TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP LỚN

MÔN HỌC: PHÂN TÍCH DỮ LIỆU

ĐỀ TÀI: CHẨN ĐOÁN BỆNH TIỂU ĐƯỜNG DỰA VÀO BỘ DỮ LIỆU KẾT QUẢ XÉT NGHIỆM MÁU

Giảng viên hướng dẫn : Hồ Hướng Thiên
Sinh viên thực hiện : Tạ Thị Thiên Thanh - 2151013088
Phạm Công Thuận - 2151013097
Lớp : DH21CS01

MỤC LỤC

I. MỞ ĐẦU

a

II. TỔNG QUAN VỀ PHÂN TÍCH DỮ LIỆU

1. Khái niệm

a

2. Quá trình phân tích dữ liệu

a

3. Các kỹ thuật phân tích dữ liệu

d

4. Các ứng dụng của phân tích dữ liệu

a

III. MÔ TẢ DỮ LIỆU

Dữ liệu lâm sàng là dữ liệu thu được tại một cơ sở khám chữa bệnh chẳng hạn như trạm y tế, phòng khám, bệnh viện. Đề tài sử dụng bộ dữ liệu lâm sàng của hơn 5000 bệnh nhân được thu thập để phục vụ cho việc nghiên cứu chức năng thận, sức khỏe tim mạch và chẩn đoán bệnh tiểu đường. Đây là bộ dữ liệu quan trọng để đánh giá nguy cơ mắc bệnh tim và bệnh tiểu đường kèm theo tình trạng suy giảm chức năng thận. Đề tài tập trung chủ yếu vào khả năng chẩn đoán bệnh tiểu đường bằng các phương pháp phân tích dữ liệu dựa trên bộ dữ liệu lâm sàng thu được sau khi xét nghiệm máu.

Đề tài sử dụng các file dữ liệu như sau:

1. Diabetes-Classification.csv: Bao gồm, … thể hiện, 9 thuộc tính và 1 cột phân lớp. Bộ dữ liệu này được xem là bộ dữ liệu gốc chứa toàn bộ kết quả xét nghiệm và chẩn đoán bệnh tiểu đường của bệnh nhân.
2. train.csv: … thể hiện, 9 thuộc tính và 1 cột phân lớp. Bộ dữ liệu này chứa dữ liệu trích từ 90% số thể hiện của bộ dữ liệu gốc dùng để huấn luyện mô hình.
3. test.csv: … thể hiện, 9 thuộc tính, không có cột phân lớp. Đây là bộ dữ liệu thử nghiệm mô hình được trích từ 10% thể hiện còn lại của bộ dữ liệu gốc, do đó các dữ liệu này chưa từng xuất hiện trong bộ dữ liệu huấn luyện.
4. validate.csv: Bao gồm … thể hiện, 9 thuộc tính và 1 cột phân lớp. Bộ dữ liệu này chứa toàn bộ dữ liệu trong file “test.csv” nhưng có thêm cột phân lớp được lấy từ bộ dữ liệu gốc và được dùng để đánh giá kết quả huấn luyện mô hình.

Các thuộc tính có trong bộ dữ liệu bao gồm:

1. Age: Tuổi của bệnh nhân.
2. Gender: Giới tính của bệnh nhân bao gồm 2 giá trị: “M” tương ứng với giới tính Nam và “F” là giới tính nữ.
3. Body Mass Index (BMI): Chỉ số BMI của bệnh nhân. BMI là chỉ số khối cơ thể dùng để xác định cân nặng của một người đang thiếu cân, thừa cân hay cân đối.
4. Chol: Tỷ lệ cholesterol có trong máu. Cholesterol là một loại chất béo được sản sinh ra từ việc tiêu thụ thức ăn hoặc cơ thể tự sản xuất.
5. TG (Triglycerides): Tỷ lệ triglycerides có trong máu. Triglycerides là một dạng chất béo trung tính chứa 3 axit béo và có nguồn gốc từ mỡ động vật, thực vật mà bệnh nhân tiêu thụ.
6. HDL (High-Density Lipoprotein): Chỉ số Lipoprotein tỷ trọng cao. Lipoprotein tỷ trọng cao được xem như một loại cholesterol có lợi giúp vận chuyển cholesterol dư thừa tích trữ dưới mạch máu về gan để xử lý và đào thải ra ngoài.
7. LDL (Low-Density Lipoprotein): Chỉ số Lipoprotein tỷ trọng thấp. Lipoprotein tỷ trọng thấp các cholesterol có hại làm tăng nguy cơ xơ vữa động mạch.
8. Cr (Creatinin): Chỉ số creatinin trong máu. Creatinin là một chất cặn bã được đào thải thông qua thận, từ đó phản ánh chức năng thận.
9. BUN (Blood Urea Nitrogen): Chỉ số BUN dùng để đánh giá chức năng gan và thận.

Cột “Diagnosis” là lớp của từng bệnh nhân bao gồm 2 nhãn 0 và 1. Bệnh nhân được chẩn đoán mắc bệnh tiểu đường được gán nhãn là 1 và ngược lại có nhãn là 0.

IV. TIỀN XỬ LÝ DỮ LIỆU

1. Một số thông tin tổng quan về dữ liệu trước khi tiền xử lý

	Age	Gender	BMI	Chol	TG	HDL	LDL	Cr	BUN	Diagnosis
0	50.0	F	24.0	4.20	0.90	2.40	1.40	46.0	4.7	0.0
1	26.0	M	23.0	3.70	1.40	1.10	2.10	62.0	4.5	0.0
2	33.0	M	21.0	4.90	1.00	0.80	2.00	46.0	7.1	0.0
3	45.0	F	21.0	2.90	1.00	1.00	1.50	24.0	2.3	0.0
4	50.0	F	24.0	3.60	1.30	0.90	2.10	50.0	2.0	0.0
...	...	...	...	...	...	...	...	...	...	...
5327	56.0	M	33.0	5.00	1.70	1.45	1.90	84.0	5.0	1.0
5328	61.0	F	39.0	3.80	3.00	0.90	1.70	111.0	10.5	1.0
5329	60.0	M	24.0	3.40	5.30	1.10	3.60	70.0	7.5	0.0
5330	52.0	F	24.0	5.07	1.08	1.37	3.31	57.3	4.5	0.0
5331	86.0	M	23.0	5.26	2.01	1.43	2.94	64.5	4.2	0.0

5332 rows × 10 columns

Hình 4.1.x.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5332 entries, 0 to 5331
Data columns (total 10 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Age        5310 non-null   float64
 1   Gender     5332 non-null   object
 2   BMI        5309 non-null   float64
 3   Chol       5306 non-null   float64
 4   TG         5300 non-null   float64
 5   HDL        5310 non-null   float64
 6   LDL        5314 non-null   float64
 7   Cr         5311 non-null   float64
 8   BUN        5311 non-null   float64
 9   Diagnosis  5313 non-null   float64
dtypes: float64(9), object(1)
```

Hình 4.1.x. Tổng quan thông tin của bộ dữ liệu

Hình 4.1.x. cho thấy bộ dữ liệu có tổng cộng 5332 dòng (biểu hiện) và 10 cột (gồm 9 thuộc tính và 1 cột phân lớp), trong đó:

- Thuộc tính ‘Gender’ có kiểu dữ liệu object
- Các thuộc tính còn lại có kiểu dữ liệu float

```
1 print(df.isna().sum())
```

```
Age          22
Gender        0
BMI          23
Chol         26
TG           32
HDL          22
LDL          18
Cr           21
BUN          21
Diagnosis    19
dtype: int64
```

Hình 4.1.x. Số lượng giá trị NaN của từng thuộc tính

```
1 duplicated_rows = df[df.duplicated()]
2 print(len(duplicated_rows))
```

```
200
```

Hình 4.1.x. Số lượng dòng (thể hiện) trong bộ dữ liệu

	Age	BMI	Chol	TG	HDL	LDL	Cr	BUN	Diagnosis
count	5310.000	5309.000	5306.000	5300.000	5310.000	5314.000	5311.000	5311.000	5313.000
mean	48.638	24.130	4.690	1.504	1.346	2.705	70.626	4.596	0.203
std	15.253	7.719	3.448	3.984	4.349	3.911	29.764	4.919	3.367
min	-94.000	-99.000	-82.000	-94.000	-95.000	-98.000	-93.000	-98.000	-83.000
25%	36.000	22.000	4.180	0.900	1.090	2.270	57.750	3.900	0.000
50%	49.000	24.000	4.800	1.370	1.300	2.780	70.000	4.710	0.000
75%	59.000	27.000	5.460	2.100	1.590	3.390	81.400	5.600	1.000
max	93.000	47.000	11.650	32.640	9.900	9.900	800.000	38.900	1.000

Hình 4.1.x. Các số liệu thống kê của từng thuộc tính kiểu số

Nhìn Hình 4.1.x., ta có những nhận xét như sau:

- Đối với thuộc tính Age:

Khoảng giá trị hiện tại là [-94, 93]

Khoảng giá trị chấp nhận được đối với thuộc tính tuổi là Age > 0

- Đối với thuộc tính BMI:

Khoảng giá trị hiện tại là [-99, 47]

Theo Wikipedia [17], BMI cao nhất từng ghi nhận được là 251.1 và vì BMI được tính dựa trên cân nặng và chiều cao nên khoảng BMI có thể chấp nhận được là $0 < BMI \leq 251$.

- aa

2. Tiến hành tiền xử lý dữ liệu

a

V. TÌM LUẬT KẾT HỢP BẰNG APRIORI

1. Khái quát thuật toán Apriori

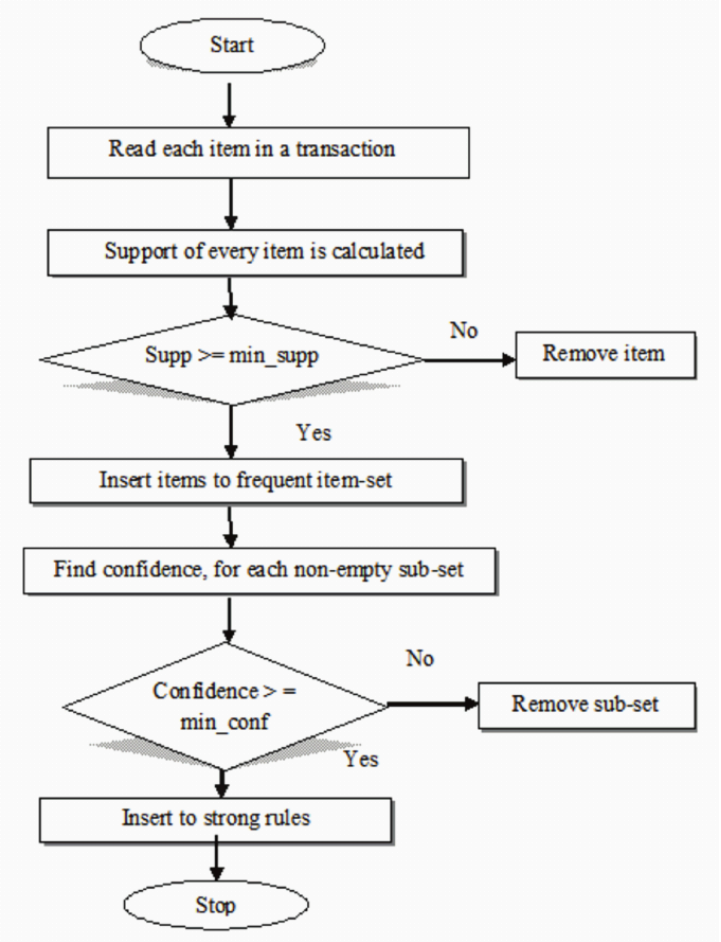
Thuật toán Apriori là một trong những thuật toán quan trọng được sử dụng trong lĩnh vực khai phá dữ liệu để phân tích các tập dữ liệu lớn và đưa ra các luật kết hợp. Ý tưởng của thuật toán là tìm tất cả tập các hạng mục (itemsets) có độ hỗ trợ (support) lớn hơn hoặc bằng độ hỗ trợ tối thiểu được quy định trước (minSupp) [16].

Độ hỗ trợ (support) là "độ đo" cho 1 tập itemset. Những tập có độ hỗ trợ càng lớn thì tỷ lệ xuất hiện càng cao và được xem là những tập "đáng quan tâm" để tiến hành khai thác. Độ hỗ trợ của tập S được tính theo công thức:

$$Supp(S) = Count(S) / N$$

Độ tin cậy (confidence) là "độ đo" cho 1 luật (rule) có dạng L -> R dùng để xác định tỷ lệ R xuất hiện mỗi khi L xuất hiện. Độ tin cậy của luật L -> R được tính như sau:

$$Conf(L \rightarrow R) = Count(L \cup R) / Count(L)$$



Hình 6.1.1. Sơ đồ khối biểu diễn các bước của Apriori

2. Thực hiện tìm luật kết hợp

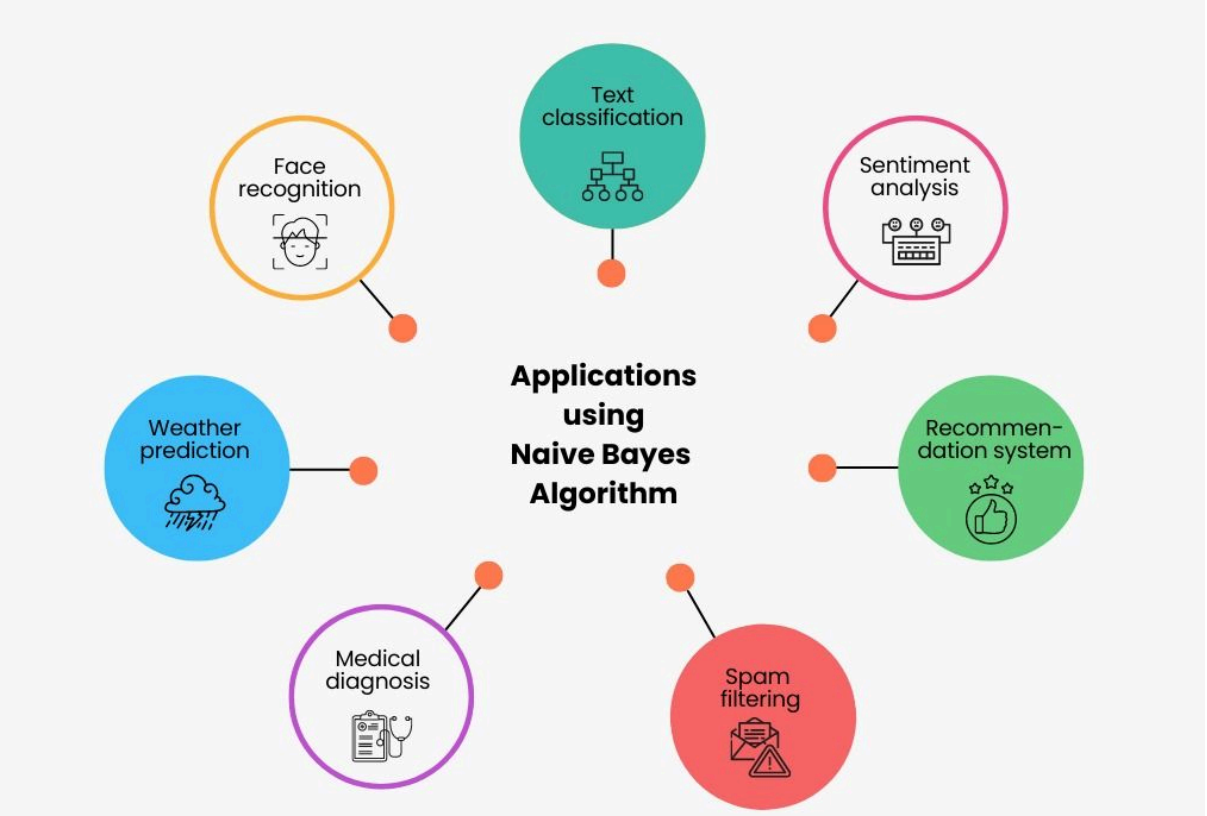
a

VI. THUẬT TOÁN PHÂN LỚP NAIVE BAYES

1. Khái quát thuật toán Naive Bayes

Naive Bayes classifier là một thuật toán thuộc nhóm các thuật toán áp dụng định lý Bayes và giả định "ngây thơ" (naive). Giả định này cho rằng mọi thuộc tính đầu vào (ví dụ: độ tuổi, nghề nghiệp, tình trạng hôn nhân,...) đều độc lập với nhau, giá trị của thuộc tính này không liên quan và không làm ảnh hưởng đến giá trị của một thuộc tính nào khác [12].

Naive Bayes thường được sử dụng trong các bài toán phân loại dữ liệu dạng văn bản, nhận diện văn bản spam, xây dựng các mô hình dự đoán và nhiều ứng dụng khác trong lĩnh vực học máy (*machine learning*).



Hình 7.1.1. Một số ứng dụng của thuật toán Naive Bayes

Đề tài sử dụng phân phối Gaussian Naive Bayes để áp dụng cho bộ dữ liệu mà thành phần của nó là các biến liên tục. Gaussian Naive Bayes là một phân phối phổ biến, dễ dàng thực hiện sau khi đã tính được giá trị trung bình và độ lệch chuẩn từ bộ dữ liệu "train" của đề tài.

2. Định lý Bayes

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Hình 7.2.1. Công thức tính xác suất có điều kiện

Công thức trên giúp chúng ta xác định được tần suất của A xảy ra khi điều kiện B đã xảy ra, ký hiệu là P(A|B). Xác suất P(A|B) có thể được tính dựa vào xác suất P(B|A), P(A) và P(B) [12]. Trong đó:

- P(A|B): Xác suất A xảy ra khi B đã xảy ra

- P(A): Xác suất A xảy ra
- P(B|A): Xác suất B xảy ra khi A đã xảy ra
- P(B): Xác suất B xảy ra

3. Ưu điểm

Bộ phân lớp Naive Bayes có một số điểm nổi trội hơn so với các thuật toán phân lớp khác như sau:

- Thuật toán đơn giản và dễ triển khai nên chỉ cần số lượng ít dữ liệu để huấn luyện cho mô hình.
- Naive Bayes có khả năng xử lý dữ liệu lớn với hiệu suất cao. Khi số lượng thuộc tính hay số thể hiện của bộ dữ liệu tăng lên thì hiệu suất suy giảm không đáng kể.
- Có thể đưa ra dự đoán cho bộ dữ liệu test một cách nhanh chóng,

4. Nhược điểm

Bên cạnh ưu điểm thì Naive Bayes cũng có những hạn chế:

- Naive Bayes giả định rằng tất cả thuộc tính đều độc lập, điều này hầu như ít xảy ra trong thực tế, làm giảm độ chính xác của mô hình.
- Nếu gặp một giá trị chưa từng xuất hiện trong bộ dữ liệu huấn luyện, Naive Bayes sẽ tính ra xác suất bằng 0 và không thể dự đoán trong tương lai.

5. Áp dụng vào dự đoán khả năng đồng ý của khách hàng

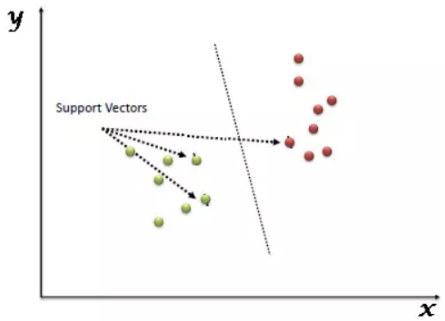
a

VII. THUẬT TOÁN PHÂN LỚP SUPPORT VECTOR MACHINES (SVM)

1. Khái quát thuật toán SVM

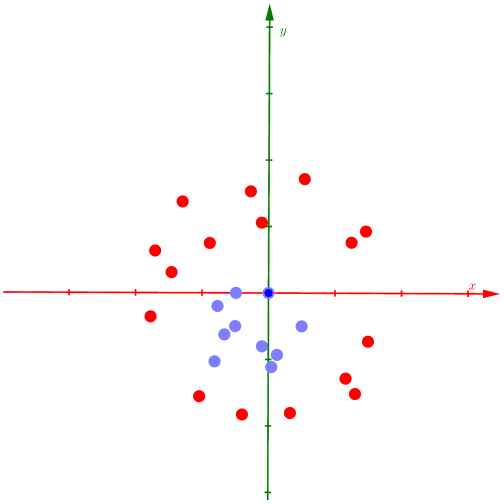
Trong lĩnh vực học máy (*machine learning*), Support Vector Machine (*SVM*) là mô hình tìm biên tối đa có giám sát (*supervised max-margin model*) phân tích dữ liệu nhằm sử dụng cho phân lớp, hồi quy và phát hiện các ngoại lệ [7]. Tuy nhiên SVM được sử dụng phổ biến trong phân lớp.

Support vector có thể được hiểu là các điểm (đối tượng) trên tọa độ. Còn support vector machine là biên giới để phân chia các điểm trên một cách tối ưu nhất [8].

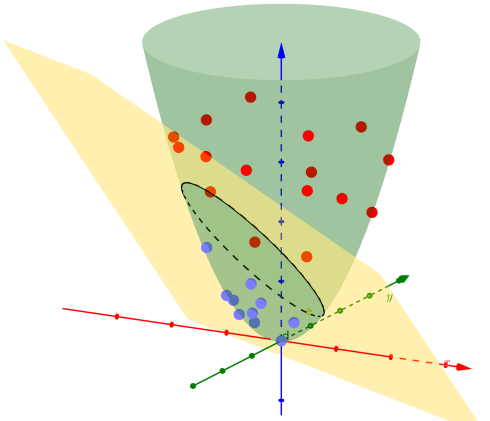


Hình 8.1.1. Hình minh họa hoạt động của SVM qua đồ thị tọa độ [8]

Ý tưởng cơ bản của SVM là chuyển đổi dữ liệu đầu vào thành dữ liệu có chiều không gian cao hơn giúp dễ dàng phân lớp dữ liệu [11]. SVM có thể xử lý cả dữ liệu phân tách tuyến tính (*linearly separable*) và phi tuyến tính (*non-linearly separable*) bằng cách dùng các loại hàm kernel (*kernel function*) khác nhau [10].



Hình 8.1.2. Dữ liệu phi tuyến tính trong không gian hai chiều [11]



Hình 8.1.3. Dữ liệu đã được chuyển đổi thành dữ liệu phân tách tuyến tính trong không gian ba chiều [11]

2. Hàm Kernel

Hàm Kernel là các hàm toán học có chức năng chuyển đổi dữ liệu đầu vào thành không gian có chiều cao hơn, khi đó dữ liệu trở nên dễ phân tách tuyến tính. Kernel SVM hoạt động bằng việc tìm một hàm số $\Phi(x)$ biến đổi dữ liệu đầu vào x từ không gian ban đầu thành dữ liệu trong không gian mới [11].

Tính chất các hàm Kernel $k()$:

- Đối xứng: $k(x,\ z)\ =\ k(z,\ x)$
- Về mặt lý thuyết, cần phải thỏa mãn điều kiện Mercer:

$$\sum_{n=1}^{N}\sum_{m=1}^{N} k(x_m x_n)c_n c_m \geq 0,\ \forall c_i \in R,\ i = 1,\ 2,\ \ldots,\ N\ (1)$$

- Trong thực hành, $k()$ có thể không thỏa mãn điều kiện Mercer nhưng vẫn cho ra kết quả nên vẫn được gọi là hàm Kernel.

Nếu hàm Kernel thỏa mãn điều kiện (1), xét $c_n = y_n \lambda_n$, ta có:

$$\lambda^T K \lambda = \sum_{n=1}^{N}\sum_{m=1}^{N} k(x_m,\ x_n) y_n y_m \lambda_n \lambda_m \geq 0,\ \forall \lambda_n\ (2)$$

Trong đó:

- K là ma trận đối xứng
- $k_{nm} = y_n y_m k(x_n,\ x_m)$ là phần tử hàng n cột m của K [11]

Tên	Công thức	kernel	Thiết lập hệ số
linear	$\mathbf{x}^T\mathbf{z}$	'linear'	không có hệ số
polynomial	$(r + \gamma\mathbf{x}^T\mathbf{z})^d$	'poly'	d: degree, γ: gamma, r: coef0
sigmoid	$\tanh(\gamma\mathbf{x}^T\mathbf{z} + r)$	'sigmoid'	γ: gamma, r: coef0
rbf	$\exp(-\gamma\|\|\mathbf{x} - \mathbf{z}\|\|_2^2)$	'rbf'	$\gamma > 0$: gamma

Hình 8.2.1. Bảng tóm tắt các hàm Kernel thông dụng và cách sử dụng trong sklearn [11]

3. Ưu điểm

SVM là một kỹ thuật được sử dụng phổ biến với những ưu điểm như sau:

- Hoạt động hiệu quả trên không gian nhiều chiều: với cơ chế hoạt động tăng chiều không gian của dữ liệu đầu vào, SVM thích hợp với các bài toán phân loại văn bản và phân tích quan điểm với số chiều lớn [13].
- Tính linh hoạt: SVM được áp dụng vào cả phân lớp và hồi quy với nhiều ứng dụng trong các lĩnh vực NLP (*Natural Language Processing*), thị giác máy tính (*Computer Vision*), … Thêm vào đó, Kernel trong SVM có thể xử lý linh động trên cả dữ liệu phân tách tuyến tính và phi tuyến tính làm tăng hiệu suất phân lớp [14].
- Khả năng chống nhiễu: SVM thuần (*Hard Margin SVM*) chưa thể hiện tốt khả năng xử lý nhiễu nhưng Soft Margin SVM đã khắc phục rất tốt bằng cách hy sinh điểm nhiễu để được một margin tốt hơn [14].

4. Nhược điểm

Những ưu điểm trên cũng đi kèm theo những hạn chế:

- Chưa thể hiện rõ tính xác suất: SVM chỉ hoạt động theo cơ chế tách dữ liệu bằng siêu phẳng và xác định dựa vào margin từ điểm dữ liệu đến siêu phẳng chứ chưa giải thích được xác suất xuất hiện của từng thể hiện trong tập dữ liệu [13, 14].
- Khó khăn trong lựa chọn Kernel: việc lựa chọn kernel ảnh hưởng rất lớn đến hiệu suất của SVM vì thế việc xác định kernel theo đặc điểm từng bộ dữ liệu là rất khó và quan trọng [14].

5. Áp dụng vào dự đoán khả năng đồng ý của khách hàng

a

VIII. KẾT LUẬN CHUNG

a

❖ CÁC TÀI LIỆU THAM KHẢO

[1] N.T.V. Hà, "Tổng quan về khai phá dữ liệu và phương pháp khai phá luật kết hợp trong cơ sở dữ liệu", 2020. [Trực tuyến]. Địa chỉ: https://tapchicongthuong.vn/bai-viet/tong-quan-ve-khai-pha-du-lieu-va-phuong-phap-khai-pha-luat-ket-hop-trong-co-so-du-lieu-69634.htm. [Truy cập 19/12/2023].

[2] Wikipedia, "Khai phá dữ liệu", 2023. [Trực tuyến]. Địa chỉ: https://vi.wikipedia.org/wiki/Khai_ph%C3%A1_d%E1%BB%AF_li%E1%BB%87u. [Truy cập 19/12/2023].

[3] Van Bien's blog, "Quy trình Khai phá dữ liệu (Process of Data mining)", 2013. [Trực tuyến]. Địa chỉ: https://bienuit.wordpress.com/2013/09/07/quy-trinh-khai-pha-du-lieu-process-of-data-mining/. [Truy cập 19/12/2023].

[4] chucvn, "Khai phá dữ liệu: Ứng dụng, hướng nghiên cứu và công cụ", 2014. [Trực tuyến]. Địa chỉ: https://bis.net.vn/forums/t/815.aspx. [Truy cập 27/12/2023].

[5] Viện IBS, "Data Mining: Ứng dụng của Data Mining trong các lĩnh vực", 2023. [Trực tuyến]. Địa chỉ: https://insight.isb.edu.vn/ung-dung-cua-data-mining-trong-cac-linh-vuc/. [Truy cập 27/12/2023].

[6] S. Moro, R. Laureano và P. Cortez, "A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems", ??? https://www.kaggle.com/datasets/prakharrathi25/banking-dataset-marketing-targets

[7] Wikipedia, "Support vector machine", 2023. [Trực tuyến]. Địa chỉ: https://en.wikipedia.org/wiki/Support_vector_machine. [Truy cập 29/12/2023].

[8] H.C. Trung, "Giới thiệu về Support Vector Machine (SVM)", 2020. [Trực tuyến]. Địa chỉ: https://viblo.asia/p/gioi-thieu-ve-support-vector-machine-svm-6J3ZgPVElmB. [Truy cập 29/12/2023].

[9] Scikit-learn, “Support Vector Machine”, 2023. [Trực tuyến]. Địa chỉ: https://scikit-learn.org/stable/modules/svm.html. [Truy cập 29/12/2023].

[10] F. Tabsharani, “support vector machine (SVM)”, 2023. [Trực tuyến]. Địa chỉ: https://www.techtarget.com/whatis/definition/support-vector-machine-SVM#:~:text=A%20support%20vector%20machine%20(SVM)%20is%20a%20type%20of%20supervised,data%20set%20into%20two%20groups. [Truy cập 29/12/2023].

[11] V.H. Tiệp, “Bài 21: Kernel Support Vector Machine”, 2017. [Trực tuyến]. Địa chỉ: https://machinelearningcoban.com/2017/04/22/kernelsmv/#-ham-so-kernel. [Truy cập 29/12/2023].

[12] V. Nga, “Tìm hiểu Naive Bayes Classification - Phần 1”, 2022. [Trực tuyến]. Địa chỉ: https://200lab.io/blog/tim-hieu-naive-bayes-classification-phan-1/. [Truy cập 5/1/2024].

[13] P.V. Toàn, “Support Vector Machine trong học máy - Một cái nhìn đơn giản hơn”, 2016. [Trực tuyến]. Địa chỉ: https://viblo.asia/p/support-vector-machine-trong-hoc-may-mot-cai-nhin-don-gian-hon-XQZkxoQmewA. [Truy cập 8/1/2024].

[14] goelaparna1520, “Support vector machine in Machine Learning”, 2023. [Trực tuyến]. Địa chỉ: https://www.geeksforgeeks.org/support-vector-machine-in-machine-learning/. [Truy cập 8/1/2024].

[15] N. Quy, “Thuật toán phân cụm K-Means”, 2021. [Trực tuyến]. Địa chỉ: https://ndquy.github.io/posts/thuat-toan-phan-cum-kmeans/. [Truy cập 10/1/2024].

[16] N.M. Đức, “Thuật toán Apriori khai phá luật kết hợp trong Data Mining”, 2019. [Trực tuyến]. Địa chỉ: https://viblo.asia/p/thuat-toan-apriori-khai-pha-luat-ket-hop-trong-data-mining-3P0lPEv85ox. [Truy cập 10/1/2024].

[17] Wikipedia, “List of heaviest people”, 2024, [Trực tuyến]. Địa chỉ: https://en.wikipedia.org/wiki/List_of_heaviest_people. [Truy cập 3/4/2024].