**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI** (Bold, size 14)

**PHÂN HIỆU TẠI TP. HỒ CHÍ MINH** (Bold, size 14)

**BỘ MÔN CÔNG NGHỆ THÔNG TIN** (Bold, size 14)



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**(Bold, size 15)**

**ĐỀ TÀI: …..…………………………………………......................**

**………………………………………………………………………**

**...........................................................................................................**

**(Bold, size 16)**

Giảng viên hướng dẫn: LÊ NHẬT TÙNG (size 14, chữ hoa)

Sinh viên thực hiện: TRẦN THỊ HOA (size 14, chữ in hoa)

Lớp : ……… (size 14, chữ in hoa)

Khoá :.…........ (size 14, chữ in hoa)

Tp. Hồ Chí Minh, năm 2017 (size 14)

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI** (Bold, size 14)

**PHÂN HIỆU TẠI TP. HỒ CHÍ MINH** (Bold, size 14)

**BỘ MÔN CÔNG NGHỆ THÔNG TIN** (Bold, size 14)



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**(Bold, size 15)**

**ĐỀ TÀI: …..…………………………………………......................**

**………………………………………………………………………**

**...........................................................................................................**

**(Bold, size 16)**

Giảng viên hướng dẫn: LÊ NHẬT TÙNG (size 14, chữ hoa)

Sinh viên thực hiện: TRẦN THỊ HOA (size 14, chữ in hoa)

Lớp : ……… (size 14, chữ in hoa)

Khoá :.…........ (size 14, chữ in hoa)

Tp. Hồ Chí Minh, năm 2017 ( size 14)

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI **CỘNG HÒA XÃ HỘI CHỦ NGHIÃ VIỆT NAM**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH** Độc lập – Tự do – Hạnh phúc

# NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

BỘ MÔN: **CÔNG NGHỆ THÔNG TIN**

-------\*\*\*-------

**Mã sinh viên:** **Họ tên SV:**

**Khóa:** **Lớp:**

1. **Tên đề tài**
2. **Mục đích, yêu cầu**
3. **Nội dung và phạm vi đề tài**
4. **Công nghệ, công cụ và ngôn ngữ lập trình**
5. **Các kết quả chính dự kiến sẽ đạt được và ứng dụng**
6. **Giáo viên và cán bộ hướng dẫn**

Họ tên:

Đơn vị công tác:

Điện thoại: Email:

|  |  |
| --- | --- |
| **Ngày tháng 03 năm 2017**  **Trưởng BM Công nghệ Thông tin** | **Đã giao nhiệm vụ TKTN**  **Giáo viên hướng dẫn** |
| **ThS. Nguyễn Thị Hải Bình** |  |

Đã nhận nhiệm vụ TKTN

Sinh viên: Ký tên:

Điện thoại: Email:

**LỜI CẢM ƠN (size 15, bold)**

*(Cách 1 tab, Time newRoman, 20)*

Size 13

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN (size 15, bold)**

*(Cách 1 tab, Time newRoman, 20)*

Size 13

|  |
| --- |
| ***Tp. Hồ Chí Minh, ngày ….… tháng ….… năm ….…***  **Giáo viên hướng dẫn**  **Lê Nhật Tùng** |

**MỤC LỤC (size 15, bold)**

*(Cách 1 tab, Time newRoman, 20)*

BẮT BUỘC DÙNG MỤC LỤC TỰ ĐỘNG

CHƯƠNG 1: TỔNG QUAN ………………………………………..…….………… 1

1.1. Tổng quan về nợ công…………………………………………………..……....... 1

1.1.1. Nợ công Việt Nam …………………………………………………………..… 2

**DANH MỤC CHỮ VIẾT TẮT**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mô tả** | **Ý nghĩa** | **Ghi chú** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**BẢNG BIỂU, SƠ ĐỒ, HÌNH VẼ (size 15, bold)**

*(Cách 1 tab, Time newRoman, 20, mỗi nội dung trình bày bắt đầu từ 1 trang mới)*

Bảng 1.1: (size 13)…………………………………………………………………........

Bảng 1.2: ..........................................................................................................................

Sơ đồ 1.1:………………………………………………………………………………..

Hình 1.1: ………………………………………………………………………………..

**Ghi chú:**

* + Xếp sau trang Mục lục
  + Chữ số thứ nhất chỉ tên chương
  + Chữ số thứ hai chỉ thứ tự bảng biểu, sơ đồ, hình,…trong mỗi chương
  + Ở cuối mỗi bảng biểu, sơ đồ, hình,…trong mỗi chương phải có ghi chú, giải thích, nêu rõ nguồn trích hoặc sao chụp,…

**CHƯƠNG 1. MỞ ĐẦU**

### Đặt vấn đề

Thị giác máy tính là một lĩnh vực đã và đang được ứng dụng rộng khắp trong hiều mặt của cuộc sống, là kĩ thuật liên quan đến việc nhận dạng và phân biệt đối tượng trong thực tế thông qua ảnh chụp của chúng. Kĩ thuật này ngày càng được áp dụng rộng rãi vào nhiều lĩnh vực của nông nghiệp như tự động hóa, nông nghiệp chính xác, phân loại nông sản, nhận dạng (loài cây, cỏ dại, trái cây...). Thực chất, đa số bài toán của thị giác máy tính là bài toán phân lớp. Chẳng hạn, việc xác định giới tính của một người thông qua ảnh chân dung của người đó, thực chất là cách xác phân lớp nam hoặc nữ, hay nhận dạng biểu cảm của con người: tức giận, vui vẻ , bất ngờ, lo lắng,…

Chính vì vậy, việc ứng dụng các thuật toán Machine learning vào thị giác máy tính ngày càng được chú trọng. Đặt biệt là các bài toán phân lớp. Trong Machine learning có nhiều bài toán phân lớp chẳng hạn như: kNN, Logistic Regression, SVM,… Nhìn chung các ,thuật toán trên ban đầu giải quyết bài toán phân lớp đơn giản (tức là gồm hai lớp - gọi là binary classification ), ngoài trừ thuật toán kNN. Nhưng sau đó, chúng được cải tiến để giải quyết các bài toán đa lớp (Multi-class).

Trong số các thuật toán kể ở trên, SVM (Support Vector Machine) là một thuật toán dễ sử dụng, đơn giản nhưng lại đem lại một kết quả với hiệu suất cao. Thậm chí khi đem so sánh với các thuật toán khác ở trong việc phân lớp (mặc dù SVM vẫn có thể áp dụng cho các bài toán hồi quy, nhưng em cảm thấy nó vẫn thích hợp cho bài toán phân lớp hơn), thì kết quả của SVM có thể gọi là nhỉnh hơn đôi phần.

Trong nội dung báo cáo thực hành chuyên môn lần này em sẽ tiến hành nghiêng cứu về thuật toán SVM, đồng thời sẽ áp dụng nó cho bài toán nhận diện chữ viết tay bằng một demo nhỏ.

### Mục tiêu nghiên cứu

Mục tiêu đề tài là tiến hành nghiên cứu về thuật toán SVM. Hiểu rõ được ý tưởng triển khai của SVM.

* 1. Phạm vi

- Nghiêng cứu về ý tưởng của SVM

- Các cơ sở về toán học

- Phương pháp biên mềm (Soft-margin) nhằm giúp SVM có thể chấp nhận nhiễu

- Phương pháp Kernel trick giải quyết các bài toán phi tuyến tính

### Cấu trúc báo cáo thực tập chuyên môn

* + 1. Chương 1: Mở đầu
    2. Chương 2: Giới thiệu chung về Machine Learning
    3. Chương 3: Thuật toán SVM

**CHƯƠNG 2. GIỚI THIỆU CHUNG VỀ MACHINE LEARNING**

##### 2.1 Tổng quan về trí tuệ nhân tạo

Ngày nay, lĩnh vực AI - Artificial Intelligence (Trí tuệ nhân tao) đang phát triển một cách vũ bão và ngày càng len lỏi vào mọi lĩnh vực trong đời sống. Mục đích chính của lĩnh vực AI là tập trung nghiêng cứu và mô phỏng lại quá trình học tập, suy nghĩ va tư duy của con người để máy có thể học. Các khái niệm này lần đầu tiên được John McCarthy, một nhà khoa học máy tính Mỹ, vào năm 1956 tại Hội nghị The Dartmouth.

Trong lịch sử phát triển AI, các nhà nghiên cứu phân thành 4 hướng tiếp cận chính:

- Hành động như người (acting humanly)

- Suy nghĩ như người (thinking humanly)

- Suy nghĩ hợp lý (thinking rationally)

- Hành động hợp lý (acting rationally)

Trong đó, mức độ mô phỏng máy tính như người là khó nhất và đây cũng là mục tiêu mà các nhà khoa học đang hướng tới. Đến năm 1950, mục tiêu trên được “khai phá” nhờ bài kiểm tra “Turing Test”. Bài test này được Alan Turing đề xuất nhằm mục đích để kiểm tra hệ thống máy tính đã đạt đến khả năng thông minh hay chưa. Bài test tiến hành bao gồm một người đặt câu hỏi, một người trả lời câu hỏi và một máy tính phân tích trả lời câu hỏi. Nếu quá trình trao đổi đó, người đặt câu hỏi không còn phân biệt được đâu là câu trả lời của con người đâu là câu trả lời của máy tính, thì máy tính được xem là thông minh. Từ cách tiếp cận này đã cho ra các nhánh sau đây:

- Natural language processing: xử lý ngôn ngữ tự nhiên

- Computer vision : thị giác máy tính

- Knowledge representation: biểu diễn tri thức

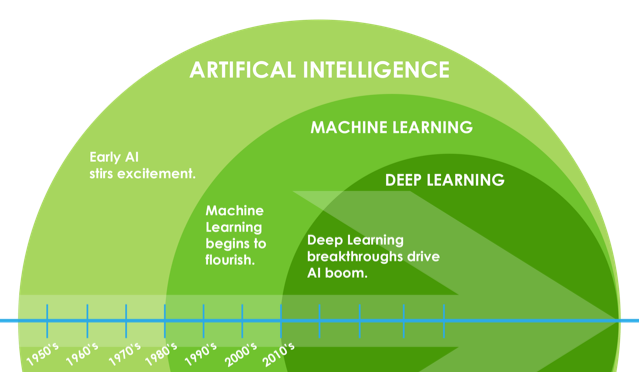
- Machine learnig: máy học

- Automated reasoning: Tự suy luận

- Robotics: Rô bốt học

- ….

##### 2.2 Machine learning (Máy học)



Machine learning là một tập con của trí tuệ nhân tạo. Theo Wikipedia Machine learning được định nghĩa như sau.

“Machine learnig là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kĩ thuật cho phép các hệ thống "học" tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Ví dụ như các máy có thể "học" cách phân loại thư điện tử xem có phải thư rác (spam) hay không và tự động xếp thư vào thư mục tương ứng. Học máy rất gần với suy diễn thống kê (statistical inference) tuy có khác nhau về thuật ngữ.

Học máy có liên quan lớn đến thống kê, vì cả hai lĩnh vực đều nghiên cứu việc phân tích dữ liệu, nhưng khác với thống kê, học máy tập trung vào sự phức tạp của các giải thuật trong việc thực thi tính toán. Nhiều bài toán suy luận được xếp vào loại bài toán NP-khó, vì thế một phần của học máy là nghiên cứu sự phát triển các giải thuật suy luận xấp xỉ mà có thể xử lý được.

Học máy có hiện nay được áp dụng rộng rãi bao gồm máy truy tìm dữ liệu, chẩn đoán y khoa, phát hiện thẻ tín dụng giả, phân tích thị trường chứng khoán, phân loại các chuỗi DNA, nhận dạng tiếng nói và chữ viết, dịch tự động, chơi trò chơi và cử động rô-bốt (robot locomotion).”

Như vậy thông qua định nghĩa trên ta có thể thấy được ML là một lĩnh vực tập trung nghiêng cứu về dữ liệu (data), và từ tập dữ liệu đó (do người lập trình đưa vào học do máy tính tự động thu thập) để từ đó máy tính có thể xử lý các bài toán liên quan.

Phân loại Machine learning. Dựa vào phương thức học ta có thể chia machine learining ra thành 4 loại sau:

- Supervised Learning (Học có giám sát): đây là thuật toán dự đoán đầu ra (outcome) của một dữ liệu mới (new input) dựa trên các cặp (input, outcome) đã biết từ trước. Cặp dữ liệu này còn được gọi là (data, label), tức (dữ liệu, nhãn). Supervised learning là nhóm phổ biến nhất trong các thuật toán Machine Learning. Học có giám sát tiếp tục được chia thành các nhánh khác bao gồm:

+ Các thuât toán Classification (Phân loại): giải quyết các bài toán dạng: “Cái này có dạng là A hay B? ”. Cụ thể là cái bài toán như: xác định giới tính một người, nhận dạng chữ viết tay, xác định email là spam hay không?…. Các thuật toán nổi tiếng trrong nhóm này bao gồm: Logistic Regression, SVM, KNN,….

+ Các thuật toán Regression (Hồi quy): giải quyết các bài toán dạng: “Với dữ liệu như thế này sẽ có kết quả xấp xỉ là bao nhiêu”. Cụ thể là cái bài toán như: “Dự đoán giá nhà với một tập dữ liệu diện tích, khoảng cách đến trung tâm thành phố,….”. Các thuật toán điển hình là: Linear Regression,Stepwise Regression,….

- Học không giám sát: tương tự với học có giám sát, phương pháp học không giám sát cũng tiến hành sát định đầu ra của một dữ liệu mới thông qua tập dữ liệu có sẵn. Tuy nhiên điểm khác biệt là học không giám sát, tập dữ liệu có sẵn (data training) chưa sát định được đầu ra (kết quả). Các bài toán Unsupervised learning được tiếp tục chia nhỏ thành hai loại:

+ Clustering (phân nhóm): Một bài toán phân nhóm toàn bộ dữ liệu thành các nhóm nhỏ dựa trên sự liên quan giữa các dữ liệu trong mỗi nhóm. Ví dụ bài toán xây dựng hệ thống sản phẩm liên quan.

+ Association: Là bài toán khi chúng ta muốn khám phá ra một quy luật dựa trên nhiều dữ liệu cho trước. Ví dụ: những khách hàng nam mua quần áo thường có xu hướng mua thêm đồng hồ hoặc thắt lưng; những khán giả xem phim Spider Man thường có xu hướng xem thêm phim Bat Man, dựa vào đó tạo ra một hệ thống gợi ý khách hàng (Recommendation System), thúc đẩy nhu cầu mua sắm.

- Semi-Supervised Learning (Học bán giám sát): Các bài toán khi chúng ta có một lượng lớn dữ liệu nhưng chỉ một phần trong chúng được gán nhãn (có kết quả sẵn) được gọi là Semi-Supervised Learning. Những bài toán thuộc nhóm này nằm giữa hai nhóm được nêu bên trên.

- Reinforcement Learning (Học tăng cường): là các bài toán giúp cho một hệ thống tự động xác định hành vi dựa trên hoàn cảnh để đạt được lợi ích cao nhất (maximizing the performance). Hiện tại, Reinforcement learning chủ yếu được áp dụng vào Lý Thuyết Trò Chơi (Game Theory), các thuật toán cần xác định nước đi tiếp theo để đạt được điểm số cao nhất.

**CHƯƠNG 3. THUẬT TOÁN SVM**

**3.2 Bài toán tối ưu**

Để có thể tiếp tục bài báo cáo, em xin phép được trình bày về một số khái niệm liên quan đến phương pháp học có giám sát và các khái niệm về bải toán tối ưu.

Đầu tiên, chúng ta sẽ tiến hành tiếp cận phương pháp học có giám sát. Như em đã đề cập trước đó, phương pháp học có giám sát là phương pháp tiến hành dựa phần nhiều vào khoa học dữ liệu. Ta có một tập các dữ liệu đã thu thập từ trước đó chúng bao gồm một tập dữ liệu khác chứa danh sách các thuộc tính, còn lại là một danh sách chứa các đầu ra (outcome) ứng với từng dữ liệu trên. Nhiệm vụ của các thuật toán học có giám sát đó là, từ một đầu vào mới -chúng bao gồm một danh sách các thuộc tính, ta sẽ thu dược một đầu ra ứng với đầu vào đó. Ví dụ với một bài toán dự đoán giá bán của một căn nhà căn cứ vào diện tích, khoảng cách từ căn nhà đến trung tâm thành phố, mức độ an ninh của khu vực xung quanh. Ta có thể nhận thấy, các giá trị của diện tích, khoảng cách, mức độ an ninh,… là các “thuộc tính” tác động trực tiếp đến giá bán của một căn nhà. Như vậy ta có thể hiểu đề bài ở đây chính là, từ một dữ liệu chứa giá trị diện tích, khoảng cách, mức độ an ninh,… hãy tìm một công thức để tính (ước lượng) giá của căn nhà ấy.

Để làm được điều này, nhất là với các bài toán ước lượng, chúng ta cần phải dựa vào một cái gì đó. Lẽ dĩ nhiên trong trường hợp này, thứ chúng ta dựa vào chính là một cơ sở dữ liệu mẫu, nó chứa một danh sách giá trị của các “thuộc tính” diện tích, khoảnh cách, an ninh,… và ứng với mỗi dòng trong danh sách là một cột chứa giá bán căn nhà.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Diện tích (m2) | Khoảng cách(km) | An ninh (mức) | Giá (triệu đồng) |
| 1 | 20 | 12 | 1 | 200 |
| 2 | 22 | 5 | 1 | 400 |
| 3 | 25 | 18 | 1 | 260 |
| 4 | 20 | 12 | 3 | 150 |
| … |  |  |  |  |

Dựa vào dữ liệu trên ta sẽ tiến hành dự đoán một căn nhà có diện tích 30 m2 , khoảng cách đến trung tâm thành phố là 10km, mức độ anh ninh khu vực xung quanh là 1 thì giá căn nhà là bao nhiêu.

Tổng kết lại, mục tiêu chính của bài toán là tiến hành xây dựng một hàm (công thức) để làm được điều đó - ước lượng được giá bán của một căn nhà. Ta có thể thấy rằng, giá của một căn nhà có tỷ lệ thuận với diện tích của nó và đồng thời tỷ lệ nghịch với khoảng cách từ nhà đến trung tâm và mức độ an ninh khu vực (với 1 là mức an toàn nhất), ta có thể đưa ra công thức đơn giản sau:



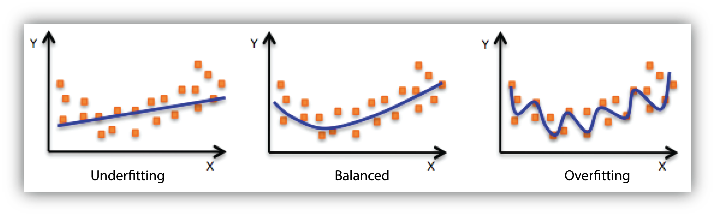
Trong đó, x1 là diện tích, x2 là khoảng cách tới trung tâm và x3 là mức độ an ninh khu vực. Còn các giá trị a1,a2,a3 là các hằng số sẽ được xác định trong quá trình kiểm thử.

Trong phương pháp học có giám sát, các hàm như trên được gọi là các hàm mục giả thuyết. Đấy là nhưng hàm sẽ giải quyết bài toán ban đầu. Hàm giả thuyết sẽ được xây dựng dựa trên các dữ liệu mẫu có sẵn. Các dữ liệu này được gọi là training data (một số tài liệu viết là training set).

Tuy nhiên trên thực tế, quá trình này không đơn giản. Các hàm giả thuyết không thể nào cho ra các đầu ra đúng 100% như trong training data, mà nó chỉ có thể xấp xỉ gần bằng. Để tiến hành đánh giá hàm giả thuyết đưa ra có độ chính xác đên đâu, để còn có thể sửa chữa. Người ta lại dùng tiếp một tập dữ liệu, đã biết trước đầu ra, để kiểm thử. Tập dữ liệu này gọi là test data. Như vậy trong phương pháp học có giám sát chia ra làm hai giai đoạn riêng biệt tách rời nhau đó là : train (tạo hàm giả thuyết, tức tạo model) và test (kiểm tra lại model).

Vậy tại sao chúng ta không gộp hai giai đoạn này thành một. Tức là gộp dữ liệu của tập huấn luyện và tập kiểm thử, rồi sau khi tiến hành xây dựng xong model, ta sẽ dùng nó để kiểm thử ? Nguyên nhân là bởi vì nếu ta dùng cả hai tập dữ liệu cho cả hai quá trình. Ta có thể sẽ gặp trường hợp, mô hình hay hàm giả thuyết được xây dựng quá “khít” với toàn bộ dữ liệu. Nhưng khi ta dùng với dữ liệu thật lại không thể ra kết quả mong đợi. Nói dễ hiểu hơn là giống như tình trạng học vẹt, học tủ ở trường mình (như môn triết học Mác-Lênin). Thầy cho chúng ta 20 bộ đề để ôn. Chúng ta ôn “tủ” 19 đề, và tất cả đều chắc như bắp. Nhưng đến khi thi thì bị “tủ đè” vì đề thì dùng đề số 20.

Trường hợp trên ta gọi là overfitting. Ngược lại với overfitting ta có underfitting, khi mà hàm ta xây dựng không hiệu quả với tập dữ liệu. Hình ảnh dưới đây sẽ cho cái nhìn rõ hơn về cả hai trường hợp.



Hai vấn đè trên đều có cách giải quyết, tuy nhiên nó vượt quá bài báo cáo của em nên em xin tạm thời bỏ qua.

Quay trở lại với phương pháp học có giám sát. Em xin tiếp tục nêu thêm một khái niệm về bài toán tối ưu, hoặc quy hoạch toán học:

***Cho trước***: một hàm f: A → R từ tập hợp A tới tập số thực

***Tìm***: một phần tử x0 thuộc A sao cho f(x0) ≤ f(x) với mọi x thuộc A ("cực tiểu hóa") hoặc sao cho f(x0) ≥ f(x) với mọi x thuộc A ("cực đại hóa").

Trong đó, A của hàm f được gọi là không gian tìm kiếm, thường được xác định bởi một tập các ràng buộc, các đẳng thức hay bất đẳng thức mà các thành viên của A phải thỏa mãn. Các phần tử của A được gọi là các lời giải khả thi, gọi là các đối số tối ưu. Hàm f được gọi là hàm mục tiêu, hoặc hàm chi phí. Lời giải khả thi nào cực tiểu hóa (hoặc cực đại hóa, nếu đó là mục đích) hàm mục tiêu được gọi là lời giải tối ưu.Các bài toán tối ưu thường được ký hiệu như sau:

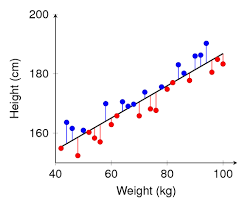
+ Với bài toán tìm max/min:  . Ký hiệu này biểu diễn bài toán tìm giá trị nhỏ nhất của x2  + 2 trên tập số thực R.

+ Đối số tối ưu: . Kí hiệu biểu diễn cho các nghiệm tìm được cho bài toán tìm min của hàm x2 +2, là tập từ (-○○ , -2].

Em đã trình bày xong về các kí hiệu mà em sẽ dùng cho phần tiếp theo của báo cáo. Tuy nhiên có một vấn đề mà em càng làm rõ, đó là giữa các thuật toán học có giám sát và bài toán tối ưu có liên quan gì với nhau. Đầu tiên chúng ta cần nhắc lại rằng, mục tiêu cốt lỗi của các thuật toán học có giám sát là xây dựng được một giả thuyết (model) mà đầu ra của nó là một giá trị cần xác định hay ước lượng như giá bán nhà. Nếu nhìn lại công thức:



Thì ta chỉ cần thay đổi một trọng số trong a0, a1,a2,a3 ,thì ta sẽ có một hàm mới. Điều đó có nghĩa là một bài toán chúng ta còn rất nhiều cách giải khác nhau. Bài toán lại chuyển thành việc tìm tập hợp a. Vấn đề là làm sau ta có thể chọn ra được tập a phù hợp nhất, cho ra kết quả gần sát nhất. Một hướng tiếp cận cho việc này đó là quy đổi nó sang một bài toán tìm giá trị lớn/nhỏ nhất. Ví dụ một bài toán hồi quy, từ cân nặng có thể ước lượng chiều cao của một người:



Rõ ràng ta có thể vẽ vô số các đường thẳng trong đồ thị trên, nhưng đường thẳng phù hợp nhất thì lại khác. Chúng ta có thể xét rằng đường thẳng phù hợp nhất chính là đường thẳng gần với tất cả các điểm dữ liệu nhất. Tức là tổng khoảng cách từ đường thẳng đến các điểm là nhỏ nhất. Khoảng cách ở đây ta có thể hiểu là độ chênh lệch giữa đầu ra của hàm giả thuyết với chiều cao thực. Vậy ta có thể chuyển thành bài toán thành

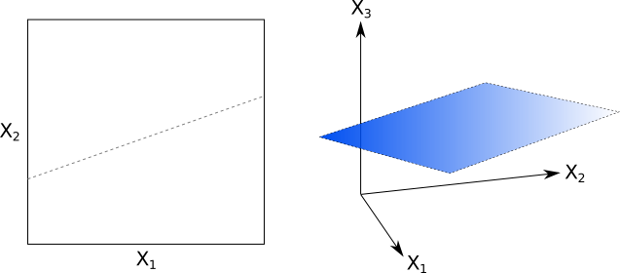
***Tìm***: 

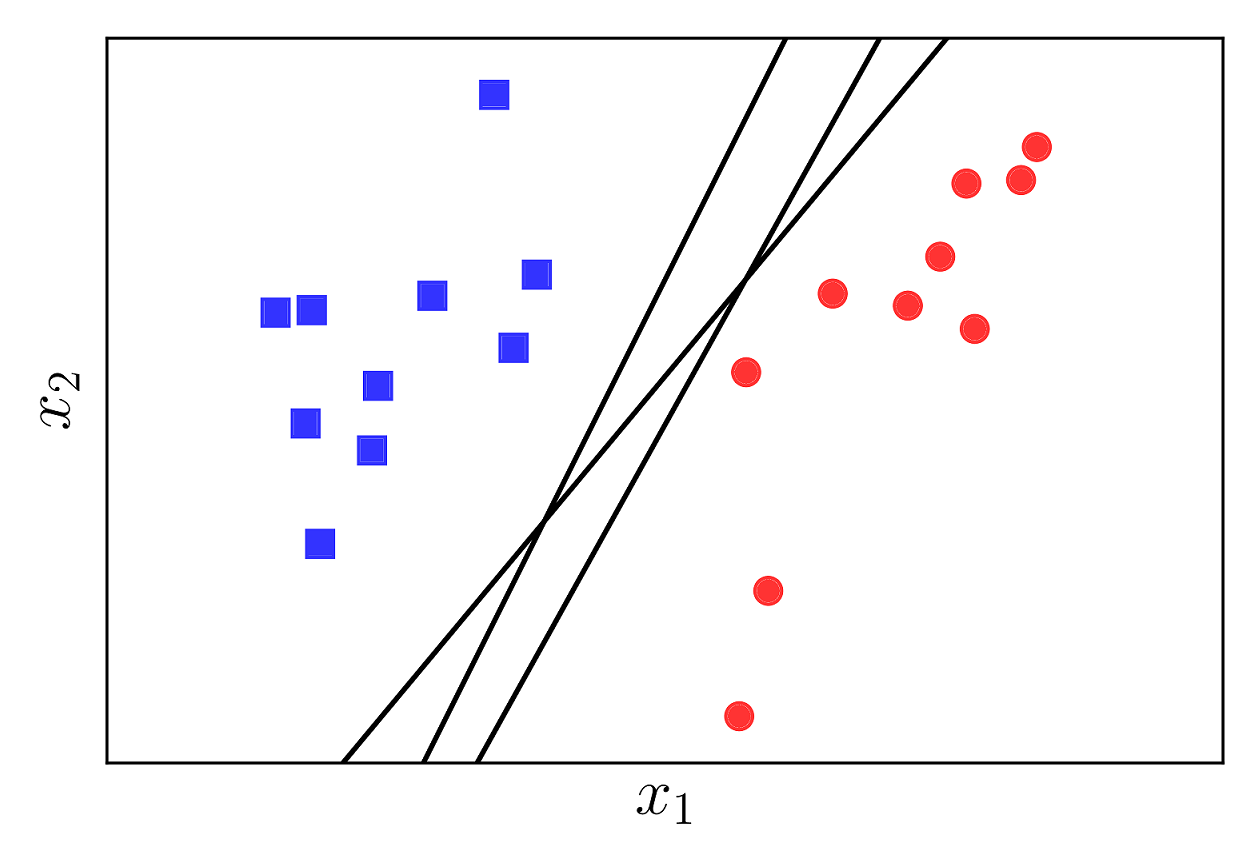
***Sao cho***: 

Trong đó, w là cân nặng, h là chiều cai. Wi và hi là dữ liệu thứ i trong tập dữ liệu, ứng với một điểm trong đồ thị.

**3.2 Thuật toán SVM**

Thuật toán SVM là một trong những thuật toán phân lớp trong phương pháp học có giám sát. Nó giúp việc phân tách hai lớp (class) thành hai phần riêng biệt. Giả sử, ta có hai class được biểu diễn bởi các điểm trong một không gian nhiều chiều. Hai class này phân biệt tuyến tính với nhau, tức là giữa các điểm trong hai tập hợp tồn tại một mặt phẳng/đường thẳng chia không gian gốc thành hai phần, chính là phạm vi của hai class. Mặt phẳng này có số chiều nhỏ hơn không gian gốc. Ví dụ nếu không gian gốc có số chiều là 2 thì thứ chia tách hai lớp là một đường thẳng, với số chiều không gian gốc là 3 thì sẽ là một mặt phẳng, với số chiều là p thì sẽ làm một không gian p-1 chiều.

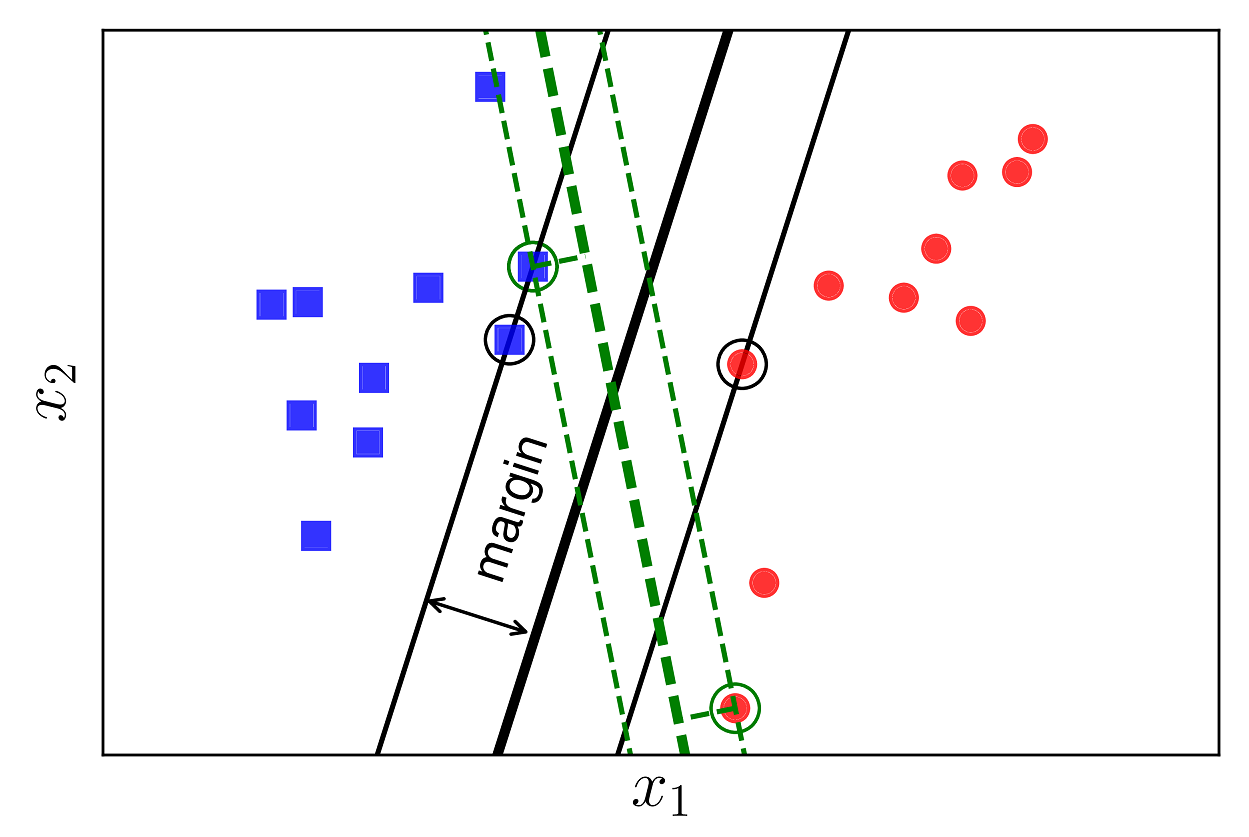
 Quay trở lại với bài toán, ta giả sử hai class của chúng ta được biểu diễn bởi một tập các điểm nằm trên không gian hai chiều. Như vậy sẽ có một đường thẳng tiến hành chia cắt hai class đó. Tuy nhiên, như em đã nói ở mục trước, ta sẽ có vô số các đường có thể đuợc vẽ ra với các hằng số a0,a1,a2,a3,…



Vậy đâu là đường thẳng tốt nhất trong tất cả các đường thẳng trên.

Việc ta xác định đường thẳng phân chia giống như chúng ta xác định đường biên cho phạm vi hai class. Đường biên phải làm sao cho phạm vi của hai class không bị thiệt quá nhiều. Chú ý phạm vi của một class phụ thuộc vào khoảng cáchngắn nhất của điểm thuộc một class đến đường thẳng phân chia. Xét ví dụ ở trên ta nhận thấy có một đường phân chia bị lệch về phía class đỏ (khoảng cách từ điểm gần nhất của class đỏ nhỏ hơn điểm gần nhất của class xanh), nên đường thẳng này không phù hợp. Vậy đường thẳng ta cần tìm phải có khoảng cách tới các điểm gần nhất là tương đương nhau. Thế mới công bằng!

Tuy nhiên, nhiêu đó cũng chưa đủ. Ta xét trường hợp tiếp theo:



Trong hinh ảnh trên tồn tại hai đường thằng lục và đen. Cả hai đường thẳng này đều chia hai class thành hai phần và khoảng cách từ điểm gần nhất đến cả hai đuờng thărng đều bằng nhau. Tuy nhiên, đường thẳng màu đen sẽ là đườg thẳn phù hợp nhất trong truờng hơp này. Lí do bởi vì no có thể chía hai class ra xa nhau nhất có thể. Điều này có lợi cho việc xác định hai class. Cũng như đường biên giới giữa hai quốc gia, chỗ nào hẻo lánh thưa dân sẽ nảy sinh ít vấn đề hơn chỗ biên giới gần các thành phố lớn.

Như vậy ý tưởng của thuật toán SVM đó là xác định một hyperlane của không gian gốc thoản mãn các điều kiện sau:

- Chia cắt hai class thành phần riêng biệt

- Có khoảng cách đến các điểm gần nhất thuộc hai class tương đương nhau.

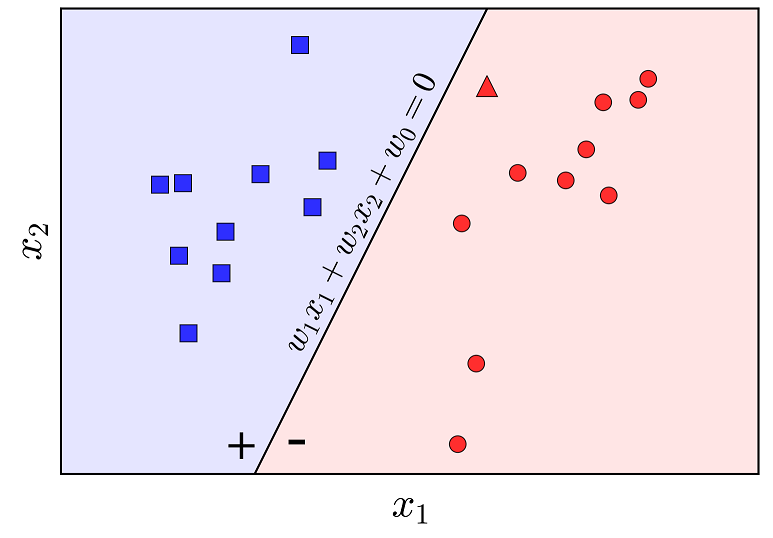
- Chia tách hai class càng xa nhau càng tốt.

3.3 Xây dựng bài toán tối ưu

Giả sử tập các dữ liệu đầu vào cảu một training set là một ma trận hai chiều X = {x0 ,x1 ,x2 ,…xn} với mỗi hàng xi , là một vector biểu diễn một điểm trên đồ thị. Đồng thời, ứng với mỗi dóng, sẽ là một đầu ra yi tương ứng. Chúng ta sẽ gọi chúng là nhãn (label), chúng sẽ được dùng để đánh dấu cho các class. Do ở đây ta chỉ xét trên hai class nên vector y chỉ có hai giá trị: y = -1 cho class màu đỏ, ngược lại y = 1 cho class màu xanh.

Như vậy ta có thể hình thành được công thức ban đầu cho đường biên cho chúng ta:



Ở đây, x1,x1,x2,…,xn là các thuộc tính biểu diễn một điểm dữ liệu tức Xi.(với x0 -bây giờ là 1).Vector W ={w0,w1,w2,….,wn} là danh sách các hằng số được thêm vào. Với không gian thuộc tính là 2 thì hàm này sẽ có dạng là 

Như vậy là chúng ta đã xác định được công thức đường biên cần tìm. Từ đây ta có thể xác định được một diểm dữ liệu mới thuộc lớp nào, với hàm f(X) >= 0 sẽ cho đầu ra là 1 tức thuộc lớp xanh và ngược lại, các điểm cho ra f(X) <0 sẽ thuộc lớp đỏ.

Bước tiếp theo đẻ có thể hoàn thiện công thức, đó là xác định tiếp các hằng số w0,w1,w2,….,wn, chính là vector W. Làm được điểu này chúng ta cần phải nắm bắt lại một số kiến thức sau. Khoảng cách từ một điểm đến một đường thẳng sẽ là :



Trong không gian ba chiều tới một mặt phẳng có phương trình f(X)=w0 + w1x1 + w2x2 +w3x3 được xác định bởi



Ta có thể được tổng quát lên không gian nhiều chiều: Khoảng cách từ một điểm (vector) tới siêu mặt phẳng (hyperplane) có phương trình f(X)=w0 + w1x1 + w2x2 +w3x3 +…+wnxn = WTX được xác định bởi:



Với ||w||2 là chiều dài vector, WT là ma trận chuyển vị của W, X là tập đầu vào (x0 =1).

Áp dụng công thức trên vào bài toán của SVM. Ta nhận thấy một điều giá trị hàm f(x) trên mỗi hàng X luôn cùng dấu với label tương ứng, do đó khi ta nhân chúng lai với nhau kết quả luôn luôn dương. Vì vậy ta hoàn toàn có thế viết lại công thức trên thành.



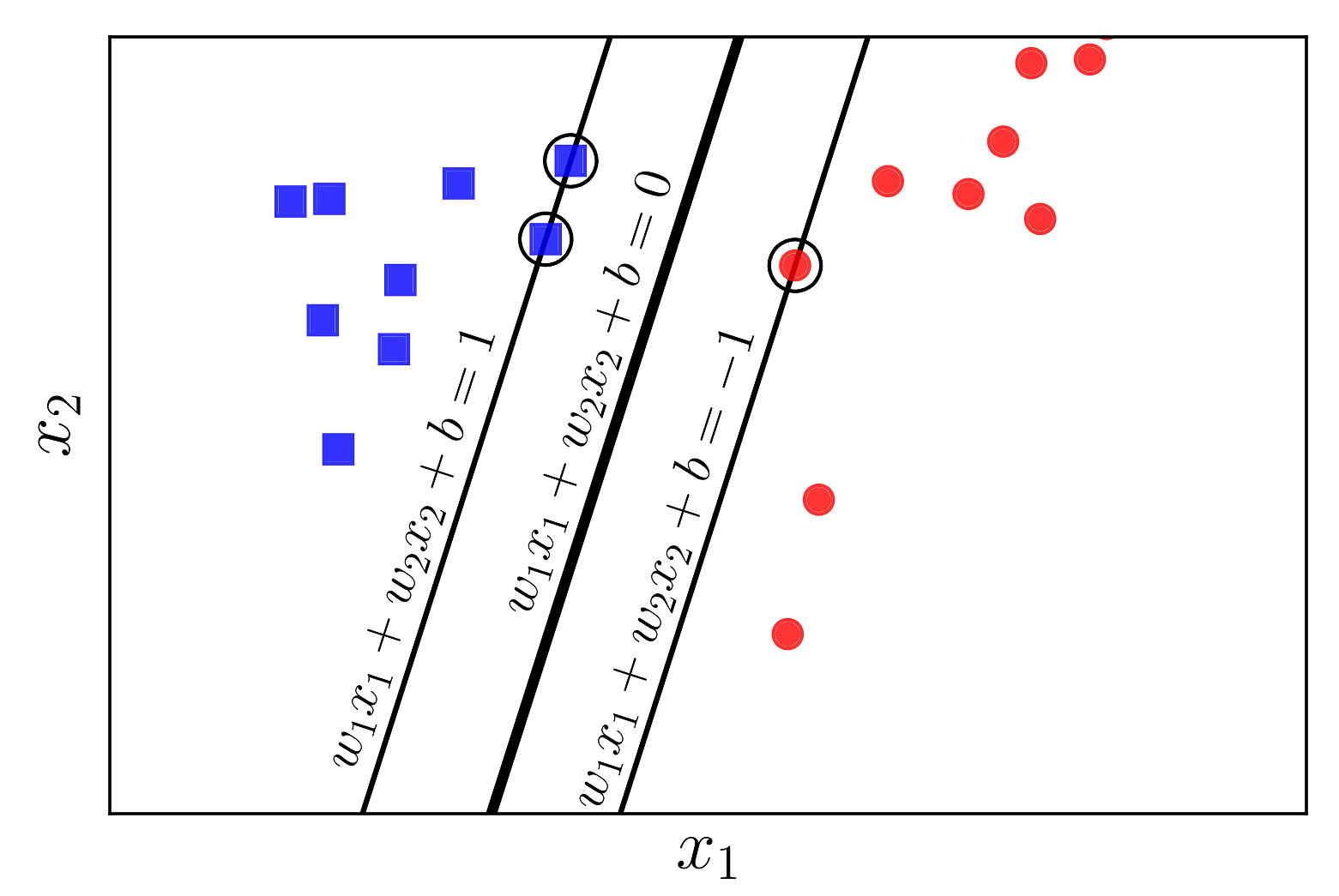
Nhắc lại về ý tưởng của SVM đó là tìm một hyperlane của không gian thuộc tính sao cho khoảng các tới các điểm gần nhất thuộc hai class là bằng nhau và có giá trị là lớns nhất. Khoảng cách này trong SVM gọi là margin và chúng được biểu diễn biểu diễn như sau:



Lời giải cho bài toán tìm W chính là các đối số tối ưu của bài toán trên:



Để có thể đơn giản hóa vấn đề, người ta sẽ giả dụ có các điểm dữ liệu gần nhất nằm ở các đường thẳng song song với đường biên. Lần lượt được biển diễn bởi WTX = -1 và WTX = 1. Chúng ta gọi chúng là các Support Vector. Lưu ý dựa vào hai công thức em vừa đề cập, không có nghĩa là khoảng các của hai Support Vector đến đường biên là 1. Mà nó còn tùy thuộc vào giá trị của W.



Với việc giá trị nhỏ nhất yi(WTX) = 1 ta sẽ có được công thức sau:



Bài toán bây giờ ta nghịch đảo ||w||2 hàm sẽ trở thành tìm min, thêm nhân thêm 1/2 để

Dễ tính đạo hàm.



Việc giải bài toán trên liên quan đến một số lý thuyết toán học tương đối phức tạp như điều kiện Karush-Kuhn-Tucker, hàm đối ngẫu Lagrange, Convex optimization … Em sẽ không nói ở bài báo cáo này.

**PHỤ LỤC**

Phụ lục 1: hướng dẫn cài đặt

Phụ lục 2: hướng dẫn sử dụng

**TÀI LIỆU THAM KHẢO**

1. T.S Lê Phan Thị Diệu Thảo, Nguyễn Thảo Phương, *Nợ công và những vấn đề cần bàn thêm*, Nhà xuất bản Kinh Tế.
2. Boulding, K.E. (1995), *Economics analysis*, Hamish Hamilton, London.
3. <Http://dantri.com.vn/c76/s76-402058/nhung-moc-chinh-cua-khung-hoang-no-chau-au.htm>, “Những mốc chính của khủng hoảng nợ châu Âu”