

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN I



BÁO CÁO CUỐI KÌ MÔN HỌC
CHUYÊN ĐỀ HỆ THỐNG THÔNG TIN

Đề tài: Ứng dụng Sigmoid SVM trong mô hình nhận dạng giọng nói nam và nữ

Giảng viên : Phạm Văn Cường

Nhóm Bài Tập Lớn : 09

Thành viên:

Vũ Minh Đức - B20DCCN202

Vũ Quang Vinh – B20DCCN740

Đình Quang Duy - B20DCCN147

Tạ Văn Phong - B20DCCN498

Trần Quang Ngọc - B20DCCN474

Nguyễn Khánh Đức - B20DCCN198

Hà Nội - 2024

MỞ ĐẦU	3
CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI	4
1. Lý do lựa chọn đề tài này:	4
2. Phương pháp tiếp cận:.....	4
3. Dữ liệu sử dụng:	5
4. Tiến trình thực hiện:	5
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ MÔ HÌNH NGHIÊN CỨU.....	6
1. Cơ sở lý thuyết.....	6
2. Sigmoid SVM	7
a. Khái niệm	7
b. Nguyên tắc hoạt động	7
c. Ưu và nhược điểm:.....	8
d. Kết luận:	8
CHƯƠNG III: THIẾT KẾ, MÔ PHỎNG, KẾT QUẢ.....	9
VÀ THẢO LUẬN.....	9
1. Tiền xử lý dữ liệu	9
1.1. Đọc dữ liệu:	9
1.2. Kiểm tra giá trị thiếu:.....	9
1.3. Mã hóa nhãn:	10
1.4. Chuẩn hóa dữ liệu:	10
2. Quy trình thực hiện.....	10
2.1. Tính toán hiển thị ma trận tương đương.....	10
2.2. Phân chia dữ liệu thành tập huấn luyện và tập kiểm tra:.....	11
2.3. Huấn luyện và đánh giá mô hình:	12
2.4. Tối ưu hóa tham số gamma:.....	12
2.5. Huấn luyện và đánh giá mô hình với tham số tối ưu:.....	14
3. Kết quả và thảo luận	14
3.1. Độ chính xác của mô hình:	14
3.2. Thảo luận:	14
Tài liệu tham khảo.....	14

MỞ ĐẦU

Trong kỷ nguyên của dữ liệu lớn và trí tuệ nhân tạo, học máy đang trở thành một trong những lĩnh vực nghiên cứu và ứng dụng phát triển nhanh nhất. Học máy cung cấp các công cụ và thuật toán giúp máy tính có khả năng học từ dữ liệu, đưa ra dự đoán và quyết định mà không cần được lập trình một cách rõ ràng cho từng nhiệm vụ cụ thể. Trong số các thuật toán học máy, Support Vector Machine (SVM) là một trong những công cụ mạnh mẽ và phổ biến nhất cho các bài toán phân loại và hồi quy.

Support Vector Machine (SVM) là một thuật toán học có giám sát, nổi bật với khả năng tìm ra siêu phẳng tối ưu để phân tách các lớp dữ liệu trong không gian nhiều chiều. Đặc biệt, SVM có thể xử lý dữ liệu phi tuyến tính một cách hiệu quả thông qua việc sử dụng các hàm kernel. Kernel là những hàm ánh xạ dữ liệu từ không gian gốc lên không gian đặc trưng cao hơn, nơi dữ liệu có thể được phân tách tuyến tính. Các loại kernel phổ biến bao gồm kernel tuyến tính (Linear Kernel), kernel đa thức (Polynomial Kernel), kernel Gaussian hoặc RBF (Radial Basis Function Kernel), và kernel sigmoid (Sigmoid Kernel).

Chủ đề chính của báo cáo này là ứng dụng Sigmoid SVM trong mô hình nhận dạng giọng nói nam và nữ. Nhận dạng giọng nói là một trong những ứng dụng quan trọng của học máy, đặc biệt là trong các hệ thống tự động như trợ lý ảo, hệ thống điều khiển bằng giọng nói, và các ứng dụng bảo mật. Việc phân biệt giọng nói nam và nữ không chỉ giúp cải thiện hiệu suất của các hệ thống này mà còn mở ra nhiều cơ hội nghiên cứu và phát triển trong lĩnh vực xử lý tín hiệu âm thanh.

Báo cáo này sẽ trình bày chi tiết về Sigmoid SVM, từ lý thuyết cơ bản đến các ứng dụng thực tế. Chúng tôi sẽ tiến hành các thí nghiệm để so sánh hiệu quả trong việc nhận dạng giọng nói nam và nữ trên các tập dữ liệu cụ thể, từ đó rút ra những kết luận về ưu và nhược điểm của loại kernel trong bài toán này. Qua đó, chúng tôi mong muốn cung cấp một cái nhìn toàn diện về khả năng ứng dụng của Sigmoid SVM trong học máy, góp phần vào sự phát triển của các hệ thống thông minh và tự động hóa trong tương lai.

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

1. Lý do lựa chọn đề tài này:

Việc phân biệt giọng nói nam và nữ là một thách thức quan trọng trong lĩnh vực nhận dạng giọng nói. Mặc dù đã có nhiều nghiên cứu về nhận dạng giọng nói, nhưng việc phát triển một mô hình có khả năng phân loại giọng nam và giọng nữ một cách chính xác vẫn còn là một lĩnh vực nghiên cứu đáng quan tâm. Điều này có thể đóng góp vào việc tạo ra các ứng dụng thực tế như hệ thống nhận dạng giọng nói có khả năng phân biệt giới tính, góp phần vào việc tăng cường trải nghiệm người dùng và hiệu quả của các ứng dụng dựa trên giọng nói. Do đó, đề tài này được lựa chọn với hy vọng đóng góp vào lĩnh vực nhận dạng giọng nói và ứng dụng của nó trong thực tế.

2. Phương pháp tiếp cận:

Trong thời gian gần đây, có nhiều phương pháp đã được áp dụng trong việc phân loại giọng nói, bao gồm các phương pháp truyền thống như Gaussian Mixture Models (GMM), Hidden Markov Models (HMM), và các phương pháp học máy hiện đại như Neural Networks (NN) và SVM.

- Gaussian Mixture Models (GMM): GMM là một phương pháp thống kê phổ biến được sử dụng trong nhận dạng giọng nói. Tuy nhiên, GMM thường đòi hỏi một số lượng lớn các tham số để mô hình hóa phân phối của dữ liệu, đặc biệt là khi xử lý dữ liệu có độ phức tạp cao. Điều này có thể dẫn đến hiệu suất kém khi áp dụng vào các bài toán phân loại với dữ liệu giọng nói phức tạp.
- Hidden Markov Models (HMM): HMM đã từng là phương pháp phổ biến trong nhận dạng giọng nói, nhưng nó thường đòi hỏi việc xác định và định cấu trúc của mô hình HMM phù hợp với từng loại dữ liệu cụ thể. Điều này đặt ra một thách thức lớn đối với việc xử lý dữ liệu giọng nói đa dạng và phức tạp.
- Neural Networks (NN): Các mạng nơ-ron đã chứng minh được hiệu suất cao trong nhiều lĩnh vực của học máy, bao gồm cả nhận dạng giọng nói. Tuy nhiên, việc huấn luyện mạng nơ-ron có thể đòi hỏi một lượng lớn dữ liệu và thời gian huấn luyện, đặc biệt là đối với các mô hình sâu.

Vì những lý do trên, chúng tôi chọn Sigmoid Support Vector Machine (SVM) cho đề tài này. SVM được biết đến với tính linh hoạt cao trong việc xử lý dữ liệu có số chiều cao và có khả năng tìm ra ranh giới phân loại tối ưu giữa các lớp dữ liệu. Đặc biệt, việc sử dụng hàm kernel trong SVM cho phép mô hình hoạt động hiệu quả trên dữ liệu phi tuyến. Hơn nữa, SVM thường ít phụ thuộc vào số lượng tham số và có khả năng tránh được hiện tượng quá khớp (overfitting) trong một số trường hợp. Do đó, SVM là lựa chọn phù hợp cho việc phân loại giọng nói nam và nữ từ dữ liệu âm thanh có tính phức tạp và đa dạng.

3. Dữ liệu sử dụng:

Để nghiên cứu về ứng dụng Sigmoid SVM trong việc nhận dạng giọng nói nam và nữ, chúng tôi đã thu thập một tập dữ liệu đa dạng từ nhiều nguồn khác nhau. Tập dữ liệu này bao gồm các mẫu giọng nói từ các đối tượng khác nhau, bao gồm cả nam và nữ, có độ tuổi và địa điểm khác nhau. Việc sử dụng tập dữ liệu đa dạng như vậy giúp đảm bảo tính đại diện và khả năng áp dụng của mô hình cho các điều kiện thực tế.

4. Tiến trình thực hiện:

Tiền xử lý dữ liệu: Đầu tiên, chúng tôi tiến hành tiền xử lý dữ liệu bằng cách trích xuất các đặc trưng từ tín hiệu âm thanh. Các đặc trưng này có thể bao gồm tần số cơ bản, biên độ, đặc điểm thời gian và các đặc trưng spectral khác. Quá trình trích xuất đặc trưng giúp chúng tôi biểu diễn dữ liệu âm thanh dưới dạng các vectơ đặc trưng có thể sử dụng để huấn luyện mô hình.

Huấn luyện mô hình SVM: Sau khi có các đặc trưng được trích xuất, chúng tôi tiến hành huấn luyện một mô hình Sigmoid Support Vector Machine (SVM). SVM được huấn luyện để phân loại giọng nói vào hai nhóm: nam và nữ. Quá trình này bao gồm việc sử dụng tập dữ liệu huấn luyện để điều chỉnh các tham số của mô hình và tối ưu hóa hiệu suất phân loại.

Đánh giá hiệu suất: Cuối cùng, chúng tôi đánh giá hiệu suất của mô hình bằng cách sử dụng các chỉ số đánh giá như độ chính xác, độ nhạy và độ đặc hiệu. Điều này giúp chúng tôi đánh giá khả năng phân loại giọng nói nam và nữ của mô hình trên tập dữ liệu kiểm thử và đánh giá tính khả thi của phương pháp đã đề xuất.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ MÔ HÌNH NGHIÊN CỨU

1. Cơ sở lý thuyết

a) Ma trận tương quan

Ma trận tương quan là một ma trận vuông R với kích thước $n \times n$, trong đó n là số lượng đặc trưng. Mỗi phần tử r_{ij} trong ma trận tương quan biểu diễn hệ số tương quan Pearson giữa đặc trưng i và đặc trưng j .

Hệ số tương quan Pearson: được tính bằng công thức:

$$r_{ij} = \frac{\text{Cov}(X_i, X_j)}{\sigma_{X_i} \sigma_{X_j}}$$

Trong đó:

$\text{Cov}(X_i, X_j)$ là hiệp phương sai giữa hai đặc trưng X_i và X_j , còn σ_{X_i} và σ_{X_j} là độ lệch chuẩn của X_i và X_j .

Giá trị của hệ số tương quan nằm trong khoảng từ -1 đến 1. Giá trị:

- $r_{ij} = 1$: Hai đặc trưng có mối tương quan dương hoàn hảo.
- $r_{ij} = -1$: Hai đặc trưng có mối tương quan âm hoàn hảo.
- $r_{ij} = 0$: Không có mối tương quan tuyến tính giữa hai đặc trưng.

Ý nghĩa:

- Ma trận tương quan giúp xác định mối quan hệ giữa các đặc trưng, từ đó có thể giảm chiều dữ liệu hoặc chọn lọc đặc trưng cho các mô hình học máy.
- Giúp nhận diện các đặc trưng có tương quan cao với nhau, từ đó tránh sự dư thừa thông tin khi huấn luyện mô hình.

2. Sigmoid SVM

a. Khái niệm

Support Vector Machines (SVM) là một thuật toán học máy phổ biến được sử dụng cho các bài toán phân loại. SVM hoạt động bằng cách tìm ra siêu phẳng tốt nhất để phân chia các điểm dữ liệu của các lớp khác nhau trong không gian đặc trưng. Một trong những hàm nhân (kernel function) có thể được sử dụng trong SVM là hàm nhân sigmoid, một hàm thường được sử dụng trong mạng nơ-ron nhân tạo.

Hàm nhân sigmoid trong SVM được định nghĩa như sau:

$$K(x_i, x_j) = \tanh(\alpha x_i^T x_j + c)$$

Trong đó:

- x_i và x_j là các vector đặc trưng đầu vào,
- α là một tham số độ dốc (slope parameter),
- c là một hằng số chặn (intercept constant),
- \tanh là hàm hyperbolic tangent.

b. Nguyên tắc hoạt động

Sigmoid SVM hoạt động theo các bước chính sau:

1. **Biến đổi không gian đặc trưng:** Hàm nhân sigmoid được sử dụng để biến đổi các điểm dữ liệu từ không gian đặc trưng ban đầu sang không gian cao hơn, nơi mà các điểm dữ liệu có thể dễ dàng phân tách hơn.
2. **Tìm kiếm siêu phẳng tối ưu:** Trong không gian đặc trưng đã biến đổi, SVM tìm kiếm siêu phẳng tối ưu để phân tách các điểm dữ liệu của các lớp khác nhau. Siêu phẳng này được xác định sao cho khoảng cách tới các điểm dữ liệu gần nhất (các vector hỗ trợ) là lớn nhất.
3. **Phân loại:** Sau khi tìm được siêu phẳng tối ưu, SVM có thể phân loại các điểm dữ liệu mới bằng cách xác định phía nào của siêu phẳng mà chúng nằm.

c. Ưu và nhược điểm:

Ưu điểm:

1. **Khả năng xử lý phi tuyến:** Hàm nhân sigmoid giúp SVM có thể xử lý các mối quan hệ phi tuyến giữa các đặc trưng và biến mục tiêu.
2. **Tính linh hoạt:** Sigmoid SVM có thể mô phỏng các quyết định phức tạp giống như mạng nơ-ron, làm cho nó hữu ích trong một số ứng dụng đặc biệt.

Nhược điểm:

1. **Khó khăn trong việc điều chỉnh tham số:** Các tham số α và c của hàm nhân sigmoid cần được điều chỉnh cẩn thận. Việc này có thể phức tạp và tốn thời gian.
2. **Hiệu suất thấp hơn trong nhiều trường hợp:** Trong thực tế, hàm nhân sigmoid thường không đạt hiệu suất cao bằng các hàm nhân khác như hàm nhân RBF hoặc hàm nhân đa thức (polynomial kernel).
3. **Quá trình tối ưu hóa phức tạp:** Quá trình tối ưu hóa trong Sigmoid SVM có thể chậm hơn, đặc biệt là với các tập dữ liệu lớn, do tính phi tuyến của hàm sigmoid.

d. Kết luận:

Sigmoid SVM là một công cụ hữu ích trong bộ công cụ SVM, đặc biệt khi khám phá các mối quan hệ phi tuyến trong dữ liệu và mô phỏng các quyết định phức tạp tương tự mạng nơ-ron. Tuy nhiên, do độ nhạy cảm với các tham số và hiệu suất có thể thấp hơn so với các hàm nhân khác trong nhiều trường hợp, Sigmoid SVM thường ít được sử dụng hơn. Việc hiểu rõ về Sigmoid SVM và cách sử dụng nó là điều quan trọng đối với những ai quan tâm đến SVM và các phương pháp kernel.

CHƯƠNG III: THIẾT KẾ, MÔ PHỎNG, KẾT QUẢ VÀ THẢO LUẬN

1. Tiền xử lý dữ liệu

1.1. Đọc dữ liệu:

```
df = pd.read_csv("../dataset/voice.csv")
df.head()
```

	meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	...	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx	label
0	0.059781	0.064241	0.032027	0.015071	0.090193	0.075122	12.863462	274.402905	0.893369	0.491918	...	0.059781	0.084279	0.015702	0.275862	0.007812	0.007812	0.000000	0.000000	male	
1	0.066009	0.067310	0.040229	0.019414	0.092966	0.073252	22.423285	634.613855	0.892193	0.513724	...	0.066009	0.107937	0.015826	0.250000	0.009014	0.007812	0.054688	0.046875	0.052632	male
2	0.077316	0.083829	0.036718	0.008701	0.131908	0.123207	30.757155	1024.927705	0.846389	0.478905	...	0.077316	0.098706	0.015656	0.271186	0.007990	0.007812	0.015625	0.007812	0.046512	male
3	0.151228	0.072111	0.158011	0.096582	0.207955	0.111374	1.232831	4.177296	0.963322	0.727232	...	0.151228	0.088965	0.017798	0.250000	0.201497	0.007812	0.562500	0.534688	0.247119	male
4	0.135120	0.079146	0.124656	0.078720	0.206045	0.127325	1.101174	4.333713	0.971955	0.783568	...	0.135120	0.106398	0.016931	0.266667	0.712812	0.007812	5.484375	5.476562	0.208274	male

5 rows x 21 columns

- Tập dữ liệu được đọc từ tệp CSV và kích thước của nó được kiểm tra.

1.2. Kiểm tra giá trị thiếu:

```
df.isnull().sum()
```

meanfreq	0
sd	0
median	0
Q25	0
Q75	0
IQR	0
skew	0
kurt	0
sp.ent	0
sfm	0
mode	0
centroid	0
meanfun	0
minfun	0
maxfun	0
meandom	0
mindom	0
maxdom	0
dfrange	0
modindx	0
label	0
dtype:	int64

- Xác minh rằng không có giá trị thiếu trong dữ liệu

1.3. Mã hóa nhãn:

```
from sklearn.preprocessing import LabelEncoder

X = df.drop('label', axis=1)
y = df['label']

gender_encoder = LabelEncoder()
y = gender_encoder.fit_transform(y)

array([1, 1, 1, ..., 0, 0, 0])
```

- Cột nhãn ('label') được mã hóa từ dạng chuỗi ('male', 'female') sang dạng số (0, 1).

1.4. Chuẩn hóa dữ liệu:

```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
scaler.fit(X)
X = scaler.transform(X)
```

- Dữ liệu được chuẩn hóa để đảm bảo các đặc trưng nằm trong cùng một phạm vi, giúp mô hình học máy hoạt động hiệu quả hơn.

2. Quy trình thực hiện

2.1. Tính toán hiển thị ma trận tương đương

```
# Chọn chỉ các cột số từ DataFrame
numeric_df = df.select_dtypes(include=['number'])

# Tính toán tương quan giữa các cột số
data_corr = numeric_df.corr()

# Vẽ biểu đồ heatmap
plt.figure(figsize=(20, 20))
sb.heatmap(data_corr, annot=True)
plt.show()
```



- Ma trận tương quan được tính toán giữa các cột số trong DataFrame và được hiển thị dưới dạng biểu đồ heatmap để nhận diện mối quan hệ giữa các đặc trưng.

2.2. Phân chia dữ liệu thành tập huấn luyện và tập kiểm tra:

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.35, random_state=1)
```

- Dữ liệu được chia thành tập huấn luyện (65%) và tập kiểm tra (35%).

2.3. Huấn luyện và đánh giá mô hình:

```
from sklearn.svm import SVC
```

```
svc=SVC(kernel='sigmoid') #Default hyperparameters
svc.fit(X_train,y_train)
y_pred=svc.predict(X_test)
print('Accuracy Score:')
print(metrics.accuracy_score(y_test,y_pred))
```

```
Accuracy Score:
0.7953110910730388
```

```
arr=[]
arr_g=np.arange(0.001,0.02,0.001)
for j in arr_g:
    svc=SVC(kernel='sigmoid',gamma=j)
    svc.fit(X_train,y_train)
    y_pred=svc.predict(X_test)
    arr.append(metrics.accuracy_score(y_test,y_pred))
```

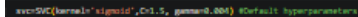
```
[ ] arr
```

```
[0.9224526600541028,
0.9567177637511272,
0.9648331830477908,
0.9666366095581606,
0.9666366095581606,
0.9657348963029756,
0.9657348963029756,
0.9666366095581606,
0.9657348963029756,
0.9657348963029756,
0.9657348963029756,
0.9648331830477908,
0.9639314697926059,
0.9639314697926059,
0.9585211902614968,
0.9567177637511272,
0.9540126239855726,
0.9477006311992786,
0.9440937781785392]
```

- Mô hình SVC với kernel 'sigmoid' được huấn luyện trên tập huấn luyện và đánh giá độ chính xác trên tập kiểm tra.

2.4. Tối ưu hóa tham số gamma:

[0.9226660144551373, 0.9542367128538913, 0.9624406021642773, 0.9646508006229286, 0.964334344926726, 0.9637034301002275, 0.9637024318172742, 0.9621251447510282, 0.



- Dải giá trị của tham số **gamma** được thử nghiệm, và độ chính xác chéo của mô hình được tính toán để tìm giá trị tối ưu của **gamma**.

2.5. Huấn luyện và đánh giá mô hình với tham số tối ưu:

```
svc=SVC(kernel='sigmoid',C=1.5, gamma=0.004) #Default hyperparameters
svc.fit(X_train,y_train)
y_pred=svc.predict(X_test)
print('Accuracy Score:')
print(metrics.accuracy_score(y_test,y_pred))
```

```
Accuracy Score:
0.9684400360685302
```

- Mô hình SVC được huấn luyện lại với các tham số tối ưu và đánh giá độ chính xác trên tập kiểm tra.

3. Kết quả và thảo luận

3.1. Độ chính xác của mô hình:

Mô hình đạt độ chính xác là 96.84%, cho thấy mô hình hoạt động hiệu quả trong việc phân loại giọng nói là nam hay nữ.

3.2. Thảo luận:

- **Hiệu quả của mô hình:** Với độ chính xác đạt được, mô hình **LinearSVC** cho thấy khả năng phân loại tốt. Tuy nhiên, có thể cần thêm các biện pháp đánh giá khác như ma trận nhầm lẫn (confusion matrix) để hiểu rõ hơn về các lỗi phân loại.
- **Tiền xử lý dữ liệu:** Các bước tiền xử lý như mã hóa nhãn và chuẩn hóa dữ liệu đóng vai trò quan trọng trong việc nâng cao hiệu quả của mô hình.
- **Các đặc trưng quan trọng:** Việc xác định và trực quan hóa các đặc trưng tương quan cao với nhãn giúp hiểu rõ hơn về mối quan hệ giữa các đặc trưng và nhãn, cung cấp thông tin giá trị cho việc cải thiện mô hình.

Tài liệu tham khảo

[1] [https://thongke.cesti.gov.vn/dich-vu-thong-ke/tai-lieu-phan-tich-thong-ke/861-thong-ke-mo-ta-trong-nghien-cuu-dai-luong-tuong-quan#:~:text=%E2%80%93%20Ma%20tr%E1%BA%ADn%20t%C6%B0%C6%A1ng%20quan%20\(Correlation,t%C6%B0%C6%A1ng%20quan%20gi%E1%BB%AFa%20hai%20bi%E1%BA%BFn](https://thongke.cesti.gov.vn/dich-vu-thong-ke/tai-lieu-phan-tich-thong-ke/861-thong-ke-mo-ta-trong-nghien-cuu-dai-luong-tuong-quan#:~:text=%E2%80%93%20Ma%20tr%E1%BA%ADn%20t%C6%B0%C6%A1ng%20quan%20(Correlation,t%C6%B0%C6%A1ng%20quan%20gi%E1%BB%AFa%20hai%20bi%E1%BA%BFn)

[2]

<https://www.geeksforgeeks.org/major-kernel-functions-in-support-vector-machine-svm/>