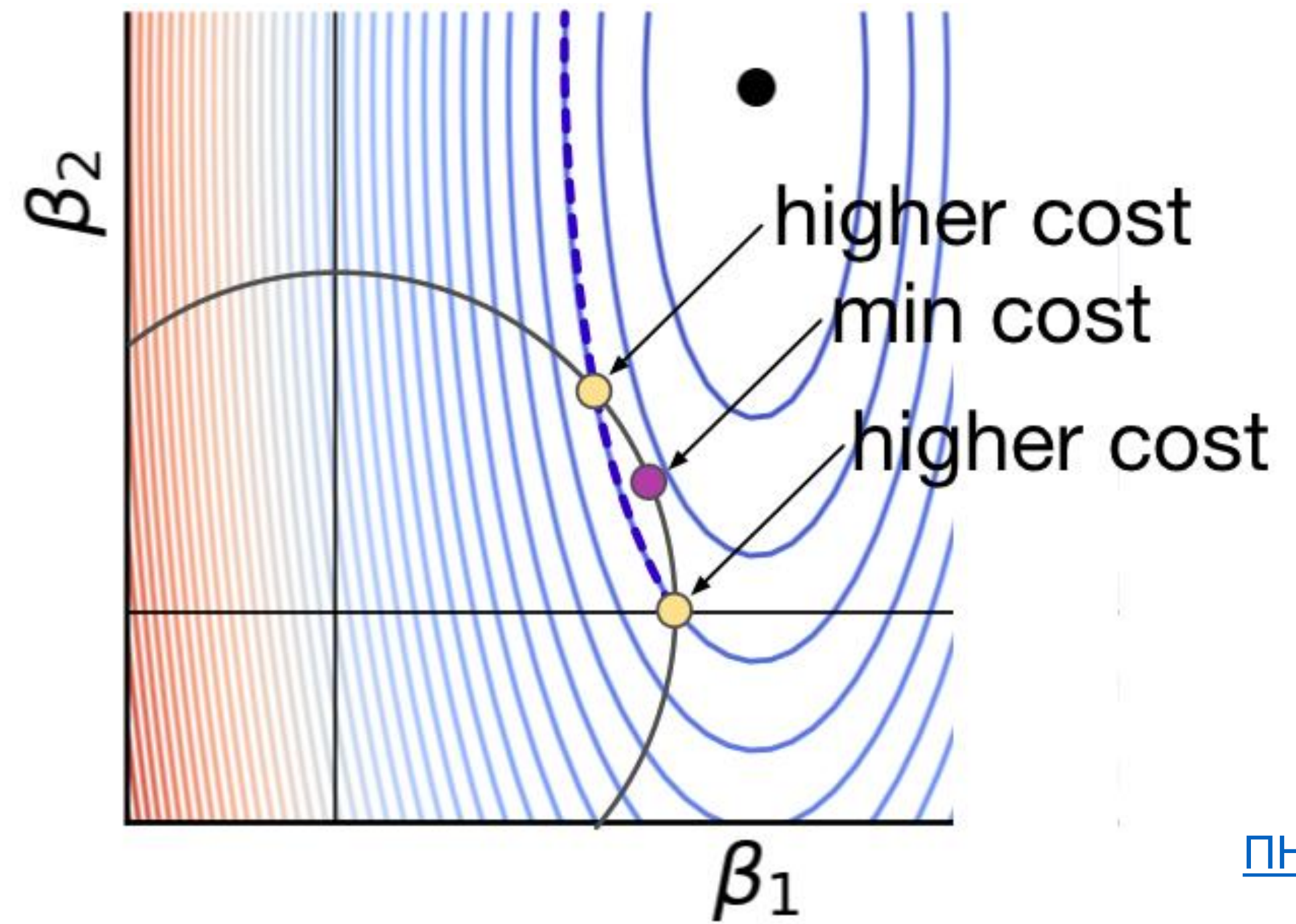


Γιατί το L1 είναι πιο πιθανό σε μηδενικούς συντελεστές από το L2;

(α) L1 διαμάντι περιορισμού



(β) L2 κύκλος περιορισμού



[ΠΗΓΗ](#)





Κανονικοποίηση L1 vs L2

Κανονικοποίηση L1:

- Τείνει να παράγει λύσεις που έχουν:
 - λίγα μεγάλα λάθη και
 - πολλά ασήμαντα λάθη
- Η κατανομή των λαθών θα είναι πολύ «χαλαρή»
- Πιο εύρωστο στις ακραίες τιμές

Κανονικοποίηση L2:

- Τείνει να παράγει λύσεις που έχουν:
 - πολύ λίγα μεγάλα λάθη (που τιμωρούνται πολύ περισσότερο)
 - πολλά μικρά λάθη που εξακολουθούν να είναι σημαντικά
- Η κατανομή των σφαλμάτων θα είναι περισσότερο «ισορροπημένη»
- Παράγει μια καλύτερη «προσαρμογή»

[ΠΗΓΗ](#)





Παράδειγμα

```
In [1]: from sklearn import linear_model
...: reg = linear_model.LinearRegression()
...: reg.fit([[0, 0], [1, 1], [2, 2]], [0, 1, 2])
Out[1]:
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
                 normalize=False)

In [2]: from sklearn import linear_model
...: reg = linear_model.Ridge(alpha=.5)
...: reg.fit([[0, 0], [0, 0], [1, 1]], [0, .1, 1])
Out[2]:
Ridge(alpha=0.5, copy_X=True, fit_intercept=True, max_iter=None,
       normalize=False, random_state=None, solver='auto', tol=0.001)

In [3]: from sklearn import linear_model
...: reg = linear_model.Lasso(alpha=0.1)
...: reg.fit([[0, 0], [1, 1]], [0, 1])
Out[3]:
Lasso(alpha=0.1, copy_X=True, fit_intercept=True, max_iter=1000,
       normalize=False, positive=False, precompute=False, random_state=None,
       selection='cyclic', tol=0.0001, warm_start=False)
```





Βελτίωση μοντέλου μέσω ρυθμίσεων υπερπαραμέτρων

Τα μοντέλα έχουν υπερπαραμέτρους:

- Παλινδρόμηση k-κοντινότερου γείτονα: k
- πολυωνυμική παλινδρόμηση: πολυωνυμικός βαθμός

Οι αλγόριθμοι μάθησης έχουν υπερπαραμέτρους:

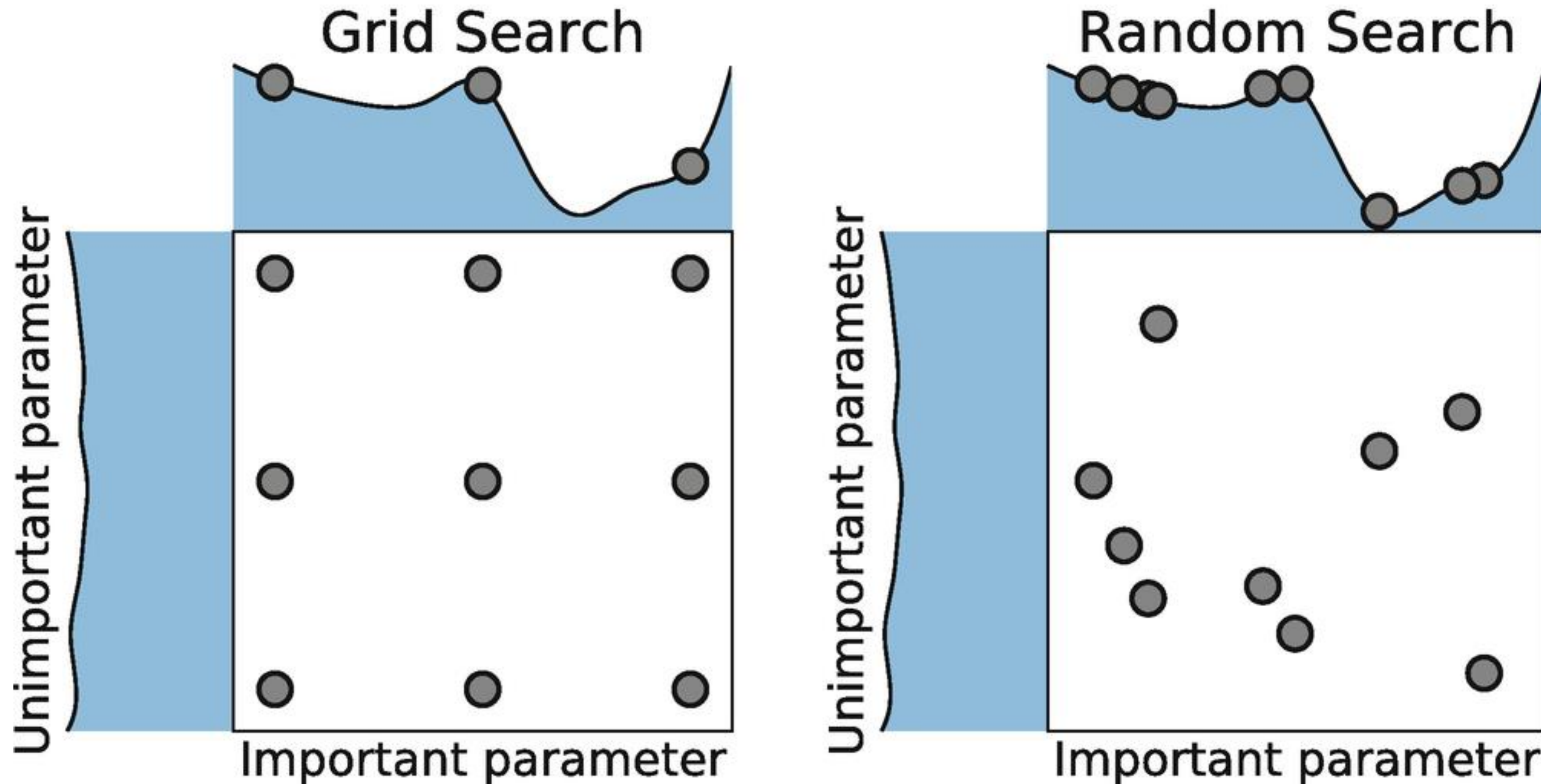
- στοχαστική κάθοδο κλίσης: ρυθμός μάθησης, ορμή
- συντελεστής τακτοποίησης στη λειτουργία απώλειας

- Οι ρυθμίσεις υπερπαραμέτρων είναι **σημαντικές** για την απόδοση ενός αλγορίθμου μάθησης - συνδυασμού μοντέλου
- Συντονιστείτε υπερπαραμέτροι χρησιμοποιώντας διασταυρούμενη επικύρωση
- Δύο απλές μέθοδοι:
 - **Αναζήτηση πλέγματος**: καθορίστε ένα σύνολο τιμών για κάθε υπερπαραμέτρο και δοκιμάστε όλους τους συνδυασμούς τιμών
 - **Τυχαία αναζήτηση**: τυχαία δειγματοληψία υπερπαραμέτρων από καθορισμένες περιοχές





Βελτίωση μοντέλου μέσω ρυθμίσεων υπερπαραμέτρων





Υπερπαραμετρική ρύθμιση σε scikit-learn

Αναζήτηση πλέγματος: [sklearn.model_selection.GridSearchCV](#)

Τυχαία αναζήτηση: [sklearn.model_selection.RandomizedSearchCV](#)





Βελτίωση μοντέλου μέσω της σύνθεσης

«Περισσότερα κεφάλια είναι καλύτερα από ένα»

«Η σοφία του πλήθους»

Σύνολο μοντέλων: εκπαίδευση πολλαπλών μοντέλων αντί ενός για την επίτευξη καλύτερων προγνωστικών επιδόσεων

Περισσότερα στην επόμενη διάλεξη.





Σας ευχαριστούμε

