

Βελτιστοποίηση (Optimization)

Μηχανική Μάθηση

ΔΠΜΣ Επιστήμης Δεδομένων & Μηχανικής Μάθησης

Γιώργος Αλεξανδρίδης – gealexan@mail.ntua.gr

Εισαγωγικές Έννοιες

Το ζητούμενο της επιβλεπόμενης μάθησης

- Εύρεση εκείνων των τιμών των παραμέτρων θ του συστήματος μάθησης, για τις οποίες η **αντικειμενική συνάρτηση** (*objective function*) ή **συνάρτηση απώλειας** (*loss function*) $J(\theta)$ γίνεται **βέλτιστη**

- Παραδείγματα

1. Γραμμική Παλινδρόμηση

$$J(\theta) = \sum_{i=1}^N \|x_i \theta - y\|^2 \text{ και } \min_{\theta} J(\theta)$$

2. Εκτίμηση Μέγιστης Πιθανοφάνειας

$$J(\theta) = \sum_{i=1}^N \log p_{\theta}(x_i) \text{ και } \max_{\theta} J(\theta)$$

3. Μηχανικές Διανυσμάτων Υποστήριξης

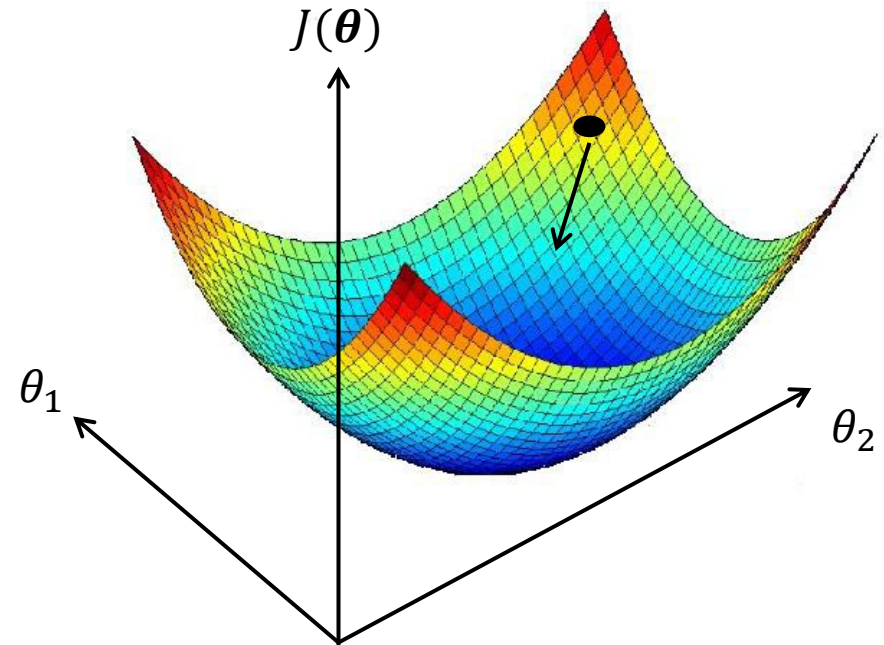
$$J(\theta, \xi_i) = \|\theta\|^2 + C \sum_{i=1}^N \xi_i \text{ και } \min_{\theta} J(\theta)$$

$$\text{subject to } \xi_i \leq 1 - y_i x_i^T \theta, \xi_i \geq 0$$

- Χωρίς βλάβη της γενικότητας, θα υποθέσουμε ότι το ζητούμενο είναι να **ελαχιστοποιήσουμε την αντικειμενική συνάρτηση**

Συναρτήσεις απώλειας

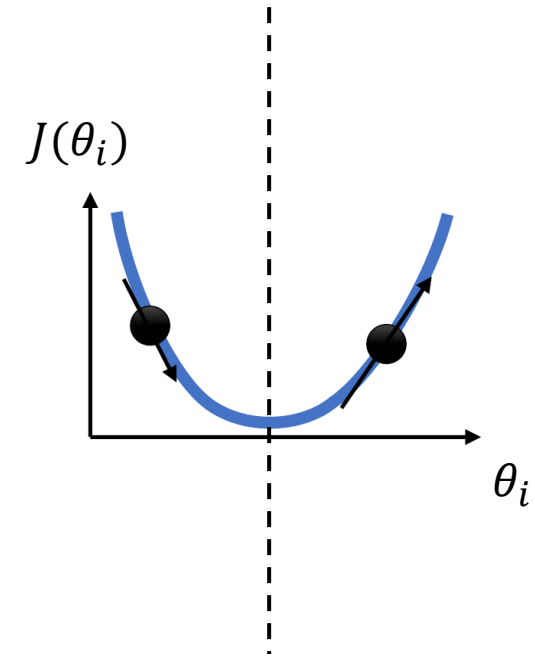
- Υποθέτουμε ότι το μοντέλο μας έχει δύο παραμέτρους (δεξιά σχήμα)
- Βέλτιστη τιμή παραμέτρων θ^*
$$\theta^* \leftarrow \arg \min_{\theta} J(\theta)$$
- Απλός αλγόριθμος προσδιορισμού θ^*
 1. Ξεκίνα με τυχαία αρχική ανάθεση θ
 2. Βρες κατεύθυνση v όπου η $J(\theta)$ μειώνεται
 3. $\theta \leftarrow \theta + \eta v$
 4. Επανάλαβε τα βήματα 2 ως 4 μέχρι τη σύγκλιση στο θ^*
- Το η είναι μια μικρή σταθερά που καλείται **ρυθμός μάθησης** (*learning rate*) ή **βήμα** (*step size*)



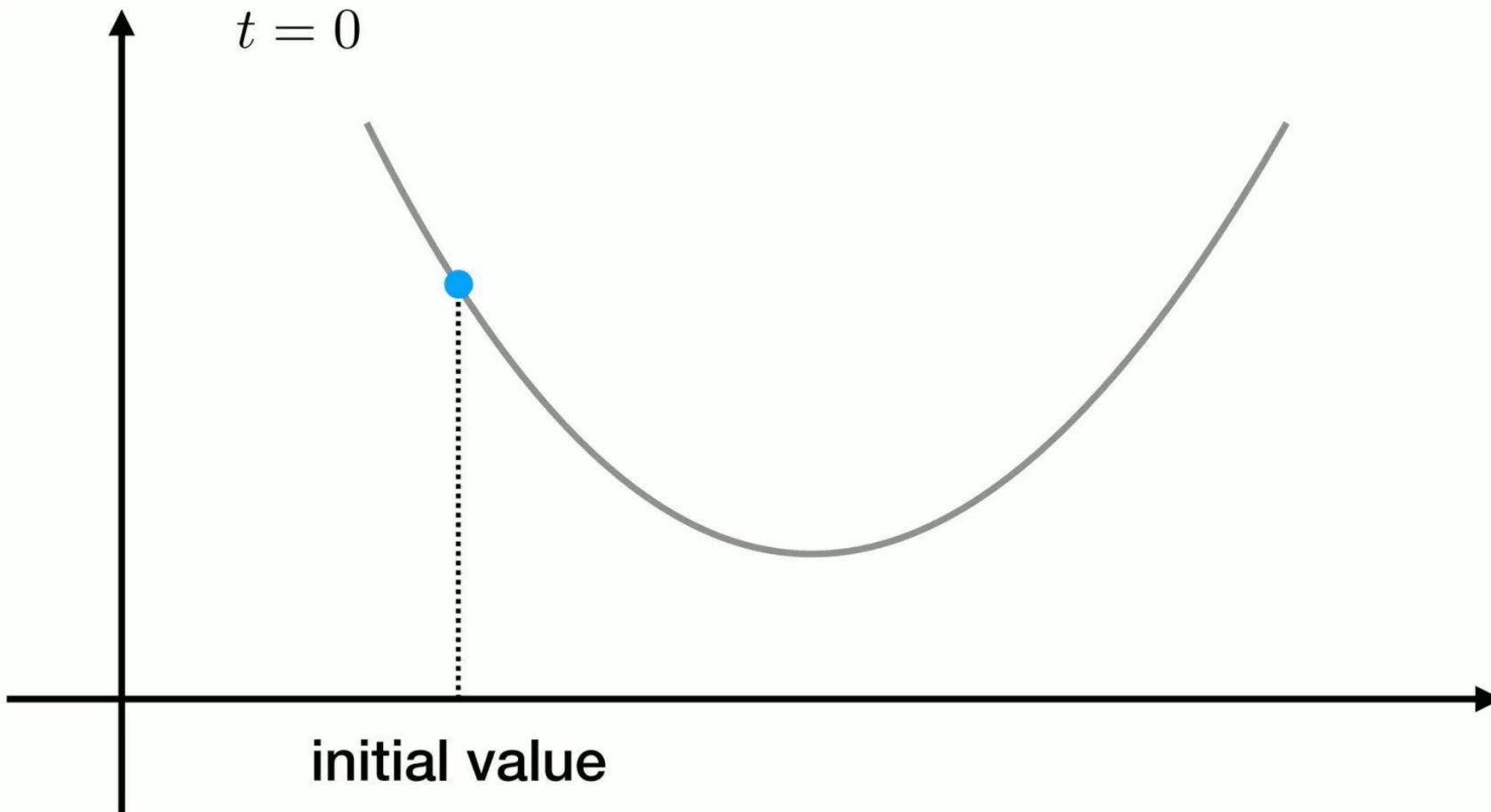
Κατάβαση Κλίσης (Gradient Descent)

- Προς τα ποια κατεύθυνση *μειώνεται* η *αντικειμενική συνάρτηση*;
 - Υπολογισμός της *κλίσης* $\frac{\partial J(\theta)}{\partial \theta_i}$ της $J(\theta)$ ως προς παράμετρο θ_i
- **Κατεύθυνση μείωσης κλίσης $J(\theta_i)$** (δεξιά σχήμα)
 - Αν η *κλίση* της J ως προς θ_i είναι **αρνητική** (αριστερό υποτμήμα), θα πρέπει να *κινηθούμε προς τα δεξιά*
 - Αν η *κλίση* της J ως προς θ_i είναι **θετική** (δεξιό υποτμήμα), θα πρέπει να *κινηθούμε προς τα αριστερά*
- Σε κάθε περίπτωση και για κάθε παράμετρο *κινούμαστε* σε κατεύθυνση **αντίθετη της κλίσης** της αντικειμενικής συνάρτησης ως προς τη συγκεκριμένη παράμετρο $v_i = -\frac{\partial J(\theta)}{\partial \theta_i}$

$$\theta \leftarrow \theta - \eta \nabla_{\theta} J(\theta)$$
$$\nabla_{\theta} J(\theta) = \left[\frac{\partial J(\theta)}{\partial \theta_1}, \frac{\partial J(\theta)}{\partial \theta_2}, \dots, \frac{\partial J(\theta)}{\partial \theta_n} \right]$$



Παράδειγμα Κατάβασης Κλίσης



Κατάβαση Κλίσης

Gradient Descent

Τεχνικές Κατάβασης Κλίσης

- Διαφέρουν ως προς την ποσότητα των δεδομένων εκπαίδευσης που χρησιμοποιούνται σε κάθε ενημέρωση
 1. Κατάβαση κλίσης με **ενιαία δέσμη** δεδομένων (*batch gradient descent*)
 2. **Στοχαστική** κατάβαση κλίσης (*stochastic gradient descent* - SGD)
 3. Κατάβαση κλίσης με **μικρό-δέσμες** δεδομένων (*mini-batch gradient descent*)

Ενιαία δέσμη δεδομένων

- Υπολογίζει την *κλίση* σε όλα τα N στιγμιότυπα του συνόλου δεδομένων εκπαίδευσης

$$\theta \leftarrow \theta - \eta \nabla_{\theta} \sum_{i=1}^N J(\theta; x_i, y_i)$$

- **Πλεονεκτήματα**

- Εγγυημένη σύγκλιση στο ολικό ελάχιστο για **κυρτές** (*convex*) αντικειμενικές συναρτήσεις και σε τοπικό ελάχιστο για **μη-κυρτές** (*non-convex*)

- **Μειονεκτήματα**

- Πολύ αργή σύγκλιση
- **Μη-υπολογίσιμη** (*intractable*) για πολύ μεγάλα σύνολα δεδομένων
 - Που δεν χωράνε, δηλαδή, στη μνήμη του υπολογιστή
- Δεν υποστηρίζει την *online μάθηση*

Στοχαστική κατάβαση κλίσης

- Υπολογίζει την κλίση σε κάθε στιγμιότυπο x_i του συνόλου δεδομένων εκπαίδευσης

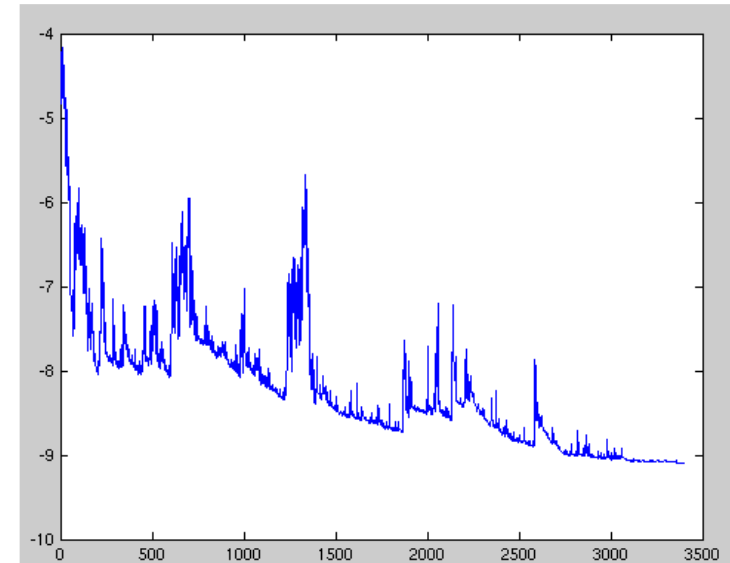
$$\theta \leftarrow \theta - \eta \nabla_{\theta} J(\theta; x_i, y_i)$$

- Πλεονεκτήματα**

- Πολύ πιο γρήγορη* από την κατάβαση κλίσης ενιαίας δέσμης δεδομένων (προηγούμενη περίπτωση)
- Υποστηρίζει την online μάθηση*

- Μειονεκτήματα**

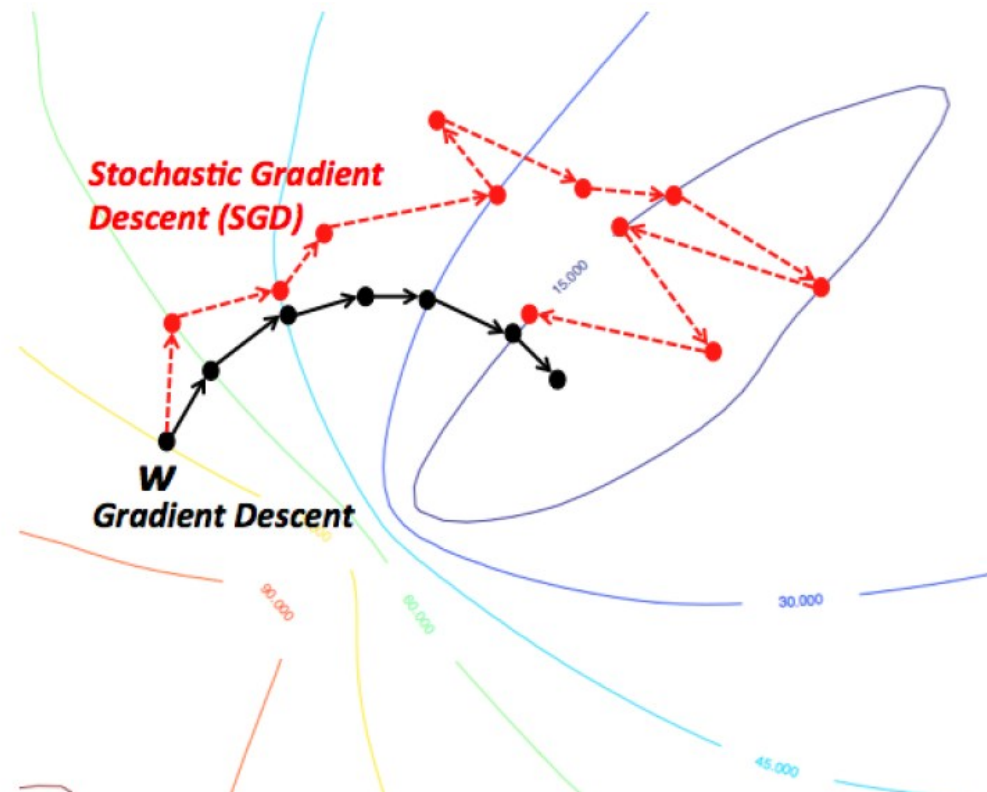
- Μπορεί να εμφανίζεται *μεγάλη απόκλιση* στις ενημερώσεις των παραμέτρων



Πηγή: https://en.wikipedia.org/wiki/Stochastic_gradient_descent#/media/File:Stogra.png

Σύγκριση των δύο μεθόδων

- Η *στοχαστική κατάβαση κλίσης* εμφανίζει παρόμοια συμπεριφορά σύγκλισης με τη *κατάβαση κλίσης ενιαίας δέσμης* αν ο **ρυθμός μάθησης μειώνεται σταδιακά** με το χρόνο



Κατάβαση κλίσης σε μικρό-δέσμες δεδομένων

- Υπολογίζει την κλίση σε όλα τα $p \ll N$ στιγμιότυπα του συνόλου δεδομένων εκπαίδευσης

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \sum_{i=1}^p J(\boldsymbol{\theta}; x_i, y_i)$$

- **Πλεονεκτήματα**

- *Μειώνει τις αποκλίσεις μεταξύ των ενημερώσεων*
- Μπορεί να υπολογιστεί γρήγορα με πράξεις πολλαπλασιασμού πινάκων

- **Μειονεκτήματα**

- Το μέγεθος δέσμης είναι **υπερ-παράμετρος** της διαδικασίας εκπαίδευσης
 - Άρα πρέπει να βρεθεί η βέλτιστη τιμή της
 - Συνήθως χρησιμοποιούνται δυνάμεις του 2 μεταξύ 8 και 256.

- *Συνηθέστερα* χρησιμοποιούμενη τεχνική

- Αναφέρεται και ως *στοχαστική κατάβαση κλίσης* παρότι χρησιμοποιούνται μικροδέσμες

Σύγκριση τεχνικών

Τεχνική	Ακρίβεια	Ταχύτητα Ενημέρωσης	Χρήση Μνήμης	Online Μάθηση
Ενιαία δέσμη	Καλή	Χαμηλή	Υψηλή	Όχι
Στοχαστική	Καλή*	Υψηλή	Χαμηλή	Ναι
Μικρο-δέσμες	Καλή	Μεσαία	Μεσαία	Ναι

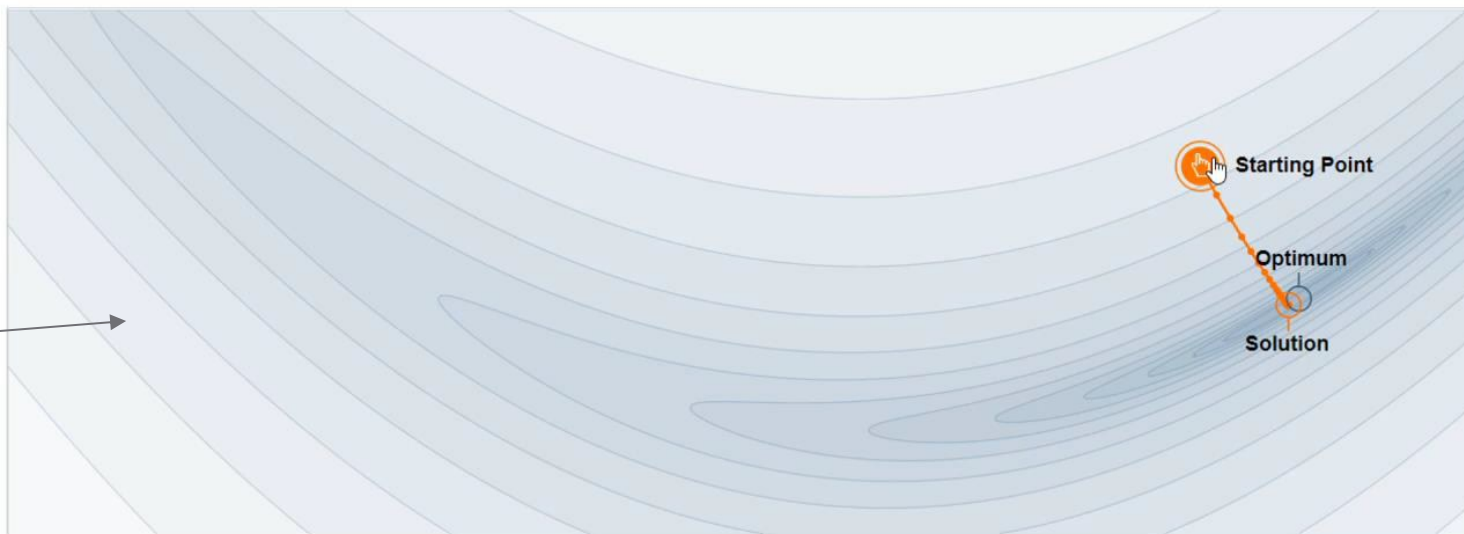
* Υπό την προϋπόθεση ότι μειώνεται σταδιακά ο ρυθμός μάθησης

Προκλήσεις

1. Αποφυγή τοπικών ελαχίστων
2. Επιλογή ρυθμού μάθησης
3. Καθορισμός διαδικασίας μεταβολής (μείωσης) του ρυθμού μάθησης

Απεικόνιση κατάβασης κλίσης

Κάθε καμπύλη
ίδιου χρώματος
έχει την ίδια
τιμή $J(\theta)$



Step-size $\alpha = 0.00043$



Momentum $\beta = 0.0$



We often think of Momentum as a means of dampening speeding up the iterations, leading to faster convergence. It allows a larger range of other interesting behavior. What is going

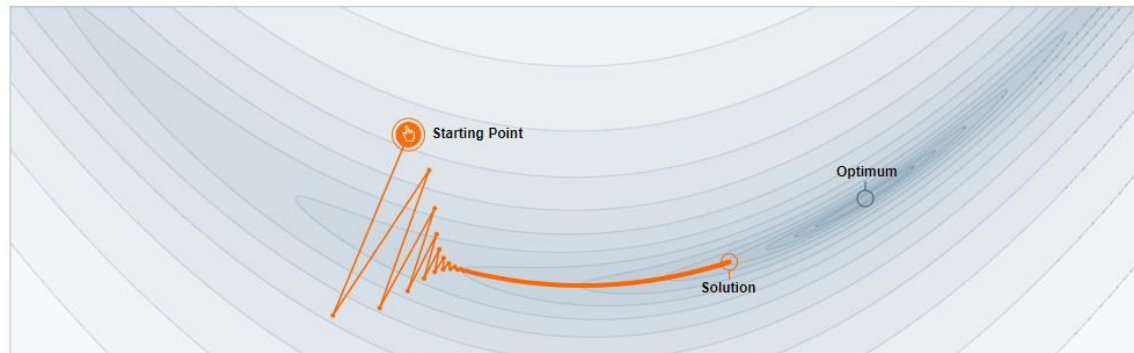
Πηγή: <https://distill.pub/2017/momentum/>

Επιφάνειες Συναρτήσεων Απώλειας

Loss surfaces

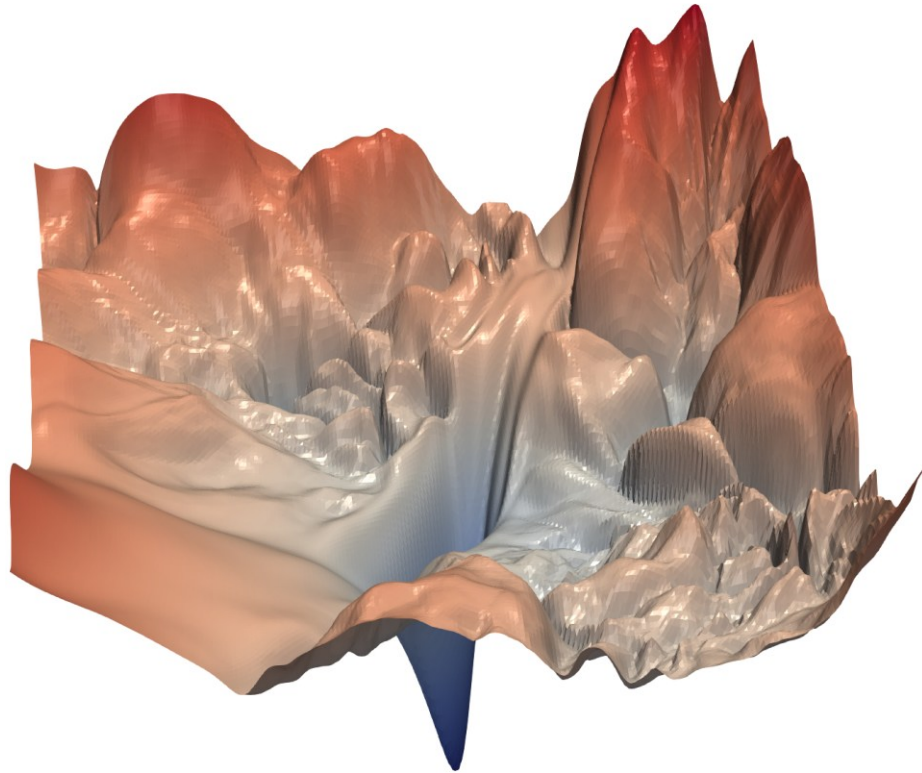
Κίνηση αντίθετα από την κλίση

- Η *ενημέρωση των παραμέτρων* του μοντέλου σε κατεύθυνση *αντίθετη* από την *κλίση* της συνάρτησης απώλειας δεν εγγυάται πάντα το *καλύτερο* αποτέλεσμα
 - Ειδικά αν η συνάρτηση απώλειας είναι *μη-κυρτή*
- Υπενθύμιση
 - **Κυρτές** συναρτήσεις είναι όσες είναι *διπλά-παραγωγίσιμες* στο πεδίο ορισμού τους και ταυτόχρονα η *δεύτερη παράγωγος* τους είναι παντού στο πεδίο ορισμού τους *μη-αρνητική*

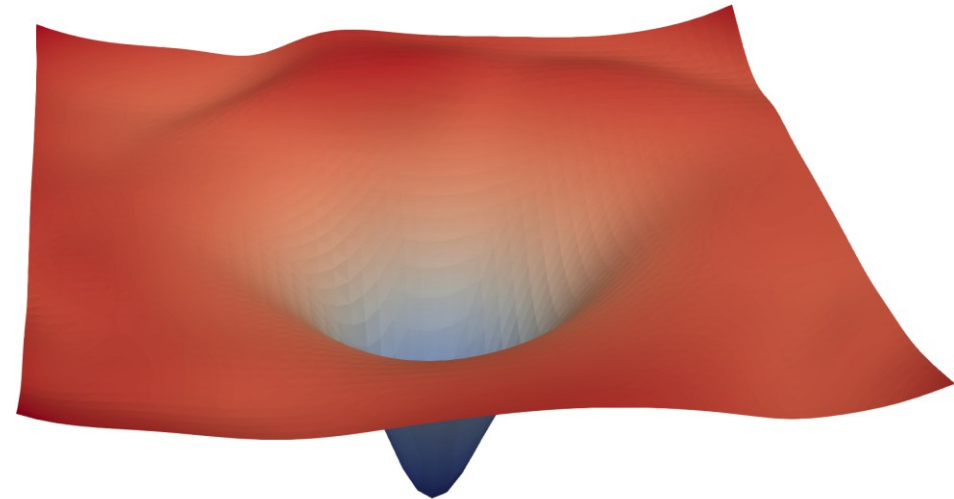


Συναρτήσεις απώλειας νευρωνικών δικτύων

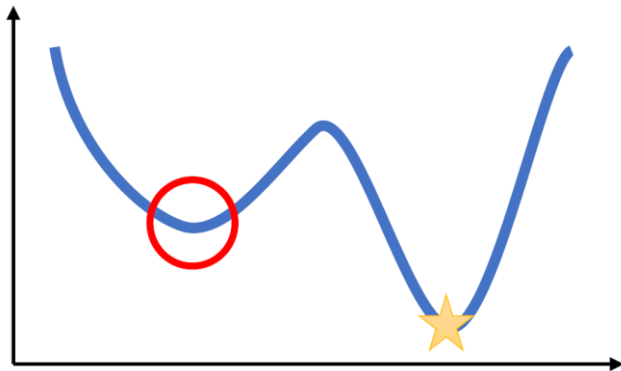
ResNet-56 χωρίς skip connections



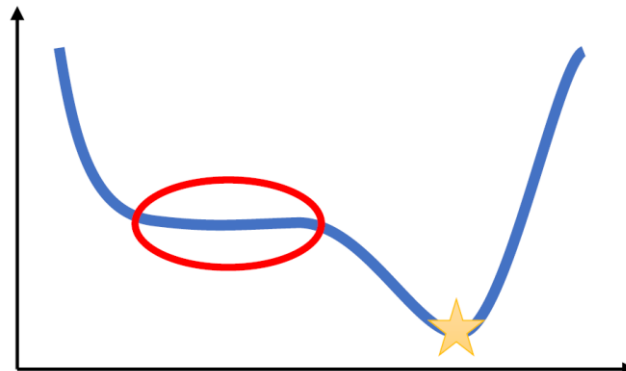
ResNet-56 με skip connections



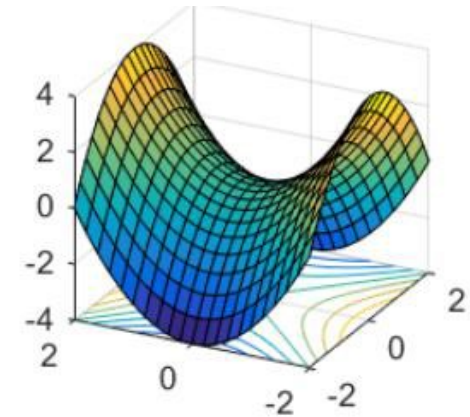
Ειδικές περιπτώσεις μη-κυρτότητας



Τοπικό ελάχιστο



«Οροπέδιο»



Σαγματικό Σημείο

Ειδικές περιπτώσεις μη-κυρτότητας

- **Τοπικά ελάχιστα**

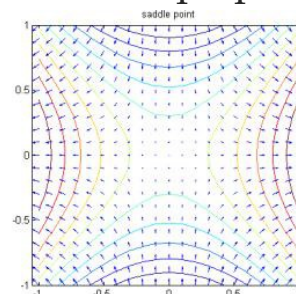
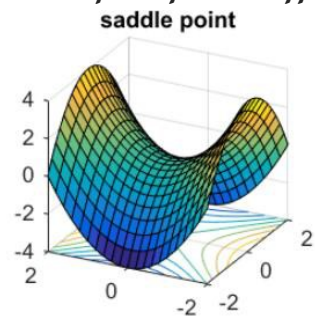
- Παρότι φαίνονται να είναι σοβαρό πρόβλημα, στην πράξη η *επίδρασή τους μετριάζεται όσο αυξάνουν οι παράμετροι του μοντέλου*
- Σε *μεγάλα μοντέλα*, παρότι υπάρχουν, *εμπειρικά* έχει φανεί πως δεν είναι πολύ χειρότερα από τα ολικά ελάχιστα

- **Οροπέδια**

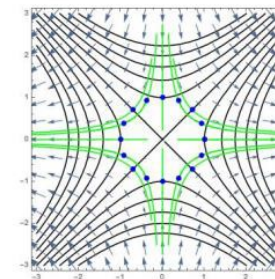
- *Δεν πρέπει να επιλέγουμε εξ'αρχής πολύ μικρούς ρυθμούς μάθησης για να μην «κολλάμε» σε οροπέδια*

- **Σαγματικά σημεία**

- *Πολύ μικρές κλίσεις στα σαγματικά σημεία*
- Στην πράξη έχει παρατηρηθεί ότι *τα περισσότερα «κρίσιμα» σημεία σε συναρτήσεις σφάλματος νευρωνικών δικτύων είναι σαγματικά σημεία*



Gradient vectors



Gradient flows (green)

Επιτάχυνση κλίσης

Μέθοδος Newton

- Ανάπτυγμα Taylor συνάρτησης απώλειας γύρω από σημείο θ_0

$$J(\theta) \approx J(\theta_0) + \nabla_{\theta} J(\theta_0)(\theta - \theta_0) + \frac{1}{2}(\theta - \theta_0)^T \nabla_{\theta}^2 J(\theta_0)(\theta - \theta_0)$$

- Αν το μοντέλο έχει n παραμέτρους τότε

- **Κλίση** $\nabla_{\theta} J(\theta_0) = \left[\frac{\partial J(\theta_0)}{\partial \theta_1}, \frac{\partial J(\theta_0)}{\partial \theta_2}, \dots, \frac{\partial J(\theta_0)}{\partial \theta_n} \right]$ έχει n παραμέτρους

- **Εσσιανός Πίνακας** $\nabla_{\theta}^2 J(\theta_0) = \begin{bmatrix} \frac{\partial^2 J(\theta_0)}{\partial \theta_1 \partial \theta_1} & \dots & \frac{\partial^2 J(\theta_0)}{\partial \theta_1 \partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 J(\theta_0)}{\partial \theta_n \partial \theta_1} & \dots & \frac{\partial^2 J(\theta_0)}{\partial \theta_n \partial \theta_n} \end{bmatrix}$ έχει n^2 παραμέτρους

- **Κλίση** γύρω από το τοπικό ελάχιστο θ^* μηδενική, οπότε

$$\theta^* \leftarrow \theta_0 - \left(\nabla_{\theta}^2 J(\theta_0) \right)^{-1} \nabla_{\theta} J(\theta_0)$$

- Όρος $\left(\nabla_{\theta}^2 J(\theta_0) \right)^{-1} \nabla_{\theta} J(\theta_0)$ έχει *πολυπλοκότητα* $\mathcal{O}(n^3)$ που τον καθιστά απαγορευτικό για μεγάλα μοντέλα

- Για αυτό το λόγο αποφεύγουμε μεθόδους που απαιτούν τον υπολογισμό δευτέρων παραγώγων και προσπαθούμε να «επιταχύνουμε» την κατάβαση κλίσης

Ορμή (Momentum)

- **Κεντρική Ιδέα**

- Αν διαδοχικές κλίσεις «δείχνουν» προς την ίδια κατεύθυνση, θα πρέπει να κινηθούμε προς τα εκεί πιο γρήγορα

- **Όρος ορμής**

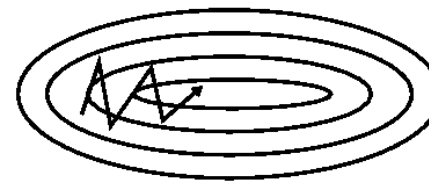
- Προσθήκη ενός κλάσματος γ (συνήθως 0,9) του διανύσματος του προηγούμενου βήματος στο τωρινό

$$v_t \leftarrow \gamma v_{t-1} + \eta \nabla_{\theta} J(\theta) \text{ και } \theta \leftarrow \theta - v_t$$

- Μειώνει το εύρος της ενημέρωσης για τις παραμέτρους εκείνες που αλλάζουν κατεύθυνση στην κλίση
- Αυξάνει το εύρος της ενημέρωσης για τις παραμέτρους εκείνες που δεν αλλάζουν κατεύθυνση στην κλίση



SGD χωρίς ορμή

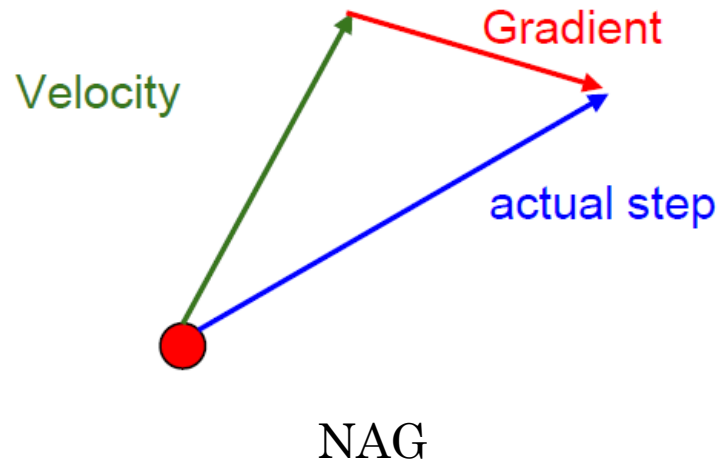
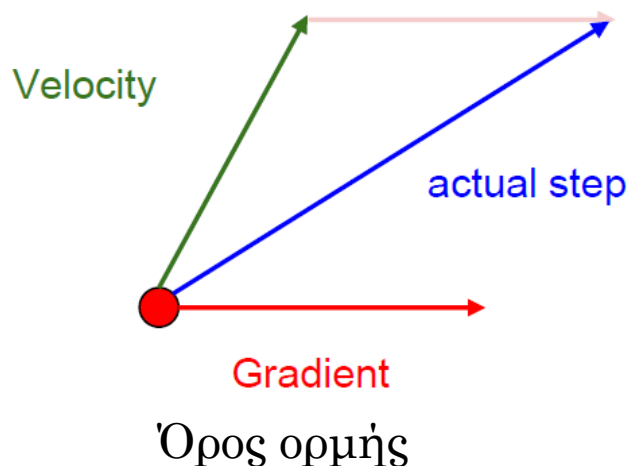


SGD με ορμή

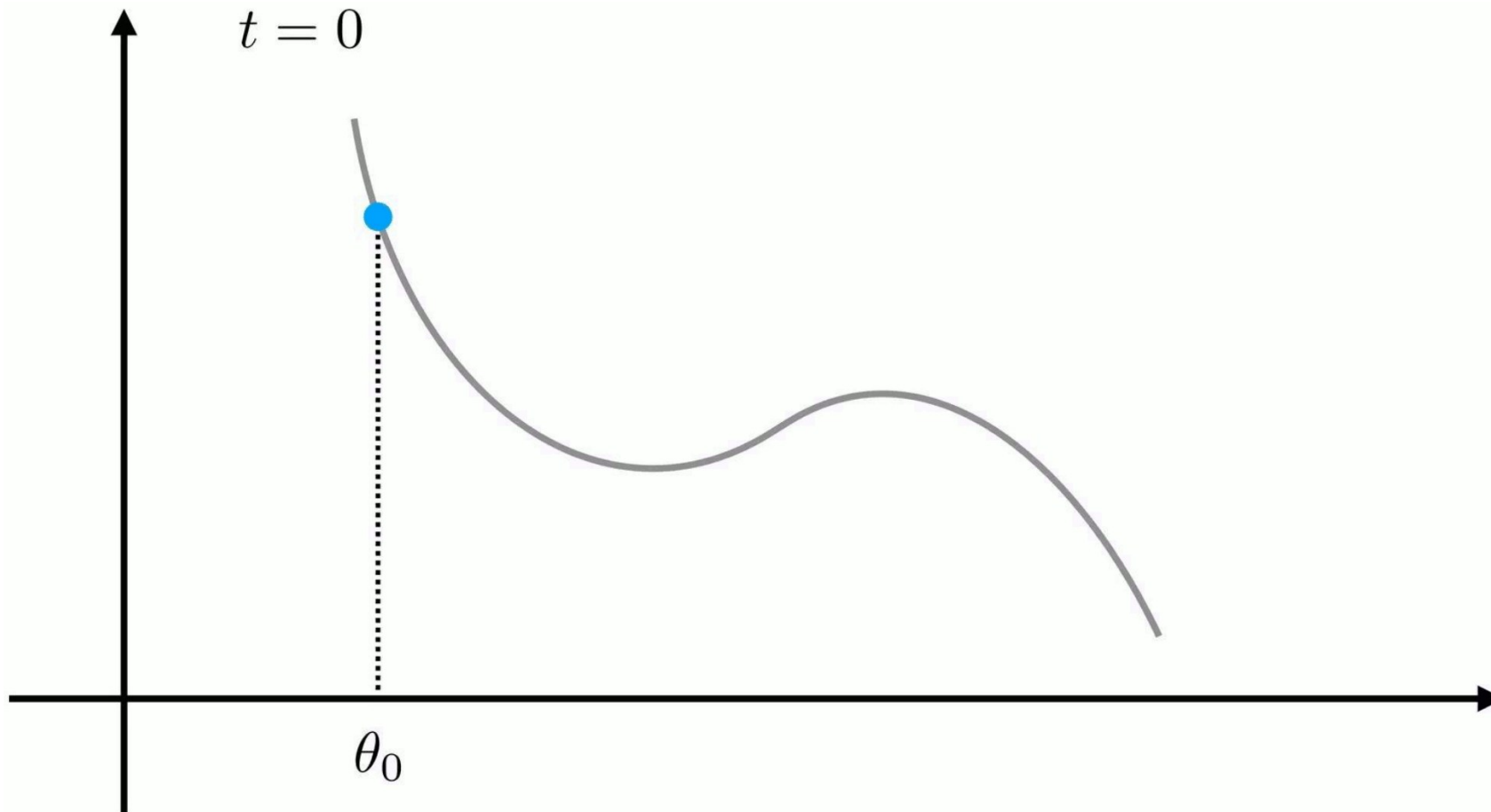
Επιταχυνόμενη Κλίση Nesterov

- Η προσθήκη όρου ορμής επιταχύνει την κατάβαση στα τυφλά
 - Πρώτα υπολογίζει την κλίση και στη συνέχεια πραγματοποιεί ένα μεγάλο άλμα
- Η **επιταχυνόμενη κλίση Nesterov** (*Nesterov Accelerated Gradient – NAG*)
 - Πρώτα πραγματοποιεί το άλμα στην κατεύθυνση της προηγούμενης κλίσης $\theta - \gamma v_{t-1}$
 - Κατόπιν «μετρά» που έχει καταλήξει και κάνει διόρθωση των παραμέτρων

$$v_t \leftarrow \gamma v_{t-1} + \eta \nabla_{\theta} J(\theta - \gamma v_{t-1}) \text{ και } \theta \leftarrow \theta - v_t$$

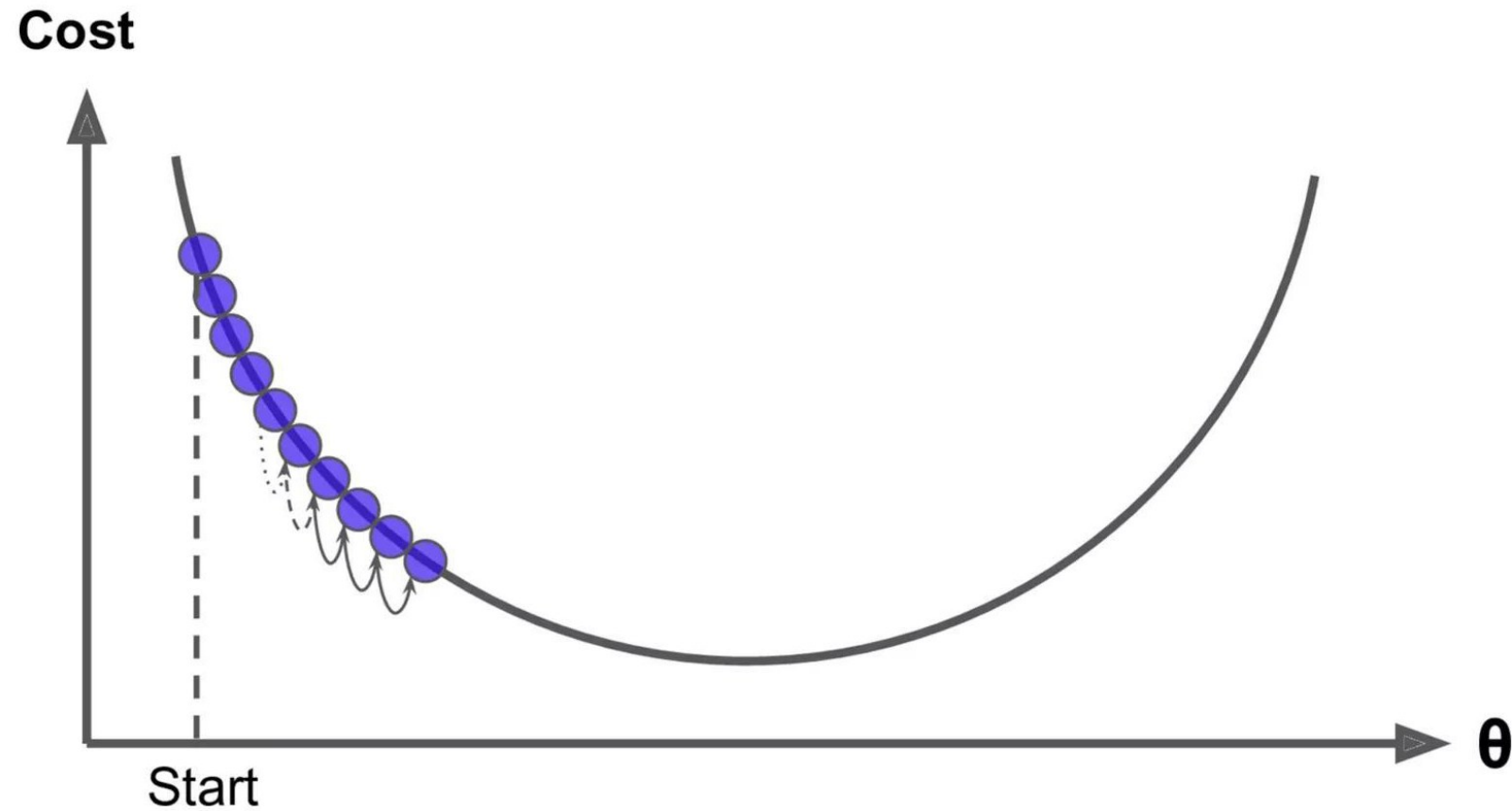


Κατάβαση κλίσης με ορμή

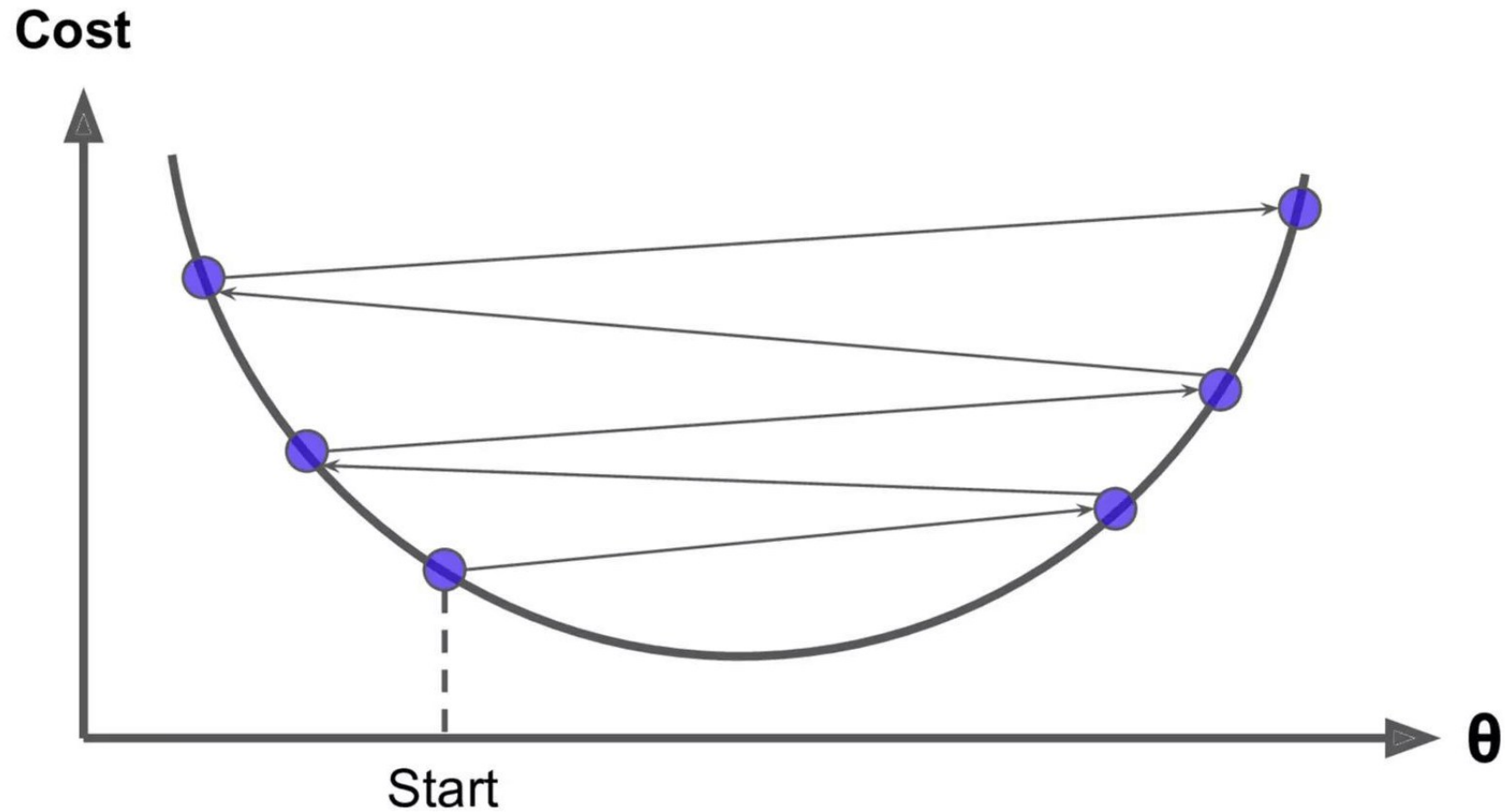


Επίδραση Ρυθμού Μάθησης

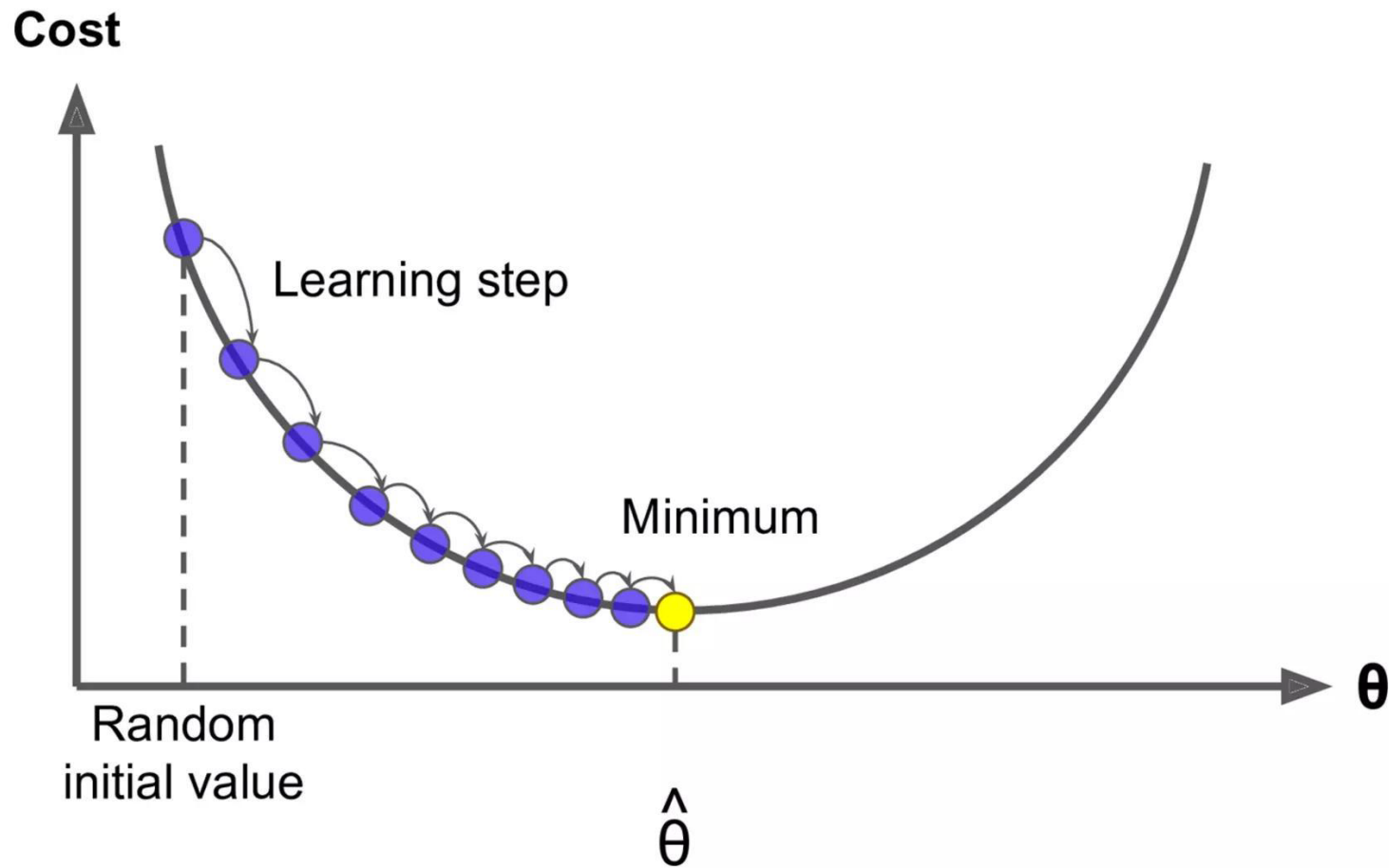
Χαμηλός ρυθμός μάθησης



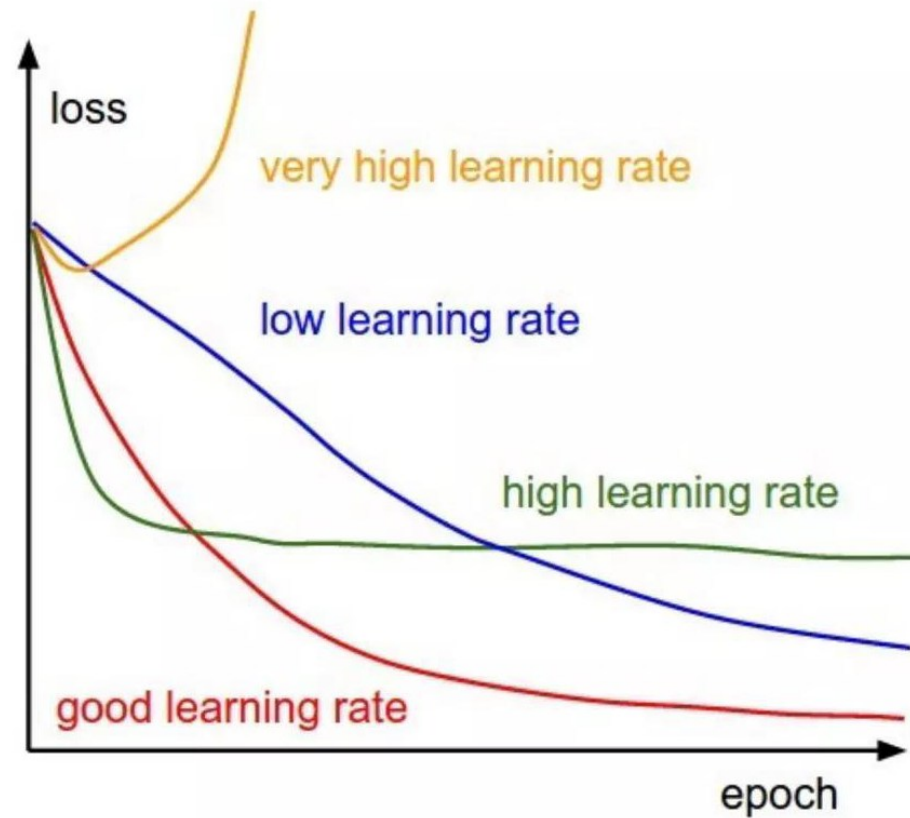
Υψηλός ρυθμός μάθησης



Κανονικός ρυθμός μάθησης



Επίδραση ρυθμού μάθησης



Adagrad

- Προσαρμόζει το ρυθμό μάθησης στις παραμέτρους του μοντέλου
 - Μεγάλες ενημερώσεις για μη-συχνές παραμέτρους, μικρές ενημερώσεις για συχνές παραμέτρους
- Κανόνας ενημέρωσης SGD: $\theta_{t+1} = \theta_t - \eta \cdot g_t$ και $g_t = \nabla_{\theta_t} J(\theta_t)$
- Ο Adagrad διαιρεί το ρυθμό μάθησης με την τετραγωνική ρίζα του αθροίσματος των τετραγώνων των προηγούμενων κλίσεων

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \odot g_t$$

- $G_t \in \mathbb{R}^{d \times d}$ διαγώνιος πίνακας με το i-οστό του στοιχείο να είναι ίσο με το άθροισμα των τετραγώνων των κλίσεων του θ_i μέχρι τη χρονική στιγμή t
- ϵ όρος ομαλοποίησης για να αποφευχθεί η διαίρεση με το μηδέν
- $\odot$ πολλαπλασιασμός Hadamard

Adagrad

- **Πλεονεκτήματα**

- *Κατάλληλος για χρήση όταν τα δεδομένα είναι αραιά*
- *Βελτιώνει σημαντικά την ευρωστία του SGD*
- *Δεν απαιτεί τη «χειροκίνητη» ρύθμιση του ρυθμού μάθησης*

- **Μειονεκτήματα**

- *Συσσωρεύει τετραγωνικές κλίσεις στον παρονομαστή και έτσι προκαλεί μεγάλη μείωση του ρυθμού μάθησης*

Adadelta

- Αντιμετωπίζει το *μειονέκτημα* του *Adagrad* μέσω του *περιορισμού* του *πλήθους* των *παρελθοντικών κλίσεων* που λαμβάνονται υπόψη, σε παράθυρο σταθερού μεγέθους

- **Ενημέρωση SGD**

$$\theta_{t+1} = \theta_t + \Delta\theta_t \text{ και } \Delta\theta_t = -\eta \cdot g_t$$

- Ορίζει έναν *τρέχοντα μέσο όρο* του τετραγώνου των κλίσεων $\mathbb{E}[g^2]_t$ τη χρονική στιγμή t

$$\mathbb{E}[g^2]_t = \gamma \mathbb{E}[g^2]_{t-1} + (1 - \gamma) g_t^2$$

- γ : όρος αντίστοιχου του παράγοντα ορμής, συνήθως λαμβάνει την τιμή 0,9

- **Ενημέρωση Adadelta**

$$\Delta\theta_t = -\frac{\eta}{\sqrt{\mathbb{E}[g^2]_t + \epsilon}} g_t$$

RMSProp

- Προτάθηκε από τον *Geoff Hinton* την ίδια περίοδο με τον *Adadelta* και έχει παρόμοια φιλοσοφία
- Επίσης διαιρεί το ρυθμό μάθησης με τον τρέχοντα μέσο όρο του τετραγώνου των κλίσεων

$$\begin{aligned}\mathbb{E}[g^2]_t &= \gamma \mathbb{E}[g^2]_{t-1} + (1 - \gamma) g_t^2 \\ \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t - \frac{\eta}{\sqrt{\mathbb{E}[g^2]_t + \epsilon}} g_t\end{aligned}$$

- Ο όρος φθοράς γ συνήθως τίθεται στο 0,9 ενώ ο ρυθμός μάθησης η στο 0,001

Επίδραση Ρυθμού Μάθησης και Ορμής

Adaptive Moment Estimation (Adam)

- Όπως οι *Adadelta* και *RMSprop*, ο **Adam** υπολογίζει τον κινούμενο μέσο όρο των προηγούμενων τετραγωνικών κλίσεων v_t
- Όπως οι βελτιστοποιητές με παράγοντα ορμής, υπολογίζει τον κινούμενο μέσο όρο των προηγούμενων κλίσεων m_t

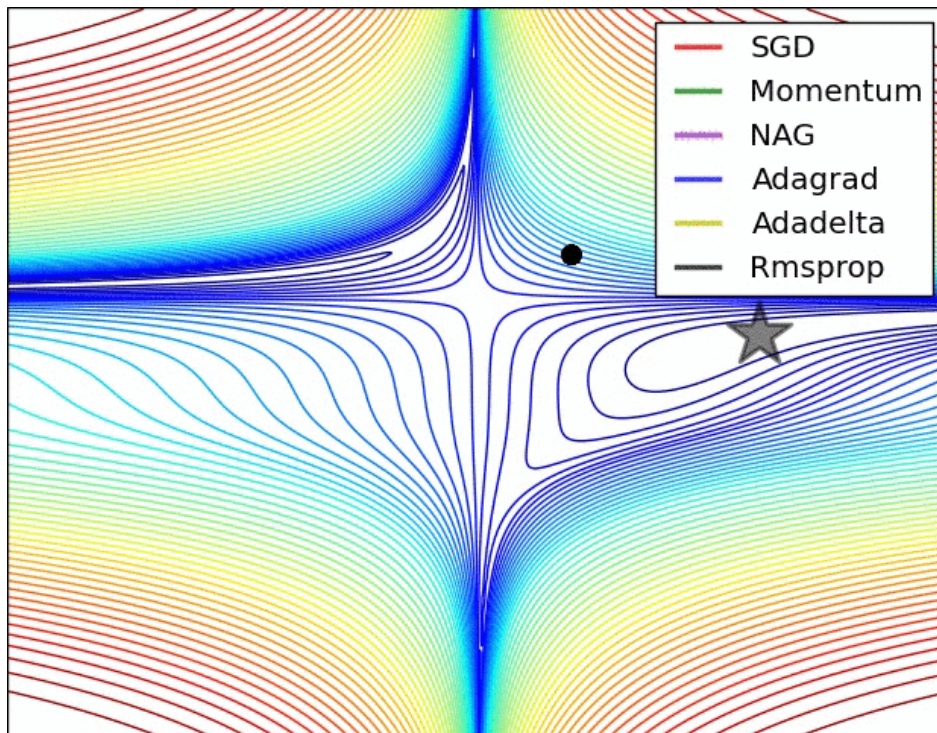
- **Ενημέρωση Adam**

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \end{aligned}$$

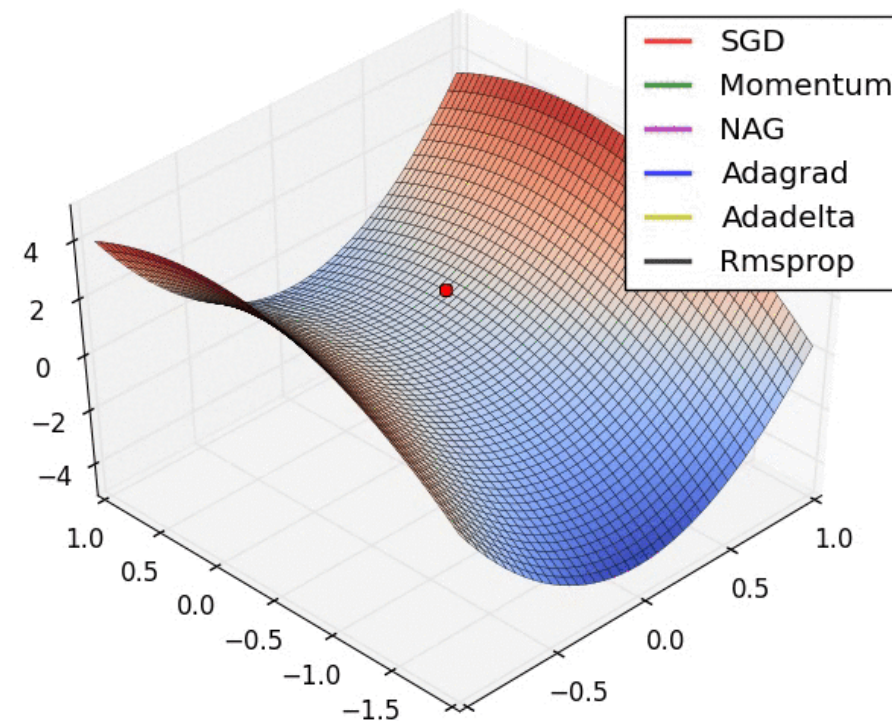
- m_t : μέση τιμή κλίσεων (διάνυσμα που αρχικοποιείται στο 0)
 - v_t : (μη-κεντραρισμένη) διασπορά κλίσεων (διάνυσμα που αρχικοποιείται στο 0)
 - β_1, β_2 : ρυθμός φθοράς
- Αφαίρεση **μεροληψίας** (*bias*) προς το 0 από m_t, v_t
$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t} \text{ και } \widehat{v}_t = \frac{v_t}{1 - \beta_2^t}$$
- Κανόνας Ενημέρωσης Adam: $\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\widehat{v}_t} + \epsilon} \widehat{m}_t$

Οπτικοποίηση των αλγορίθμων

Επιφάνειες συναρτήσεων απώλειας



Σαγματικό σημείο



Πηγή: <https://imgur.com/a/visualizing-optimization-algos-Hqolp>

Καταλληλότητα

- Μέθοδοι του *προσαρμοστικού* ρυθμού μάθησης (Adagrad, Adadelta, RMSProp, Adam) είναι *καταλληλότεροι* για *προβλήματα με αραιά χαρακτηριστικά*
- *Adagrad, Adadelta, RMSProp, Adam* έχουν *ικανοποιητική* απόδοση σε *παρόμοιες συνθήκες*
- Οι *δημιουργοί* του *Adam* ισχυρίζονται ότι το *βήμα διόρθωσης της μεροληψίας* τον *καθιστά ελαφρώς καλύτερο* από τον *RMSProp*

Βιβλιογραφία

- Quian, N. (1999) - “On the momentum term in gradient descent learning algorithms” – *Neural networks : the official journal of the International Neural Network Society*, 12(1):145-151
- Nesterov, Y. (1983) – “A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$ ” - *Doklady ANSSSR (translated as Soviet.Math.Docl.)*, 269:543-547
- Duchi, J. (2011) – “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization” - *Journal of Machine Learning Research*, 12:2121-2159.
- Zeiler, M. D. (2012) – “ADADELTA: An Adaptive Learning Rate Method” *arXiv preprint arXiv:1212.5701*.
- Kingma, D. P. (2015) – “Adam: a Method for Stochastic Optimization.” *International Conference on Learning Representations*, pages 1-13.