TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**DỰ ÁN CUỐI KÌ MÔN**

**NHẬP MÔN HỌC MÁY**

**NGHIÊN CỨU, ĐÁNH GIÁ**

**NHỮNG VẤN ĐỀ HỌC MÁY**

*Người hướng dẫn*: **LÊ ANH CƯỜNG**

*Người thực hiện*: **HOÀNG TRUNG KIÊN – 52100903**

Lớp **: 21050301**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**DỰ ÁN CUỐI KÌ MÔN**

**NHẬP MÔN HỌC MÁY**

**NGHIÊN CỨU, ĐÁNH GIÁ**

**NHỮNG VẤN ĐỀ HỌC MÁY**

Người hướng dẫn: **LÊ ANH CƯỜNG**

Người thực hiện: **HOÀNG TRUNG KIÊN**

Lớp **: 21050301**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến Trường Đại học Tôn Đức Thắng đã đưa môn học Nhập Môn Học Máy vào chương trình giảng dạy.

Lời cảm ơn thứ hai, em xin dành cho khoa Công nghệ thông tin đã cung cấp những tài liệu học tập đầy đủ, chi tiết.

Xin gửi lời cảm ơn chân thành thầy Lê Anh Cường, người đã giảng dạy chúng em trong suốt quá trình tham gia môn học. Chúng em rất biết ơn những bài tập thầy giao đã giúp chúng em củng cố kiến thức, đã giải đáp mọi thắc mắc của chúng em.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi / chúng tôi và được sự hướng dẫn của Thầy Lê Anh Cường;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Kiên*

*Hoàng Trung Kiên*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Bài báo cáo này sẽ chú trọng vào hai vấn đề như sau

* Tìm hiểu và so sánh các phương pháp Optimizer trong việc huấn luyện mô hình học máy.
* Tìm hiểu về Continual Learning và Test Production khi xây dựng một giải pháp học máy để giải quyết một bài toán nào đó.

**MỤC LỤC**

[DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT 3](#_Toc154147774)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 4](#_Toc154147775)

[CHƯƠNG 1 – TÌM HIỂU, SO SÁNH CÁC PHƯƠNG PHÁP OPTIMIZER 5](#_Toc154147776)

[1.1 Optimizer là gì 5](#_Toc154147777)

[1.2 Các loại Optimizer 5](#_Toc154147778)

[1.1.1 Gradient Descent 5](#_Toc154147779)

[1.1.2 Stochastic Gradient Descent 6](#_Toc154147780)

[1.1.3 Mini-Batch Gradient Descent 7](#_Toc154147781)

[1.1.4 Momentum 7](#_Toc154147782)

[1.1.5 Adagrad 9](#_Toc154147783)

[1.1.6 RMSProp 9](#_Toc154147784)

[1.1.7 Adam 10](#_Toc154147785)

[1.3 Đánh giá các loại Optimizer 12](#_Toc154147786)

[CHƯƠNG 2 – TÌM HIỂU VỀ CONTINUAL LEARNING VÀ TEST PRODUCTION 14](#_Toc154147787)

[1.1 Continual Learning là gì 14](#_Toc154147788)

[1.2 Test Production là gì 14](#_Toc154147789)

[1.3 Ứng dụng trong trong giải quyết bài toán học máy 15](#_Toc154147790)

[CHƯƠNG 3 – TỔNG KẾT 16](#_Toc154147791)

[3.1 Kết quả đạt đươc 16](#_Toc154147792)

[TÀI LIỆU THAM KHẢO 17](#_Toc154147793)

# DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

**CÁC CHỮ VIẾT TẮT**

GD Gradient Descent

SGD Stochastic Gradient Descent

Adagrad Adaptive Gradient

RMSProp Root Mean Sqaure Propagation

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

Hình 1.1: Cách Gradient Descent hoạt động 5

Hình 1.2: So sánh giữa SGD và GD 6

Hình 1.3: So sánh GD và Momentum 7

Hình 1.4: Công thức tính Adagrad 8

Hình 1.5: Công thức của RMSProp 9

Hình 1.6: Mô tả cách hoạt động của Adam 10

**DANH MỤC BẢNG**

Bảng 1.1: Đánh giá các Thuật toán Optimizer 11

CHƯƠNG 1 – TÌM HIỂU, SO SÁNH CÁC PHƯƠNG PHÁP OPTIMIZER

* 1. Optimizer là gì

Trong việc xử lí những vấn đề học máy, có rất nhiều người đã và đang sử dụng những thuật toán tối ưu mà không hề hay biết. Vậy tối ưu (optimizer) là gì? Tối ưu là một loại thuật toán hay một phương thức được xây dựng với mục đích học được các feature (hay pattern) của các dữ liệu đầu vào chẳng hàn như là weight (trọng số), tốc độ học (learning rate) để giảm thiểu mất mát

* 1. Các loại Optimizer

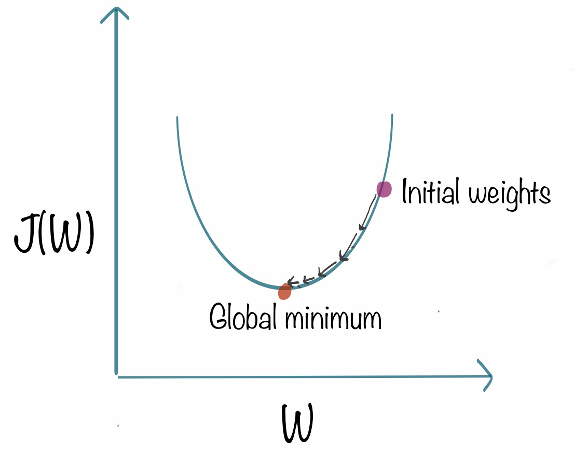
Dựa trên sự thay đổi của weight và learning rate trong neuron network trong quá trình xử lí mất mát sẽ được định nghĩa theo sự tối ưu mà ta sử dụng. Sau đây là một số thuật toán tối ưu phổ biến hiện nay:

* + 1. Gradient Descent

Gradient descent là một trong những thuật toán cơ bản những đồng thời cũng là thuật toán phổ biến nhất, thường được dùng trong các bài toán phân loại (classification) và hồi quy (regression). Ngoài ra lan truyền ngược (backpropagation) cũng sử dụng gradiend descent.

Để chi tiết hơn, thuật toán gradient descent là một thuật toán cơ bản được xây dựng dựa trên việc đạo hàm của hàm mất mát. Tại đó, nó sẽ thực hiện điều chỉnh trọng số truyền vào sao cho hàm có thể đạt giá trị cực tiểu. Hàm này được định nghĩa bằng công thức như sau

Công thức này được tạo ra để cập nhật lại nghiệm sau mỗi vòng lặp và dáu trừ ở đây tượng trung cho việc ngược hướng đạo hàm để có thể xác định được nghiệm sẽ di chuyển theo hướng nào.

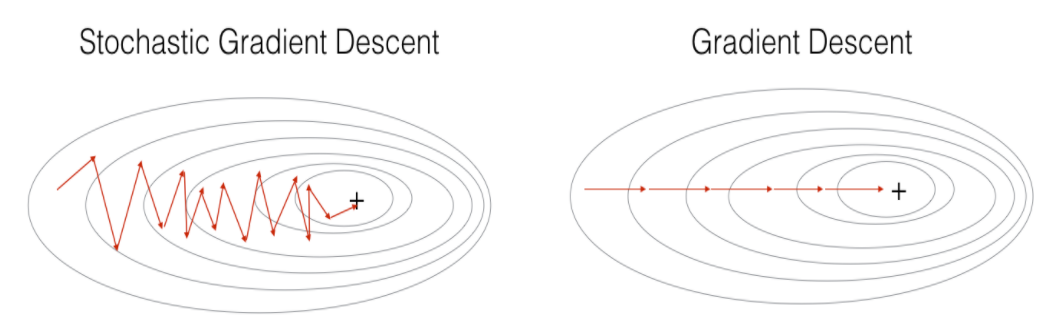


Hình 1.1: Cách Gradient Descent hoạt động

* + 1. Stochastic Gradient Descent

Stochastic có thể coi là một biến thể của gradient descent. Thay vì sau mỗi epoch sẽ thực hiện cập nhật trọng số một lần thì đồi với thuật toán này nếu mỗi epoch có N điểm dữ liệu thì sẽ tiến hành cập nhật trọng số N lần. Vì vậỵ, nếu trong tập dữ liệu có 1000 dòng dữ liệu thì thuật toán sẽ cập nhật tham số 1000. Sau đây là công thức của hàm tối ưu này:

với {x(i), y(i)} là một điểm trên tập dữ liệu



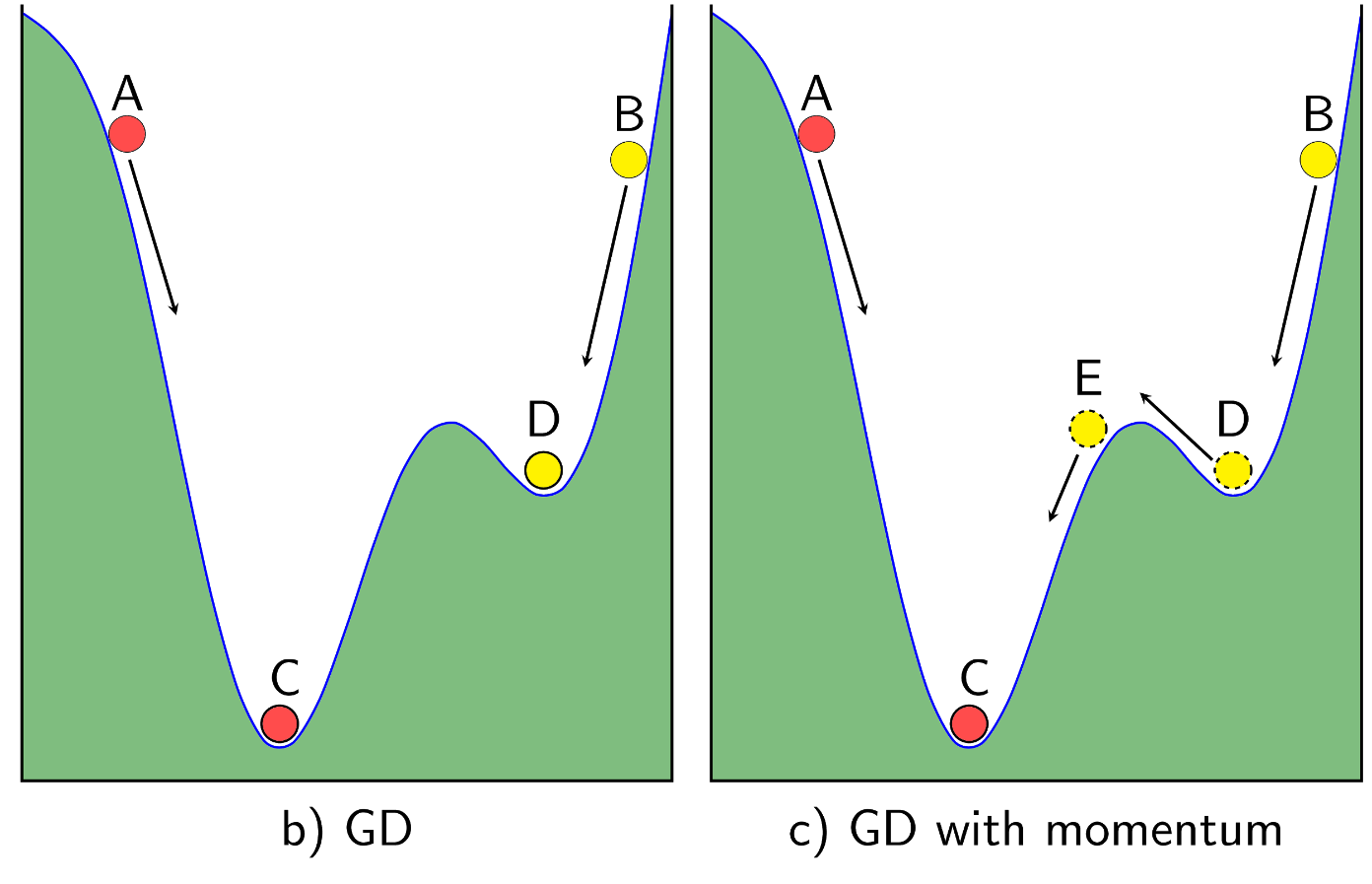
Hình 1.2: So sánh giữa SGD và GD

* + 1. Mini-Batch Gradient Descent

Mini-Batch có thể được xem là một biến thể của thuật toán GD. Được xem như là sự cải tiến của cả SGD và GD. Thuật toán sẽ cập nhật tham số mỗi batch. Vì thế, tập dữ liệu được chia thành từng batch và sẽ tham số sẽ được cập nhật sau mỗi batch. Sau đây là công thức của hàm này  
với B(i) là batch của tập dữ liệu

* + 1. Momentum

Momentum là thuật toán được xây dựng dựa trên việc giảm đi phương sai lớn của SGD và làm giả=m giới hạn của tiến trình.



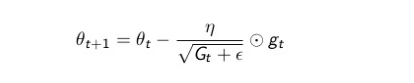
Hình 1.3: So sánh GD và Momentum

Trong thuật toán momentum, ta có thể hiểu dưới góc độ vật lí như sau; Như hình b, khi ta thả 2 viên bi ở lần lượt hai điểm khác nhau là A và B thì viên bi tại A sẽ đến được (cực tiểu toàn cục) còn trong khi viên bi thả tại B chỉ dừng ở D (cực tiểu cục bộ) vốn là điểm mà ta không mong muốn. Vì vậy ta sẽ phải cung cấp cho viên bi một vận tốc ban đầu đủ lớn để viên bi có thể đạt đến điểm C và vượt qua điểm E. Và đây chính là ý tưởng cho thuật toán momentum. Sau đây là công thức của hàm này

Trong đó:

* là tọa độ mới
* là tọa độ cũ
* gama: thường là 0.9
* learning\_rate: tốc độ học
* gradient: đạo hàm của f
  + 1. Adagrad

Adagrad hay còn gọi Adaptive Gradient, không giống với các thuật toán trước đó. Thay vì learning rate không hề đổi trong suốt quá trình huấn luyện (learning rate là hằng số) thì Adagrad coi learning rate là một tham số tức là trong lúc huấn luyện dữ liệu, learning rate sẽ biến thiên.

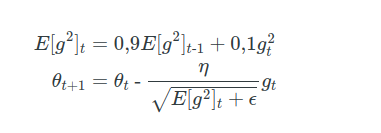


Hình 1.4: Công thức tính Adagrad

Trong đó:

* n: hằng số
* : gradient tại thời điểm t
* hệ số tránh lỗi
* G: Hệ số ma trận chéo mà mỗi phần tử trên đường chéo (i, i) là bình phương của đạo hàm vector
  + 1. RMSProp

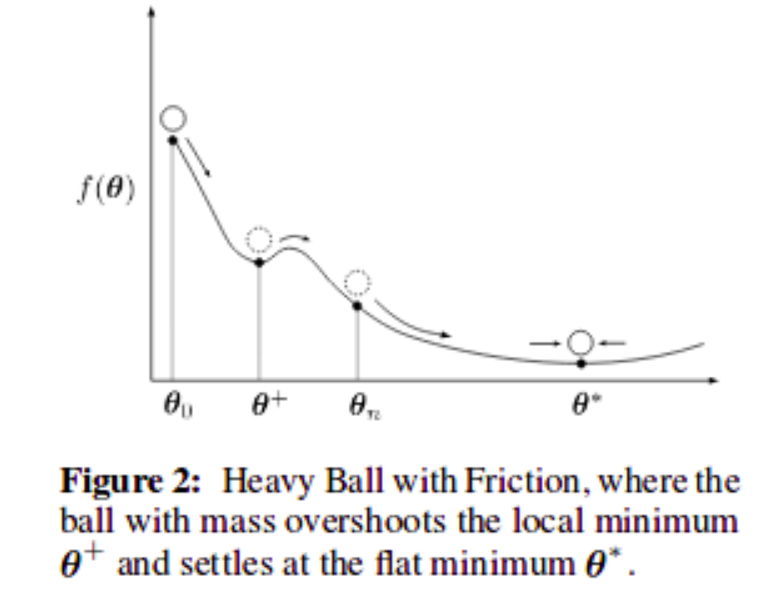
RMSProp hay còn gọi Root Mean Square Propagation, cũng giống như Adagrad là thay đổi các tham số trong quá trình huấn luyện. Nhưng khác với Adagrad thi RMSProp sẽ giải quyết vấn đề tỷ lệ học giảm dần của Adagrad bằng cách chia tỷ lệ học cho trung bình của bình phương gradient



Hình 1.5: Công thức của RMSProp

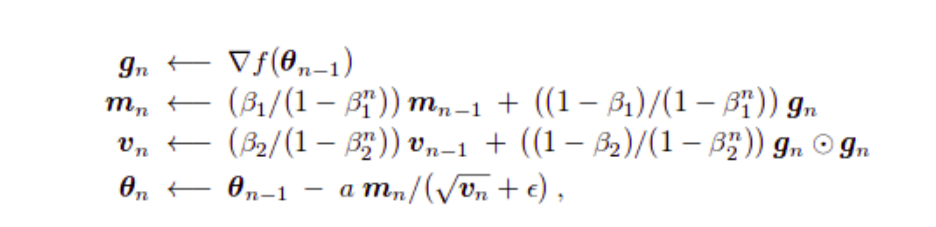
* + 1. Adam

Để có thể nói về Adam thì nó là sự kết hợp của Momentum và RMSProp. Có thể giải thích theo hiện tượng vật lí thì Momentum giống như một quả cầu lao xuống dốc, thì Adam được xem như là một quả cầu rất nặng có ma sát, vì vật nó dễ dàng vượt qua cực tiểu địa phương (local minium) và khi tới cực tiểu toàn cục (global minium) nó không mất nhiều thời gian dao động qua lại quang đích vì nó có ma sart nên dễ dừng lại hơn



Hình 1.6: Mô tả cách hoạt động của Adam

Ta có công thức như sau:



Hình 1.7: Công thức của Adam

* 1. Đánh giá các loại Optimizer

Bảng 1.1: Đánh giá các Thuật toán Optimizer

|  |  |  |
| --- | --- | --- |
| **Loại thuật toán** | **Ưu điểm** | **Nhược điểm** |
| Gradient Descent | * Dễ dàng tính toán * Dễ dàng áp dụng * Dể hiểu | * Có thể bị kẹt ở cực tiểu địa phương * Nếu quá nhiều dữ liệu thì tốn rất nhiều thời gian để đạt được kết quả * Tốn nhiều bộ nhớ |
| Stochastic Gradient Descent | * Dễ dàng đạt được kết quả * Tốn ít bộ nhớ * Có thể đạt được cực tiểu mới | Phương sai cao trong tham số của mô hình  Di chuyển nhanh khỏi cực tiểu toàn cục sau khi đạt được |
| Mini-Batch Gradient Descent | * Xử lí song song một cách hiệu quả * Dễ dàng đạt được kết quả * Tận dụng hiệu quả bộ nhớ | * Khó khăn trong việc chọn learning rate * Khó khăn trong việc chọn kích thước của batch * Mức độ phức tạp cao hơn |
| Momentum | * Giảm sự dao động và phương sai cao của tham số * Dễ dàng đạt được kết quả hơn GD | * Thêm một tham số bậc cao nên cần phải chọn một cách cẩn thận và thường xuyên |
| Adagrad | * Tốc độ học thay đổi dựa trên tham số huấn luyện * Không cần điều chỉnh thường xuyên tốc độ học * Có thể huấn luyện dữ liệu thưa thớt | * Tốn nhiều thời gian tính toán vì cần phải xử lí hàm bậc hai * Cái tốc độ sẽ luôn giảm kết quả trong huấn luyện chậm |
| RMSProp | * Giảm thời gian huấn luyện của mạng neuron * Tránh được overfitting | * Giảm tốc độ hội tụ và khiến learning rate nhạy cảm |

CHƯƠNG 2 – TÌM HIỂU VỀ CONTINUAL LEARNING VÀ TEST PRODUCTION

* 1. Continual Learning là gì

Continual Learning hoặc còn được biết đến là Lifelong Learning, là khả năng một mô hình học máy có thể thích ứng và cập nhật thông tin liên tục theo thời gian khi dữ liệu mới khả dụng. Trong học máy truyền thống, những mô hình thường được huấn luyện trên những tập dữ liệu đã có sẵn, và những kiến thức đo sẽ không đổi sau khi huấn luyện. Vì vậy continual learning được giao cho nhiệm vụ giải quyết thử thách trong việc thích ứng với dữ liệu mới mà không quên đi những kiến thức trước đó.

Trong nội dung của mạng neuron và trí tuệ nhân tạo, continual learning hướng đến việc khiến cho các mô hình có khả năng học hỏi từ những trải nghiệm mới trong khi vẫn giữ nguyên những kiến thức có được từ những công việc trước đó. Điều này cực kì quan trọng trong ứng dựng thực tế khi mà các mô hình phải thích ứng với sự thay đổi của môi trường hoặc sự tiến hóa của việc phân chia dữ liệu.

* 1. Test Production là gì

Test Production, là một quy trình để kiểm tra biểu hiện của mô hình trong học máy hay trí tuệ nhân tạo. Dưới đây là một số cách để kiểm tra:

* *Unit tests*: Kiểm tra những component riêng biệt có nhiệm vụ riêng biệt
* *Intergation tests*: Kiểm tra tính kết hợp của những component riêng biệt
* *System tests*: Kiểm tra thiết kế của hệ thống cho với những kết quả đầu ra dựa trên dữ liệu đầu vào.
* *Acceptance tests*: Kiểm tra để xác nhận những yêu cầu đã đạt được
* *Regression tests*: Kiểm tra dựa trên những lỗi đã thấy trước đó để đảm bảo điều đó không lăp lại.
  1. Ứng dụng trong trong giải quyết bài toán học máy

Để có thể áp dụng Continual Learning trong việc giải quyết những bài toán học máy thì sau đây là một số cách tiếp cận:

* *Replay Buffer*:
  + Bảo trì buffer để lưu những giá trị cũ
  + Sử dụng lại các kiểu dữ liệu từ buffer trong quá trình huấn luyện một cách định kỳ để tránh quên
* *Regularization Techniques*:
  + Sử dụng những phương pháp chính quy hóa như Elastic Weight Consoliation (EWC) hoạc Learning without Forgetting (LwF) để hạn chế những thay đổi trong những tham số quan trọng từ nhiệm vụ trước.
* *Online Learning*:
  + Cài đặt online learning cho những mô hình mà cần cập nhật dữ liệu liên tục khi dữ liệu mới.

Ngoài ra còn một số những cách khác để có thể thực hiện contrinual learning.

Sau khi đã thực hiện được những mô hình trên chúng ta sẽ thực hiện những kiểm tra tên mô hình đã thực hiện dựa trên những phương pháp kiểm tra trên.

CHƯƠNG 3 – TỔNG KẾT

3.1 Kết quả đạt đươc

Dựa trên những công việc mà em đã làm xuyên suốt trong bài báo cáo này, đây là những gì em đạt được:

* Hiểu hơn về các thuật toán Optimzer trong xử lí bài toán học máy và cách nó hoạt động
* Mở rộng kiến thức về Continual Learning cũng như cách nó ứng dựng
* Nhận ra Test Production, và cách ứng dụng nó trong việc kiểm tra mô hình

# TÀI LIỆU THAM KHẢO

**Tiếng Việt**

1. <https://viblo.asia/p/optimizer-hieu-sau-ve-cac-thuat-toan-toi-uu-gdsgdadam-Qbq5QQ9E5D8>

**Tiếng Anh**

1. <https://towardsdatascience.com/optimizers-for-training-neural-network-59450d71caf6.>
2. <https://dev.to/amananandrai/10-famous-machine-learning-optimizers-1e22>
3. [https://www.linkedin.com/pulse/mechanics-pros-cons-machine-learning-optimization-sofia-m%C3%A9ndez#:~:text=The%20advantage%20of%20RMSProp%20is,sensitive%20to%20the%20learning%20rate.](https://www.linkedin.com/pulse/mechanics-pros-cons-machine-learning-optimization-sofia-m%C3%A9ndez%23:~:text=The%20advantage%20of%20RMSProp%20is,sensitive%20to%20the%20learning%20rate.)
4. <https://www.datacamp.com/blog/what-is-continuous-learning>
5. <https://madewithml.com/courses/mlops/testing/>