***SVM (Máy vector hỗ trợ) thuộc loại Supervise learning.***

Sơ lược:

Support Vector Machines (SVM) (máy hỗ trợ hỗ trợ) đã được được đề xuất bởi Vapnik và đồng nghiệp của ông vào những năm 1970. Sau đó nó trở nên nổi tiếng và phổ biến vào những năm 1990.

Ban đầu, SVM là một phương pháp phân loại tuyến tính(linear classification). Nó tìm thấy một siêu phẳng (còn gọi là phân loại tuyến tính(linear classification)) để tách hai lớp dữ liệu.

Đối với phân loại phi tuyến tính mà không có siêu phẳng tách dữ liệu tốt, các kernel funct (hàm nhân) sẽ được được sử dụng.

¨ Các hàm kernel đóng vai trò chuyển đổi dữ liệu thành một không gian khác, trong đó dữ liệu có thể phân tách tuyến tính.

Đôi khi, chúng ta gọi SVM tuyến tính khi không có chức năng kernel được sử dụng. (trên thực tế, SVM tuyến tính sử dụng nhân tuyến tính(linear kernel))

SVM có một lý thuyết mạnh mẽ hỗ trợ hiệu suất của nó.

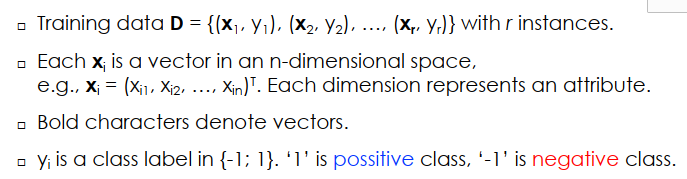
Nó có thể hoạt động tốt với các bài toán có số chiều rất cao.

Bây giờ nó là một trong những phương pháp mạnh mẽ và phổ biến nhất.

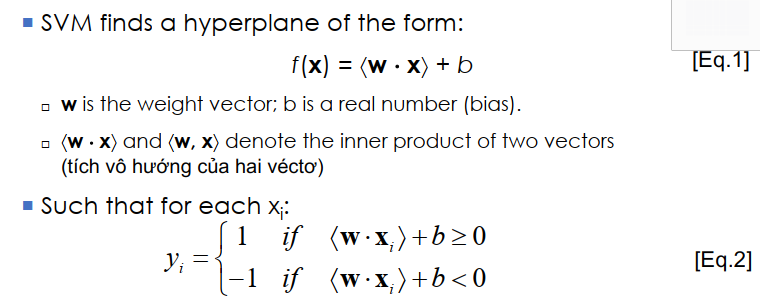
Đối với phân loại văn bản, SVM tuyến tính hoạt động rất tốt.

***The linearly separable case (trường hợp khả phân tuyến tính):***

*Problem representation(Đại diện vấn đề):*



*Linear separability assumption (Giả định về khả năng phân tách tuyến tính):* tồn tại một siêu phẳng (dạng tuyến tính) tách biệt rõ ràng hai lớp (Giả thuyết tồn tại tại một siêu phẳng mà phân tách 2 lớp được)



* ***Eq là viết tắt của phương trình***

Siêu phẳng (H0) phân tách lớp dương và lớp âm có dạng:



Nó còn được gọi là ranh giới/bề mặt quyết định.

Nhưng có thể có vô số cách phân tách siêu phẳng.

Chúng ta nên chọn cái nào?

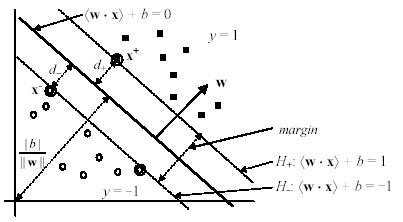
Diagram

Description automatically generated

*Hyperplane with maximum margin (Siêu phẳng với lề tối đa):*

SVM chọn siêu phẳng với lề tối đa. (SVM tìm siêu phẳng mà có lề lớn nhất)

Đã chứng minh được rằng siêu phẳng có biên độ tối đa có sai số tối thiểu trong số tất cả các siêu phẳng có thể.



*Marginal hyperplanes: (Siêu phẳng biên)*

Giả sử rằng hai lớp trong dữ liệu của chúng tôi có thể được tách ra rõ ràng bởi một siêu phẳng.

Biểu thị (x+,1) trong lớp khẳng định và (x-,-1) trong lớp phủ định gần siêu phẳng phân cách nhất H0



Ta định nghĩa hai siêu phẳng biên song song như sau:

Text, letter

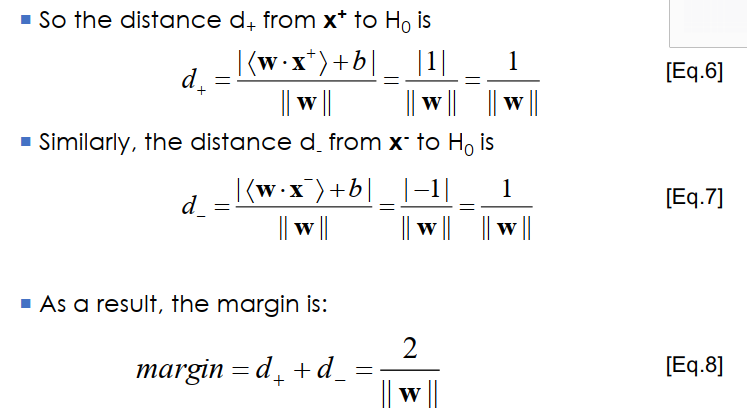
Description automatically generated

* *Crosses: chéo*
* *Parallel: song song*

*Margin (mức lề):* được định nghĩa là khoảng cách giữa hai siêu phẳng biên.

Text, letter

Description automatically generated



***Learning with max margin***

SVM học một bộ phân loại H0 với lề tối đa, tức là, siêu phẳng có lề lớn nhất trong số tất cả

các siêu phẳng có thể.

Nguyên tắc học tập này có thể được trình bày như sau bài toán tối ưu bậc hai:

Graphical user interface, text, application, Word

Description automatically generated

Học SVM tương đương với việc tối thiểu hóa vấn đề sau:

Text, letter

Description automatically generated

Lưu ý, nó có thể được định dạng lại thành:

A picture containing text

Description automatically generated

Đây là một vấn đề tối ưu hóa bị ràng buộc.

Text, letter

Description automatically generated

Xem xét bài toán với các ràng buộc về bất đẳng thức:

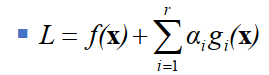
Cực tiểu hóa f(x) với điều kiện gi(x) ≤ 0

Điều kiện cần: có nghiệm x0 thỏa mãn

Text

Description automatically generated

¨ Ở đâu α! ≥ 0 là hệ số nhân Lagrange.

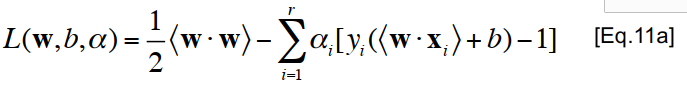


* L được gọi là hàm Lagrange.

¨ x được gọi là primal variable (biến gốc)

được gọi là dual variable (biến đối ngẫu)

Hàm Lagrange cho bài toán [Eq. 10] là:



Trường hợp mỗi αi ≥ 0 là hệ số nhân Lagrange.

Giải [Phương trình. 10] tương đương với minimax sau

vấn đề:

Text

Description automatically generated

Vấn đề nguyên thủy [Eq. 10] có thể suy ra bằng cách giải:

Text

Description automatically generated

Bài toán dual problem (đối ngẫu) của nó có nguồn gốc bằng cách giải:

Text

Description automatically generated with medium confidence

Được biết, giải pháp tối ưu cho [Eq. 10] sẽ đáp ứng một số điều kiện được gọi là Karush-Kuhn-Tucker

(KKT) điều kiện.

Nhìn chung, các điều kiện KKT không đảm bảo tính tối ưu của giải pháp.

May thay, do tính lồi của bài toán nguyên hàm [Eq.10], các điều kiện KKT đều cần thiết và

đủ để đảm bảo tính tối ưu toàn cục của lời giải. Nó có nghĩa là một vectơ thỏa mãn tất cả các điều kiện KKT cung cấp

bộ phân loại tối ưu toàn cục.

¨ Tối ưu hóa lồi là 'dễ dàng' theo nghĩa là chúng ta luôn có thể

tìm một giải pháp tốt với một sự đảm bảo có thể chứng minh được.

¨ Có nhiều thuật toán trong tài liệu, nhưng hầu hết là lặp đi lặp lại.

Trên thực tế, bài toán [Eq.10] khá khó để tìm ra một phương trình hiệu quả thuật toán. Do đó, vấn đề kép của nó là thích hợp hơn

***Karush-kuhn-tucker***

Phương trình cuối cùng [Eq. 16] xuất phát từ một kết quả tốt đẹp từ thuyết nhị nguyên.

¨ Lưu ý: bất kỳ 𝛼! > 0 sẽ ngụ ý rằng điểm liên kết xi nằm trong một siêu phẳng biên (H+ hoặc H-).

¨ Một điểm biên như vậy được đặt tên là một vectơ hỗ trợ.

¨ Một vectơ không hỗ trợ sẽ tương ứng với 𝛼! = 0

A picture containing text

Description automatically generated

***The dual form (Hình thức kép)***

Text, letter

Description automatically generated

* Counterpart: đối tác
* Zeroing: Bỏ …. Đi

Text, letter

Description automatically generated

* Equivalent: tương đương
* Primal: đầu tiên
* Efficient: có hiệu quả
* Iterative: lặp lại
* Complicated: phức tạp

Khi vấn đề đối ngẫu được giải quyết cho 𝜶, chúng ta có thể khôi phục giải pháp tối ưu cho vấn đề [Eq.10] bằng cách sử dụng KKT.

Gọi SV là tập hợp tất cả các vectơ hỗ trợ

¨ SV là tập con của dữ liệu huấn luyện.

¨ anphai > 0 cho thấy xi là một vectơ hỗ trợ.

Chúng ta có thể tính toán w\* bằng cách sử dụng [Eq.12]. Vì thế:



Để tìm b\*, chúng ta lấy chỉ số k sao cho 𝛼k > 0:

A picture containing chart

Description automatically generated

Text, letter

Description automatically generated

* Decision: quyết định
* Boundary: ranh giới
* Instance: ví dụ
* Assigned: giao
* Principle: nguyên tắc
* dot products: tích vô hướng