Trường Đại học Khoa học Tự nhiên
Khoa Công nghệ thông tin
11/01/2017

FINAL PROJECT REPORT

Machine Learning

Project:

MNIST CLASSIFICATION USING SVM

A report to: Trần Trung Kiên

Group 4: 1412630 – Đỗ Khánh Long Tường
1412652 – Phạm Đình Vương

MỤC LỤC

I. Danh sách nhóm

STT	MSSV	Họ tên	Email
1	1412630	Đỗ Khánh Long Tường	hanhdospla@gmail.com
2	1412652	Phạm Đình Vương	tulongthienvu@gmail.com

II. Giới thiệu

SVM (Support Vector Machine) là một mô hình học có giám sát được sử dụng phổ biến trong phân lớp dữ liệu. Trong phần báo cáo này, nhóm sẽ trình bày việc sử dụng SVM trong phân lớp ảnh chữ số viết tay và đưa ra những đánh giá, nhận xét về độ lỗi và khả năng tổng quát hóa của SVM thông qua linear kernel và RBF kernel. Từ đó, chọn ra được những siêu tham số tốt nhất áp dụng vào việc học nhằm thu được hàm dự đoán mà cho ra độ lỗi nhỏ trên tập test.

Bộ dữ liệu được sử dụng là bộ MNIST. Mỗi mẫu (example) trong bộ MNIST gồm: input là ảnh chữ số viết tay grayscale có kích thước 28 × 28 (như vậy, véc-tơ input sẽ có số chiều là 28 × 28=784), "correct ouput" ∈ {0, 1, …, 9} cho biết chữ số tương ứng của ảnh.

III. Nội dung thực hiện

Trong phần báo cáo này, nhóm sẽ sử dụng linear kernel và RBF kernel để áp dụng vào huấn luyện SVM.

1. Huấn luyện SVM dùng linear kernel

Đối với linear kernel, nhóm sẽ chọn ra siêu tham số C tốt nhất (chọn C khiến độ lỗi trên tập validation là nhỏ nhất) áp dụng vào huấn luyện nhằm thu được hàm dự đoán tốt nhất có thể.

Dưới đây là bảng thống kê những siêu tham số C được sử dụng để chọn lựa. Ứng với mỗi C thông qua việc học, ta có được độ lỗi trên tập train, độ lỗi trên tập validation và thời gian huấn luyện.

Constrain C	E_{in}	E_{val}	Training Time (s)
0.001	0.07610	0.0691	766.767234021
0.01	0.05594	0.0563	405.537752850
0.1	0.04188	0.0519	759.518000126
1	0.02754	0.0577	968.698999882
10	0.01692	0.0716	1452.18799996
100	0.01300	0.0777	4491.96499991

Biểu diễn dữ liệu thu được dưới dạng đồ thị:

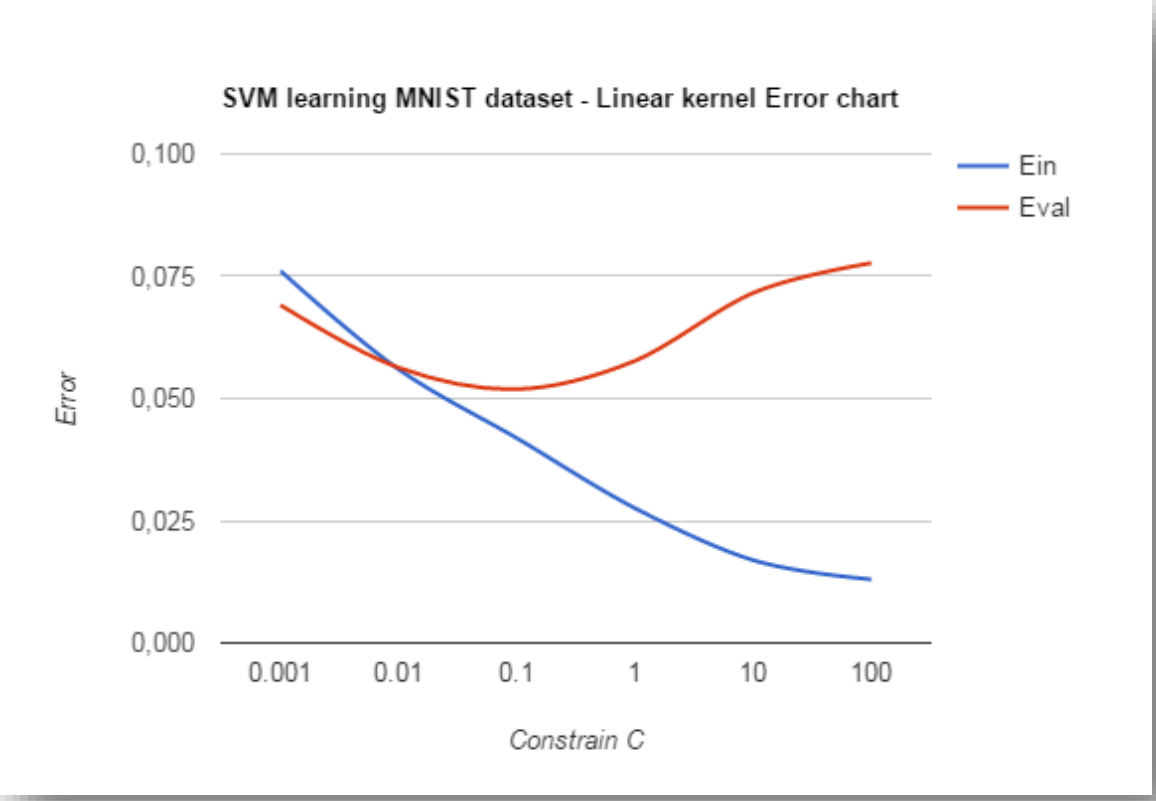

Figure 1: Kết quả các độ lỗi E_{in} và E_{val} với các giá trị tham số C khác nhau.

Nhận xét:

- C nhỏ cho biết độ lỗi cho phép của soft margin lớn → khả năng tổng quát hóa tốt. Tuy nhiên khi C quá nhỏ thì có thể dẫn đến trường hợp underfitting.
- C càng tăng thì Ein và Eval càng giảm đến khi C = 0.1 thì Ein tiếp tục giảm trong khi đó Eval bằng đầu tăng → bắt đầu xuất hiện overfitting do C càng tăng thì độ lỗi cho phép càng nhỏ do đó hyperplane bắt đầu cố gắng fit tập huấn luyện hết sức có thể.

Kết luận: Dựa theo nhận xét, biểu đồ và bảng số liệu, ta thấy rằng với C = 0.1 thì Eval = 0.0519 là giá trị nhỏ nhất trong các giá trị Eval có được. Vậy, ta chọn C = 0.1 là giá trị siêu tham số C dùng để huấn luyện nhằm thu được hàm dự đoán tốt nhất.

Kết quả: Sau khi huấn luyện và kiểm tra đánh giá dựa trên tập test, ta có kết quả sau:

- Độ lỗi trên tập huấn luyện: 0.04125
- **Độ lỗi trên tập test: 0.0526**
- Thời gian huấn luyện: 365.522s

2. Huấn luyện SVM dùng RBF kernel

Đối với RBF kernel, nhóm sẽ chọn ra siêu tham số C và γ tốt nhất (chọn cặp C và γ khiến độ lỗi trên tập validation là nhỏ nhất) sau đó sử dụng cặp siêu tham số đó để huấn luyện mô hình SVM dùng RBF kernel nhằm thu được hàm dự đoán tốt nhất có thể.

Dưới đây là bảng thống kê những siêu tham số C và γ được sử dụng để chọn lựa. Ứng với mỗi cặp C, γ thông qua việc học, ta có được độ lỗi trên tập train, độ lỗi trên tập validation và thời gian huấn luyện.

C \ γ	0.0001	0.001	0.01	0.1	1	10
E_{in}	0.88644	0.88644	0.88644	0.88644	0.88644	0.8864
E_{val}	0.8936	0.8936	0.8936	0.8936	0.8936	0.8936
Time (s)	6291	5024	5960	6544	4904	7439
E_{in}	0.88644	0.88644	0.4749	0.88644	0.88644	0.8864
E_{val}	0.8936	0.8936	0.4596	0.8936	0.8936	0.8936
Time (s)	4928	5884	5161	6525	6233	8753
E_{in}	0.88644	0.2537	0.0939	0.7843	0.88644	0.8864
E_{val}	0.8936	0.2279	0.0822	0.7828	0.8936	0.8936
Time (s)	4925	4370	2224	5066	7467	6311
E_{in}	0.23658	0.0982	0.04702	0.28952	0.88644	0.8864
E_{val}	0.2105	0.0861	0.0422	0.3125	0.8936	0.8936
Time (s)	3904	1609	2166	4722	6916	7398
E_{in}	0.0997	0.0642	0.01526	0.00004	0.0	0.0
E_{val}	0.0879	0.0589	0.0223	0.0448	0.8176	0.8936
Time (s)	1460	630	322	6602	14076	10151
E_{in}	0.06864	0.0379	0.00058	0.0	0.0	0.0
E_{val}	0.0631	0.0408	0.0165	0.0434	0.8138	0.8936
Time (s)	637	324	309	6030	18055	18986
E_{in}	0.04984	0.00942	0.0	0.0	0.0	0.0
E_{val}	0.0524	0.0282	0.168	0.0434	0.8138	0.8936
Time (s)	427	256	318	6436	17083	18743
E_{in}	0.00058	0.00008	0.0	0.0	0.0	0.0
E_{val}	0.0165	0.0285	0.168	0.0434	0.8138	0.8936
Time (s)	272	280	308	6349	19515	19744

Biểu diễn dữ liệu thu được dưới dạng đồ thị:

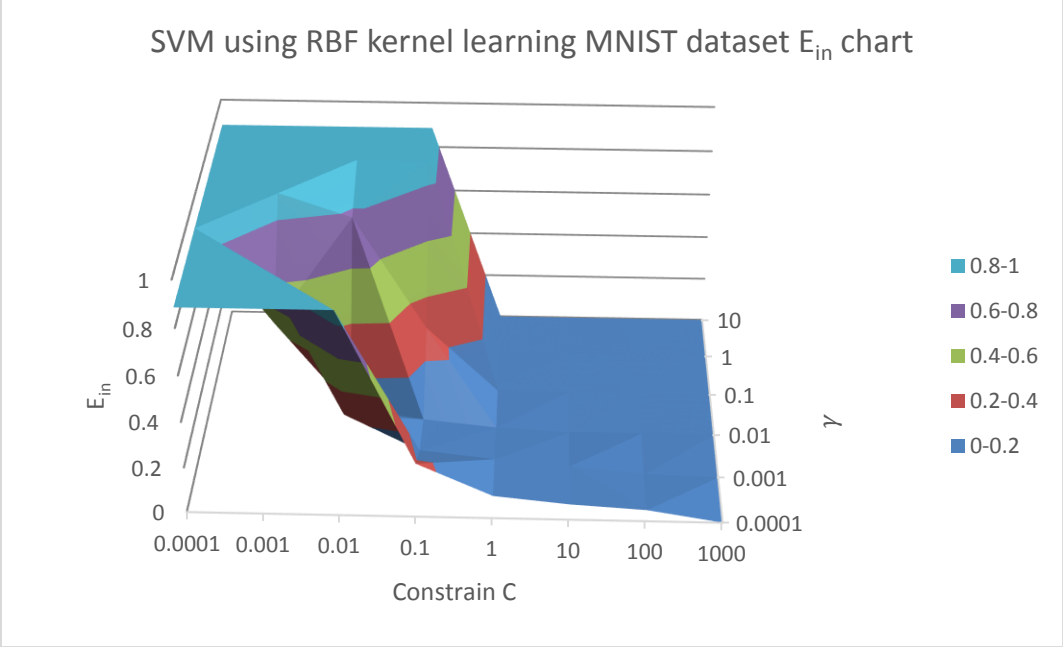

Figure 2: Kết quả các độ lỗi E_{in} với các giá trị tham số C và γ khác nhau.

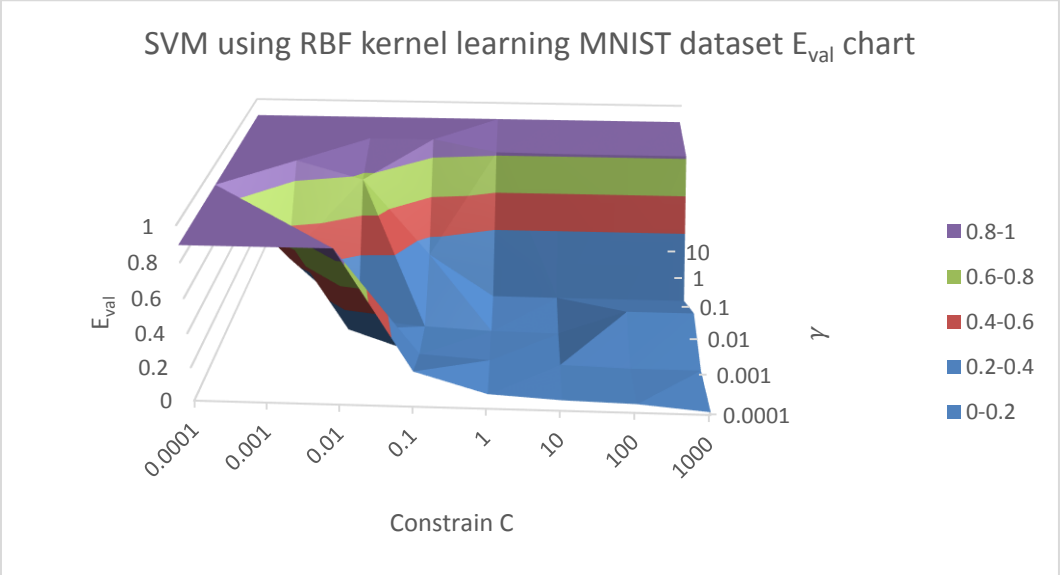

Figure 3: Kết quả các độ lỗi E_{val} với các giá trị tham số C và γ khác nhau.

Nhận xét:

- C nhỏ cho biết độ lỗi cho phép của soft margin lớn và γ nhỏ khiến độ rộng Basis Function trong RBF $\exp(-\gamma\|x - x_n\|^2)$ rộng → khả năng tổng quát hóa tốt nhưng khi C quá nhỏ hoặc γ quá nhỏ thì sẽ dẫn tới trường hợp underfitting.
- C và γ càng tăng thì E_{in} và E_{val} càng giảm nhưng đến khi C = 10, γ = 0.01 thì bắt đầu xuất hiện overfitting.

Kết luận: Dựa theo nhận xét, biểu đồ và bảng số liệu, ta thấy rằng với cặp C = 10, γ = 0.01 thì Eval = 0.0165 là giá trị nhỏ nhất trong các giá trị Eval có được. Vậy, ta chọn cặp giá trị C = 10, γ = 0.01 là các giá trị siêu tham số dùng để huấn luyện nhằm thu được hàm dự đoán tốt nhất.

Kết quả: Sau khi huấn luyện và kiểm tra đánh giá dựa trên tập test, ta có kết quả sau:

- Độ lỗi trên tập huấn luyện: 0.00058
- **Độ lỗi trên tập test: 0.018**
- Thời gian huấn luyện: 296.014s

IV. Đánh giá tiến độ thực hiện và đóng góp của các thành viên:

STT	Nội dung thực hiện	Thực hiện chính
1	Lập trình SVM – Linear Kernel qua module sci-kitlearn	Đỗ Khánh Long Tường
2	Lập trình SVM – RBF Kernel qua module sci-kitlearn	Phạm Đình Vương
3	Huấn luyện SVM Linear Kernel với C = 0.01, C = 0.1, C = 1, C = 10, C = 100	Đỗ Khánh Long Tường
4	Huấn luyện SVM RBF Kernel với C = 0.0001, C = 0.01, C = 0.1, C = 1, C = 10, C = 100, C = 1000 ứng với γ = 0.0001, γ = 0.001, γ = 0.01, γ = 0.1	Phạm Đình Vương
5	Huấn luyện SVM RBF Kernel với C = 0.0001, C = 0.01, C = 0.1, C = 1, C = 10, C = 100, C = 1000 ứng với γ = 1, γ = 10	Đỗ Khánh Long Tường
6	Huấn luyện SVM và kiểm tra trên tập test	Phạm Đình Vương
7	Báo cáo chi tiết	Đỗ Khánh Long Tường