TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP GIỮA KỲ**

**NHẬP MÔN HỌC MÁY**

*Người hướng dẫn*: **GV LÊ ANH CƯỜNG**

*Người thực hiện*: **NGUYỄN DƯƠNG BÌNH AN – 52000734**

**BÙI TUẤN HƯNG – 52000761**

**LƯU TRẦN NGỌC ANH – 51800839**

Nhóm **: 2**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP GIỮA KỲ**

**NHẬP MÔN HỌC MÁY**

*Người hướng dẫn*: **GV LÊ ANH CƯỜNG**

*Người thực hiện*: **NGUYỄN DƯƠNG BÌNH AN – 52000734**

**BÙI TUẤN HƯNG – 52000761**

**LƯU TRẦN NGỌC ANH – 51800839**

Nhóm **: 2**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

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

MỤC LỤC

[MỤC LỤC 1](#_Toc116836668)

[Bài 1: (5 điểm) 2](#_Toc116836669)

[1.1. Mô tả dữ liệu: 2](#_Toc116836670)

[1.2. Thông tin các thuộc tính trong bộ dữ liệu: 2](#_Toc116836671)

[Bài 2: (3 điểm) 4](#_Toc116836672)

[Bài 3 (2 điểm): 5](#_Toc116836673)

[3.1. Gradient Descent 5](#_Toc116836674)

[3.2. Thuật toán Stochastic Gradient Descent 5](#_Toc116836675)

[a. *Thứ tự lựa chọn điểm dữ liệu* 6](#_Toc116836676)

[*b.* *Ưu điểm:* 6](#_Toc116836677)

[*c.* *Nhược điểm:* 6](#_Toc116836678)

[3.3. Thuật toán Adam 6](#_Toc116836679)

[*a.* *Tổng quan về Adam* 6](#_Toc116836680)

[Adam là gì 6](#_Toc116836681)

[Adam sinh ra từ đâu 7](#_Toc116836682)

[Adagrad 7](#_Toc116836683)

[RMSProp - Root Mean Square Root 7](#_Toc116836684)

[*b.* *Thuật toán Adam* 8](#_Toc116836685)

[*c.* *Tóm lại* 9](#_Toc116836686)

[TÀI LIỆU THAM KHẢO 10](#_Toc116836687)

# Bài 1: (5 điểm)

*Giải quyết một bài toán phân loại (classification) trong học máy với các yêu cầu sau:*

* *Tự đặt ra hoặc tự tìm Bài toán với dữ liệu có sẵn hoặc dữ liệu tự xây dựng. Dữ liệu phải là dạng có cấu trúc, tức là dạng bảng biểu với cột là thuộc tính (attribute) và mỗi dòng là một đối tượng (instance).*

*Dữ liệu phong phú với nhiều thuộc tính, các thuộc tính thuộc nhiều kiểu data khác nhau (numerical, categorical). Số lượng các phần tử (các dòng) càng nhiều càng tốt.*

* *Thực hiện các bước đọc dữ liệu, chuẩn hoá dữ liệu trước khi đưa vào mô hình để học.*
* *Sử dụng ít nhất 3 mô hình phân loại khác nhau. So sánh các mô hình này với các độ đo: accuracy, precision, recall, f1-score của từng class và weighted average of f1-score của toàn bộ dữ liệu. So sánh về thời gian training và thời gian testing của các models này.* 
  1. **Mô tả dữ liệu:**

Tập dữ liệu được lấy từ chiến dịch tiếp thị trực tiếp qua điện thoại của một tổ chức ngân hàng ở Bồ Đào Nha.

* 1. **Thông tin các thuộc tính trong bộ dữ liệu:**

**Input variables:**

***# bank client data:***

1 - age (numeric)

2 - job : type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')

3 - marital : marital status (categorical: 'divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)

4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')

5 - default: has credit in default? (categorical: 'no','yes','unknown')

6 - housing: has housing loan? (categorical: 'no','yes','unknown')

7 - loan: has personal loan? (categorical: 'no','yes','unknown')

***# related with the last contact of the current campaign:***

8 - contact: contact communication type (categorical: 'cellular','telephone')

9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')

10 - day\_of\_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')

11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

***# other attributes:***

12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)

13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)

14 - previous: number of contacts performed before this campaign and for this client (numeric)

15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')

***# social and economic context attributes***

16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)

17 - cons.price.idx: consumer price index - monthly indicator (numeric)

18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)

19 - euribor3m: euribor 3 month rate - daily indicator (numeric)

20 - nr.employed: number of employees - quarterly indicator (numeric)

**Output variable (desired target):**

21 - y - has the client subscribed a term deposit? (binary: 'yes','no')

# Bài 2: (3 điểm)

*Feature Selection là bài toán lựa chọn các đặc trưng (feature or attribute) quan trọng và loại bỏ các đặc trưng không quan trọng hoặc dư thừa. Từ đó ta có thể xây dựng mô hình học máy hiệu quả hơn (nhanh hơn, ít tham số hơn và độ chính xác cũng tương tự hoặc thậm chí tốt hơn khi dùng toàn bộ tập feature).*

*Hãy tìm hiểu vấn đề Feature Selection với các yêu cầu sau:*

1. *Phương pháp dựa trên Correlation. Hiện thị các đồ thị minh hoạ.*
2. *Thử nghiệm các tập feature được lựa chọn khác nhau từ phương pháp ở câu (a) cho một bài toán Regression và sử dụng thuật toán Linear Regression. Dữ liệu và bài toán tự chọn. So sánh thông qua độ đo Mean Absolute Error (MAE).*

Giả sử rằng các điểm dữ liệu có số features khác nhau (do kích thước dữ liệu khác nhau hay do một số feature mà điểm dữ liệu này có nhưng điểm dữ liệu kia lại không thu thập được), và số lượng features là cực lớn. Chúng ta cần chọn ra một số lượng nhỏ hơn các feature phù hợp với bài toán (feature selection). Việc này giúp mô hình của chúng ta xử lí nhanh hơn và cũng như cho ra những dự đoán thậm chí chính xác hơn.

Hệ số tương quan Pearson (được đặt theo tên của Karl Pearson) có thể được sử dụng để tóm tắt độ tương quan của mối quan hệ tuyến tính giữa hai tập dữ liệu.

Hệ số tương quan Pearson được tính bằng hiệp phương sai của hai tập dữ liệu chia cho tích của độ lệch chuẩn của mỗi tập dữ liệu. Đây là sự chuẩn hóa hiệp phương sai giữa hai biến để đưa ra một điểm số có thể giải thích (an interpretable score).

Ảnh có chứa văn bản

Mô tả được tạo tự động

Hệ số tương quan Pearson là một thước đo chỉ phản ánh mối tương quan tuyến tính giữa các tập dữ liệu.

# Bài 3 (2 điểm):

*Tìm hiểu các thuật toán tối ưu (thuật toán học, cập nhật tham số) trong học máy. Mô tả, giải thích thuật toán và có code minh hoạ:*

* *Thuật toán Stochastic Gradient Descent*
* *Thuật toán Adam (Adam Optimization Algorithm)*

1. **Gradient Descent**

Gradient Descent là một kỹ thuật tối ưu hóa phổ biến trong ML và Học sâu và nó có thể được sử dụng với hầu hết, nếu không phải tất cả, các thuật toán học tập. Gradient là độ dốc của một hàm. Nó đo lường mức độ thay đổi của một biến để đáp ứng với những thay đổi của một biến khác. Về mặt toán học, Gradient Descent là một hàm lồi có đầu ra là đạo hàm riêng của một tập các tham số đầu vào của nó. Gradient càng lớn thì độ dốc càng lớn.

Bắt đầu từ một giá trị ban đầu, Gradient Descent được chạy lặp đi lặp lại để tìm các giá trị tối ưu của các tham số để tìm giá trị nhỏ nhất có thể có của chi phí hàm đã cho.

Có ba loại Gradient Descent:

* Batch Gradient Descent
* Stochastic Gradient Descent
* Đổ dốc màu theo lô nhỏ

Trong đồ án này em sẽ tìm hiểu về Stochastic Gradient Descent (SGD).

1. **Thuật toán Stochastic Gradient Descent**

Trong thuật toán này, tại 1 thời điểm, ta chỉ tính đạo hàm của hàm mất mát dựa trên chỉ một điểm dữ liệu xixi rồi cập nhật θ dựa trên đạo hàm này. Việc này được thực hiện với từng điểm trên toàn bộ dữ liệu, sau đó lặp lại quá trình trên. Thuật toán rất đơn giản này trên thực tế lại làm việc rất hiệu quả.

Mỗi lần duyệt một lượt qua tất cả các điểm trên toàn bộ dữ liệu được gọi là một epoch. Với GD thông thường thì mỗi epoch ứng với 1 lần cập nhật θ, với SGD thì mỗi epoch ứng với N lần cập nhật θ với N là số điểm dữ liệu. Nhìn vào một mặt, việc cập nhật từng điểm một như thế này có thể làm giảm đi tốc độ thực hiện 1 epoch. Nhưng nhìn vào một mặt khác, SGD chỉ yêu cầu một lượng epoch rất nhỏ (thường là 10 cho lần đầu tiên, sau đó khi có dữ liệu mới thì chỉ cần chạy dưới một epoch là đã có nghiệm tốt). Vì vậy SGD phù hợp với các bài toán có lượng cơ sở dữ liệu lớn và các bài toán yêu cầu mô hình thay đổi liên tục( online learning).

1. ***Thứ tự lựa chọn điểm dữ liệu***

Một điểm cần lưu ý đó là: sau mỗi epoch, chúng ta cần shuffle (xáo trộn) thứ tự của các dữ liệu để đảm bảo tính ngẫu nhiên. Việc này cũng ảnh hưởng tới hiệu năng của SGD.

Một cách toán học, quy tắc cập nhật của SGD là:



trong  là hàm mất mát với chỉ 1 cặp điểm dữ liệu (input, label) là .

**Chú ý:** chúng ta hoàn toàn có thể áp dụng các thuật toán tăng tốc GD như Momentum, AdaGrad,… vào SGD.

1. *Ưu điểm:*

Thuật toán giải quyết được đối với cơ sở dữ liệu lớn mà GD không làm được. Thuật toán tối ưu này hiên nay vẫn hay được sử dụng.

1. *Nhược điểm:*

Thuật toán vẫn chưa giải quyết được 2 nhược điểm lớn của gradient descent ( learning rate, điểm dữ liệu ban đầu ). Vì vậy ta phải kết hợp SGD với 1 số thuật toán khác như: Momentum, AdaGrad,..Các thuật toán này sẽ được trình bày ở phần sau.

1. **Thuật toán Adam**
   1. *Tổng quan về Adam*

#### Adam là gì

Adam (Adaptive Moment Estimation) là một thuật toán tối ưu hóa các hàm mục tiêu ngẫu nhiên dựa trên gradient bậc nhất và sự ước tính thích ứng của các mô-men (moment) bậc thấp.

Một phương pháp tối ưu hóa ngẫu nhiên hiệu quả chỉ yêu cầu gradient bậc nhất với bộ nhớ thấp. Phương pháp này tính toán từng tỷ lệ học thích ứng cho các tham số khác nhau.

Adam sử dụng ước tính của khoảng thời gian thứ nhất và thứ hai để điều chỉnh tốc độ học cho mỗi trong số của mạng nơ-ron.

#### Adam sinh ra từ đâu

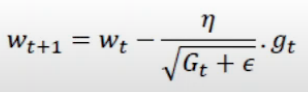
Là sự kết hợp những lợi thế của hai thuật toán tối ưu AdaGrad và RMSProp nhằm giảm dần độ dốc ngẫu nhiên.

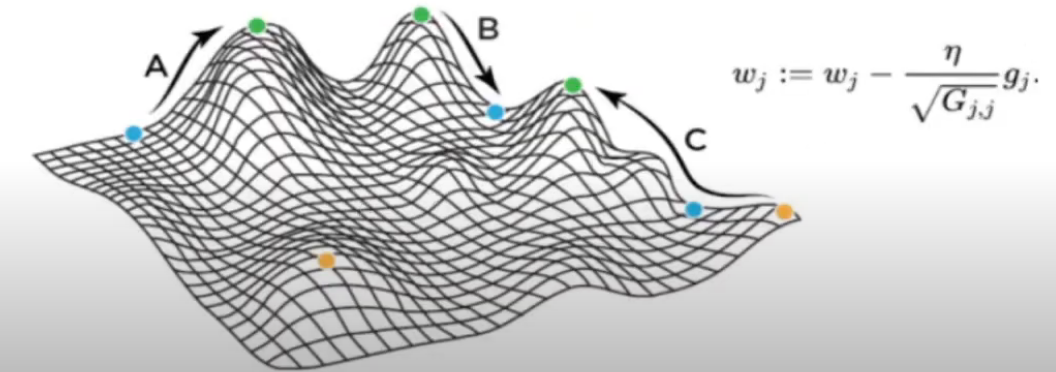
Adam thay vì điều chỉnh các tham số learning rate dựa trên thời điểm trung bình đầu tiên (giá trị trung bình) như RMSProp, Adam sử dụng giá trị trung bình của thời điểm thứ hai của các gradient (Phương sai không tập trung)

#### Adagrad

Thực hiện giảm dần độ dốc bằng cách thay đổi tốc độ học tập

Được cải thiện hơn bằng cách cho trong số học tập chính xác dựa vào đầu vào trước nó để tự điều chỉnh tỉ lệ học theo hướng tối ưu nhất thay vì với một tỉ lệ học duy nhất cho tất cả các nút.



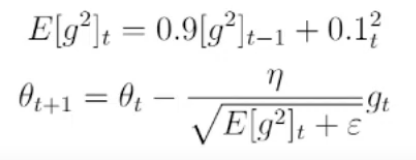


#### RMSProp - Root Mean Square Root

RMSProp là phương pháp điều chỉnh tỷ lệ học (learning rate). Sử dụng trung bình bình phương của gradient để chuẩn hóa gradient

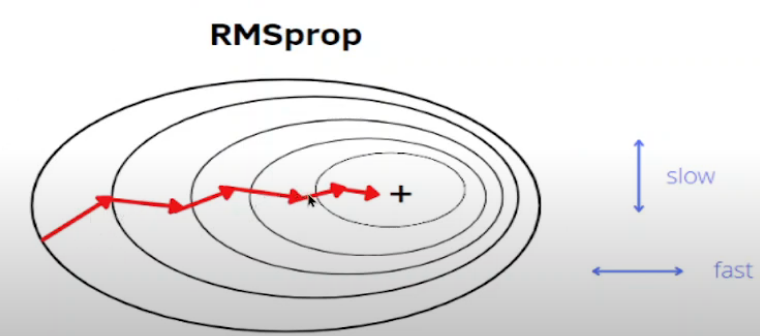
Có tác dụng cân bằng kích thước bước - giảm bước cho độ dốc lớn để tránh hiện tượng phát nổ độ dốc (Exploding Gradient), và tăng bước cho độ dốc nhỏ để tránh biến mất độ dốc (Vanishing Gradient).

RMSProp tự động điều chỉnh tốc độ học tập, và chọn một tỉ lệ học tập khác nhau cho mỗi tham số.



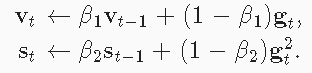
RMSProp cũng chia learning rate cho E[g2]t: Hinton đề nghị γ = 0.9 và giá trị mặc định tốt cho

η = 0.001

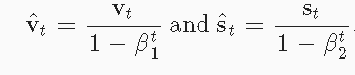


### *Thuật toán Adam*

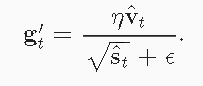
Một trong những thành phần chính của Adam là các trung bình động trọng số mũ (hay còn được gọi là trung bình rò rỉ) để ước lượng cả động lượng và mô-men bậc hai của gradient. Cụ thể, nó sử dụng các biến trạng thái



Ở đây β1 và β2 là các tham số trọng số không âm. Các lựa chọn phổ biến cho chúng là  β1=0.9 và  β2=0.999. Điều này có nghĩa là ước lượng phương sai di chuyển *chậm hơn nhiều* so với số hạng động lượng. Lưu ý rằng nếu ta khởi tạo  v0=s0=0, thuật toán sẽ có độ chệch ban đầu đáng kể về các giá trị nhỏ hơn. Vấn đề này có thể được giải quyết bằng cách sử dụng   để chuẩn hóa lại các số hạng. Tương tự, các biến trạng thái được chuẩn hóa như sau



Với các ước lượng thích hợp, bây giờ chúng ta có thể viết ra các phương trình cập nhật. Đầu tiên, chúng ta điều chỉnh lại giá trị gradient, tương tự như ở RMSProp để có được



Không giống như RMSProp, phương trình cập nhật sử dụng động lượng v^t thay vì gradient. Hơn nữa, có một sự khác biệt nhỏ ở đây: phép chuyển đổi được thực hiện bằng cách sử dụng thay vì . Trong thực tế, cách đầu tiên hoạt động tốt hơn một chút, dẫn đến sự khác biệt này so với RMSProp. Thông thường, ta chọn ϵ=10−6 để cân bằng giữa tính ổn định số học và độ tin cậy.

Bây giờ chúng ta sẽ tổng hợp lại tất cả các điều trên để tính toán bước cập nhật. Có thể bạn sẽ thấy hơi tụt hứng một chút vì thực ra nó khá đơn giản



Khi xem xét thiết kế của Adam, ta thấy rõ nguồn cảm hứng của thuật toán. Động lượng và khoảng giá trị được thể hiện rõ ràng trong các biến trạng thái. Định nghĩa khá kì lạ của chúng đòi hỏi ta phải giảm độ chệch của các số hạng (có thể được thực hiện bằng cách tinh chỉnh một chút phép khởi tạo và điều kiện cập nhật). Thứ hai, việc kết hợp của cả hai số hạng trên khá đơn giản, dựa trên RMSProp. Cuối cùng, tốc độ học tường minh η cho phép ta kiểm soát độ dài bước cập nhật để giải quyết các vấn đề về hội tụ.

* 1. *Tóm lại*

Adam kết hợp các kỹ thuật của nhiều thuật toán tối ưu thành một quy tắc cập nhật khá mạnh mẽ.

Dựa trên RMSProp, Adam cũng sử dụng trung bình động trọng số mũ cho gradient ngẫu nhiên theo minibatch.

Adam sử dụng phép hiệu chỉnh độ chệch (*bias correction*) để điều chỉnh cho trường hợp khởi động chậm khi ước lượng động lượng và mô-men bậc hai.

**TÀI LIỆU THAM KHẢO**

1. Code bài giảng
2. <https://archive.ics.uci.edu/ml/datasets/Bank+Marketing>
3. <https://en.wikipedia.org/wiki/Pearson_correlation_coefficient>
4. <https://machinelearningmastery.com/how-to-use-correlation-to-understand-the-relationship-between-variables/?fbclid=IwAR0m_kKTSNY5Hu2NMv2Ol_K3z6mLSQ_1m0oneb0T6ZV6Rt-uLiUfY4XuGgc>
5. <https://machinelearningcoban.com/2017/01/16/gradientdescent2/>
6. <https://cafedev.vn/tu-hoc-ml-stochastic-gradient-descent-sgd/>
7. <https://d2l.aivivn.com/chapter_optimization/adam_vn.html>
8. <https://github.com/akkinasrikar/Machine-learning-bootcamp/tree/master/sgd>
9. <https://github.com/yacineMahdid/artificial-intelligence-and-machine-learning/blob/master/deep-learning-from-scratch-python/Gradient%20Descent%20Optimization%20Algorithms.ipynb>