Báo cáo môn học

HỌC MÁY

Đề tài: Sử dụng SVM dự đoán giá điện thoại



Nhóm 6: Nguyễn Thế Hưởng

Trương Việt Hoàng

Nguyễn Quang Minh

Trần Đức Sơn

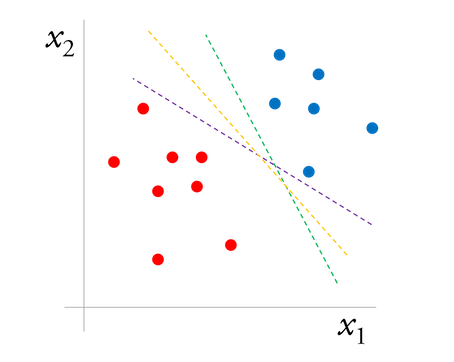
Vũ Duy Hưng

Hà Nội, ngày 27 tháng 11 năm 2020.

1. **Tổng quan về SVM**

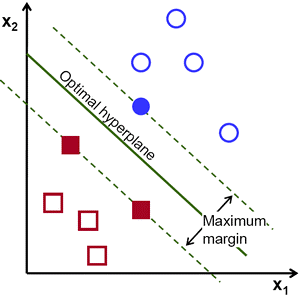
**Support Vector Machine** (SVM) là một thuật toán thuộc nhóm Supervised Learning (Học có giám sát) dùng để phân chia dữ liệu (Classification) thành các nhóm riêng biệt.

Hình dung ta có bộ data gồm các điểm xanh và đỏ đặt trên cùng một mặt phẳng.  
Ta có thể tìm được đường thẳng để phân chia riêng biệt các bộ điểm xanh và đỏ như hình bên dưới.



Margin trong SVM:

Margin là khoảng cách giữa siêu phẳng đến 2 điểm dữ liệu gần nhất tương ứng với các phân lớp.  Điều quan trọng ở đây đó là phương pháp SVM luôn cố gắng cực đại hóa margin này, từ đó thu được một siêu phẳng tạo khoảng cách xa nhất. Nhờ vậy, SVM có thể giảm thiểu việc phân lớp sai (misclassification) đối với điểm dữ liệu mới đưa vào.



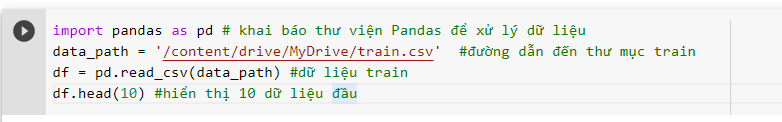
1. **SVM trong mô hình dự đoán giá điện thoại**

Với mô hình dự đoán giá điện thoại nhóm sử dụng thư viện giải SVM là SVC của Scikit-learn trên Google Colab.

Quá trình :

**Bước 1: Xử lý dữ liệu**

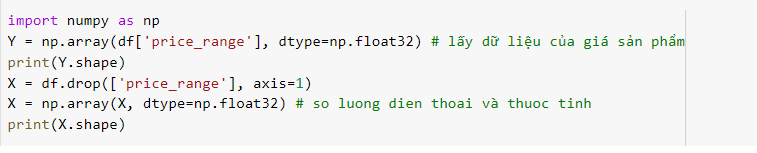
Đọc dữ liệu: Sử dụng thư viện pandas.



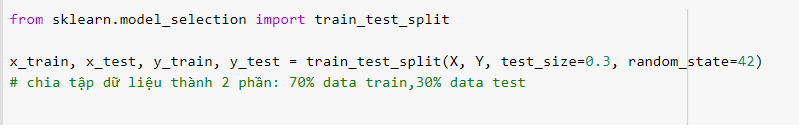
Dữ liệu ban đầu có 2000 mẫu. Mỗi mẫu gồm 21 thuộc tính



Do yêu cầu dự đoán giá thành điện thoại dựa trên các thuộc tính khác có sẵn. Ta tách dữ liệu thành 2 phần Y: giá thành (price\_range) và X : 20 thuộc tính còn lại sử dụng thư viện numpy.

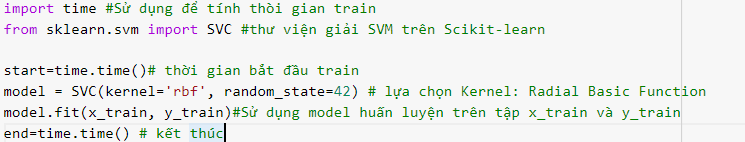


Bộ (X,Y) được chia thành 2 phần: Train và Test (Tỉ lệ 70/30) sử dụng hàm train\_test\_split.



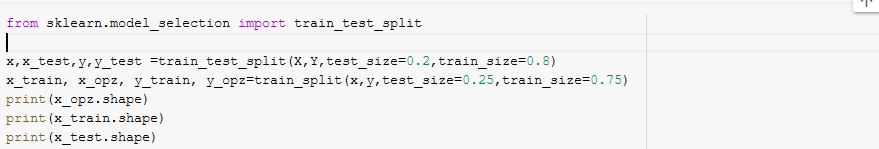
**Bước 2: Train dữ liệu**

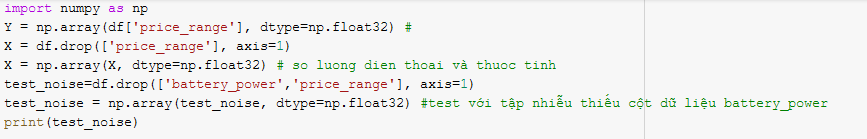
Sử dụng model SVC với kernel : rbf (Lý do chọn được đề cập ở bước 3)



**Bước 3: Tối ưu và đánh giá thuật thuật toán**

Sử dụng một số kernel của svm là: linear, rbf, poly, sigmoid

Chia tập dữ liệu thành 3 phần: Train, test và Optimize (tỉ lệ: 0.6-0.2-0.2) Dữ liệu nhiễu: mất cột battery\_power.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Kernel | Acc | Time | Tổng đường chéo ma trận confusion | Precision | Recall | F1 | Ảnh hưởng của nhiễu đến acc |
| Rbf | 95.5 | 0.043s | 572 | 0.9543 | 0.9533 | 0.9532 | 95.5>80.62 |
| linear | 99.21 | 11.408s | 584 | 0.9741 | 0.9733 | 0.9733 | 99.21>81.62 |
| Poly | 95.93 | 0.022s | 572 | 0.9536 | 0.9533 | 0.95316 | 95.93>80.06 |
| sigmoid | 19.57 | 0.1604s | 112 | 0.222 | 0.18 | 0.2014 | 19.57>20.69 |

Sử dụng các model SVM với dữ liệu nhiễu ta được mức độ ảnh hưởng của nhiễu.

B1. Bảng so sánh các loại model SVM

Linear độ chính xác cao khi không nhiễu nhưng ảnh hưởng nhiễu lớn, tốc độ chậm, acc quá cao (~100%) có thể bị overfiting. Trong khi đó với sigmoid độ chính xác thấp. Ở đây có 2 model phù hợp là rbf và poly khá tốt và ổn định hơn cả.

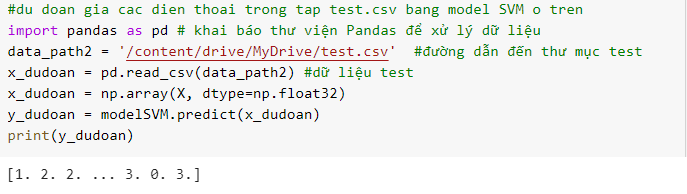
Ta lấy giá trị trung bình các model =>chọn kernel rbf.

1. **So sánh SVM với một số mô hình khác**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Acc | Time | Tổng đường chéo ma trận confusion | Precision | Recall | F1 | Ảnh hưởng của nhiễu đến acc |
| SVM (rbf) | 95.5 | 0.043s | 572 | 0.9542 | 0.9533 | 0.9532 | 95.5>80.62 |
| Logistic | 64.07 | 0.108s | 381 | 0.637 | 0.635 | 0.6356 | 64.07>63.5 |
| KNN | 96.07 | 0.0048s | 548 | 0.914 | 0.9133 | 0.9132 | 96.07>84.69 |
| DecisionTree | 100 | 0.8078s | 483 | 0.8078 | 0.805 | 0.8059 | 100>100 |

B2. Bảng so sánh SVM và các model khác

Kết quả dự đoán file Test.csv sử dụng model SVM :



1. **Tổng kết**

Là một kĩ thuật phân lớp khá phổ biến, SVM thể hiện được nhiều ưu điểm trong số đó có việc tính toán hiệu quả trên các tập dữ liệu lớn. Có thể kể thêm một số

## **Ưu điểm** :

* Xử lý trên không gian số chiều cao: SVM là một công cụ tính toán hiệu quả trong không gian chiều cao, trong đó đặc biệt áp dụng cho các bài toán phân loại văn bản và phân tích quan điểm nơi chiều có thể cực kỳ lớn.
* Tiết kiệm bộ nhớ: Do chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới nên chỉ có những điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết định.
* Tính linh hoạt - phân lớp thường là phi tuyến tính. Khả năng áp dụng Kernel mới cho phép linh động giữa các phương pháp tuyến tính và phi tuyến tính từ đó khiến cho hiệu suất phân loại lớn hơn.

## **Nhược điểm**:

* Bài toán số chiều cao: Trong trường hợp số lượng thuộc tính (p) của tập dữ liệu lớn hơn rất nhiều so với số lượng dữ liệu (n) thì SVM cho kết quả khá tồi.
* Chưa thể hiện rõ tính xác suất: Việc phân lớp của SVM chỉ là việc cố gắng tách các đối tượng vào hai lớp được phân tách bởi siêu phẳng SVM. Điều này chưa giải thích được xác suất xuất hiện của một thành viên trong một nhóm là như thế nào. Tuy nhiên hiệu quả của việc phân lớp có thể được xác định dựa vào khái niệm margin từ điểm dữ liệu mới đến siêu phẳng phân lớp mà chúng ta đã bàn luận ở trên.

## **Kết luận:**

## SVM là một phương pháp hiệu quả cho bài toán phân lớp dữ liệu. Nó là một công cụ đắc lực cho các bài toán về xử lý ảnh, phân loại văn bản, phân tích quan điểm. Một yếu tố làm nên hiệu quả của SVM đó là việc sử dụng ***Kernel function*** khiến cho các phương pháp chuyển không gian trở nên linh hoạt hơn.

Bọn em dùng 2 cách chạy :

+) Link chạy Google Colab: <https://colab.research.google.com/drive/177d4NIPeNUTcM_JNsg17BUnomLIJZkPE?usp=sharing>

+) File chạy Local trên Jupyter Notebook: <https://github.com/thehuongrbe/Hoc_May>

Contents

[**I)** **Tổng quan về SVM** 2](#_Toc57722448)

[**II)** **SVM trong mô hình dự đoán giá điện thoại** 3](#_Toc57722449)

[**Bước 1: Xử lý dữ liệu** 3](#_Toc57722450)

[**Bước 2: Train dữ liệu** 4](#_Toc57722451)

[**Bước 3: Tối ưu và đánh giá thuật thuật toán** 4](#_Toc57722452)

[**III)** **So sánh SVM với một số mô hình khác** 5](#_Toc57722453)

[**IV)** **Tổng kết** 6](#_Toc57722454)

[**Ưu điểm** : 6](#_Toc57722455)

[**Nhược điểm**: 6](#_Toc57722456)

[**Kết luận:** 7](#_Toc57722457)

*Chúng em mong nhận được đóng góp ý kiến từ thầy và mọi người.*

*Cảm ơn thầy và mọi người đã quan tâm và theo dõi !*