ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA TOÁN - CƠ - TIN HỌC

———————o0o——————–



**TÊN THÀNH VIÊN:**

* **ĐẬU VĂN AN - 21000659**
* **ĐẶNG HUY HOÀNG - 21000680**
* **ĐỖ MẠNH HÙNG - 21000682**

**LỜI MỞ ĐẦU**

Dữ liệu hình ảnh là một trong những tài nguyên quan trọng trong lĩnh vực khoa học máy tính và trí tuệ nhân tạo. Tập dữ liệu MNIST, chứa các hình ảnh chữ số viết tay từ 0 đến 9, đã trở thành nền tảng để phát triển và kiểm tra các thuật toán phân loại và nhận dạng mẫu. Trong đề tài này, chúng tôi sử dụng tập dữ liệu MNIST để thực hiện các thí nghiệm nhằm tìm hiểu và ứng dụng nhiều kỹ thuật học máy khác nhau, với mục tiêu cuối cùng là phân loại chính xác các hình ảnh số viết tay.

Từ việc giảm số chiều, phân cụm, đến sử dụng các mô hình phân loại như Naïve Bayes, ANN, và Logistic Regression, đề tài này hướng tới việc tìm hiểu hiệu quả của các phương pháp khác nhau trong việc xử lý và phân loại dữ liệu. Thông qua việc thực hiện các thí nghiệm trên tập dữ liệu do giáo viên cung cấp, chúng tôi mong muốn không chỉ khám phá cách thức hoạt động của các mô hình học máy mà còn rút ra những bài học về hiệu suất và độ chính xác.

Kết quả từ nghiên cứu này sẽ được đánh giá thông qua các chỉ số như độ chính xác (accuracy), ma trận nhầm lẫn (confusion matrix), recall và precision. Chúng tôi tin rằng việc so sánh và phân tích các kết quả này sẽ cung cấp những hiểu biết quan trọng về cách tiếp cận tốt nhất cho việc phân loại hình ảnh chữ số viết tay, từ đó đóng góp vào sự phát triển của các ứng dụng trong lĩnh vực trí tuệ nhân tạo và thị giác máy tính.

**MỤC LỤC**

[1. Giới thiệu đề tài 2](#_Toc165730986)

[1.1 Giới thiệu về đề tài 2](#_Toc165730987)

[1.2 Tổng quan về dữ liệu 2](#_Toc165730988)

[2. Giảm chiều dữ liệu với PCA và hiển thị trực quan 3](#_Toc165730989)

[2.1 Giới thiệu về PCA 3](#_Toc165730990)

[2.2 Phương pháp 3](#_Toc165730991)

[2.3 Kết Quả 4](#_Toc165730992)

[2.4 Kết Luận 5](#_Toc165730993)

[3. Sử dụng mô hình Naïve Bayes 5](#_Toc165730994)

[3.1 Giới thiệu mô hình Naïve Bayes 5](#_Toc165730995)

[3.2 Phân loại sử dụng Mô hình Naïve Bayes 6](#_Toc165730996)

[3.3 Kết quả và thảo luận 6](#_Toc165730997)

[3.4 Kết Luận 9](#_Toc165730998)

[4. Sử dụng mô hình ANN 9](#_Toc165730999)

[4.1 Giới thiệu mô hình 9](#_Toc165731000)

[4.2 Phương pháp 10](#_Toc165731001)

[4.3 Kết quả và thảo luận 11](#_Toc165731002)

[4.4 Kết luận 14](#_Toc165731003)

# 1. Giới thiệu đề tài

## 1.1 [Giới thiệu về](https://docs.google.com/document/d/1Cd3MMdAKW4BDcU7vc2UH0w4PrLfpwyNT/edit?fbclid=IwAR1peL7Fwx5sLLp5fAqjgZ_1kHSpZ6VXe4eRGHDlqhSCq7PxXiN3EQc0o9Q&pli=1#heading=h.1fob9te) đề tài

Việc phân loại ảnh là một trong những ứng dụng quan trọng của học máy và học sâu. Bộ dữ liệu MNIST, bao gồm các hình ảnh chữ số viết tay từ 0 đến 9, là một tập hợp phổ biến để thử nghiệm các thuật toán phân loại ảnh. Trong bài tiểu luận này, chúng tôi trình bày một quy trình sử dụng mạng nơ-ron nhân tạo (Artificial Neural Network - ANN) để phân loại các hình ảnh sau khi đã giảm số chiều bằng Phân tích Thành phần Chính (Principal Component Analysis - PCA). Chúng tôi sử dụng bộ dữ liệu MNIST để huấn luyện và kiểm tra mô hình ANN, đồng thời đánh giá hiệu suất bằng các chỉ số như độ chính xác, ma trận nhầm lẫn, recall, precision và F1-score.

## 1.2 Tổng quan về dữ liệu

Bộ dữ liệu MNIST bao gồm 60.000 hình ảnh cho tập huấn luyện và 10.000 hình ảnh cho tập kiểm tra

train-images-idx3-ubyte: training set images (dữ liệu ảnh train)

train-labels-idx1-ubyte: training set labels (dữ liệu nhãn ứng với ảnh train)

t10k-images-idx3-ubyte: test set images (dữ liệu ảnh test)

t10k-labels-idx1-ubyte: test set labels (dữ liệu nhãn ứng với ảnh test)

Tập huấn luyện chứa 60000 mẫu và tập test chứa 10000 mẫu. Chú ý dữ liệu ảnh các chữ số viết tay ở đây được lưu liên tiếp nhau và không theo định dạng ảnh

Cường độ Pixels được sắp xếp cạnh nhau thành dòng. Giá trị cường độ Pixel là từ 0 đến 255 (1 byte) nhưng để ngược: 0 là background (trắng),và 255 là foreground (đen), tuy nhiên khi in ảnh ra màn hình thì điều này không quan trọng.

Ở đây nhãn cho ảnh là số nguyên từ 0 đến 9 (ứng với chữ số trong ảnh).

Cấu trúc của tệp dữ liệu ảnh test (train-images-idx3-ubyte) và nhãn ảnh test (t10k-labels-idx1-ubyte) tương tự như với dữ liệu training, số lượng ảnh là 10000

# 2. Giảm chiều dữ liệu với PCA và hiển thị trực quan

## 2.1 Giới thiệu về PCA

Trong lĩnh vực học máy và thị giác máy tính, việc giảm chiều dữ liệu là một phương pháp quan trọng để làm giảm độ phức tạp của dữ liệu và tăng hiệu suất tính toán. Trong báo cáo này, chúng tôi sẽ giới thiệu việc giảm chiều dữ liệu của tập dữ liệu MNIST sử dụng phương pháp phân tích thành phần chính (PCA). Chúng tôi sẽ mô tả quá trình giảm chiều và trực quan hóa dữ liệu sau khi giảm chiều để hiểu rõ hơn về cách dữ liệu được biểu diễn trong không gian chiều thấp hơn.

## 2.2 Phương pháp

Chúng tôi sử dụng tập dữ liệu MNIST, một tập dữ liệu phổ biến trong lĩnh vực thị giác máy tính, bao gồm các hình ảnh về chữ số viết tay từ 0 đến 9. Đầu tiên, chúng tôi đã đọc dữ liệu từ các tệp gzip tương ứng với hình ảnh và nhãn. Sau đó, chúng tôi đã chuẩn hóa dữ liệu sử dụng phép chia tỷ lệ tối thiểu-tối đa để đảm bảo rằng tất cả các giá trị nằm trong khoảng từ 0 đến 1.

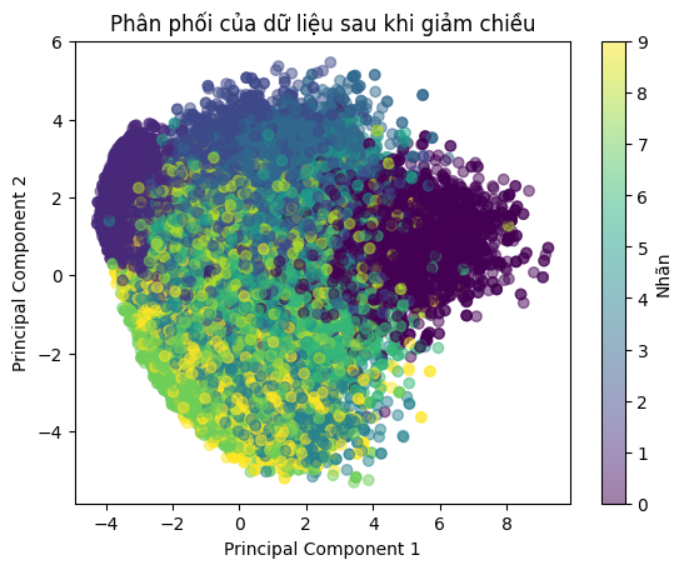
Tiếp theo, chúng tôi đã áp dụng phương pháp PCA để giảm chiều dữ liệu xuống còn 2 chiều. Quá trình này giúp chúng tôi giữ lại sự biến động chính của dữ liệu trong không gian chiều thấp hơn, đồng thời giảm độ phức tạp tính toán.

Cuối cùng, chúng tôi đã trực quan hóa dữ liệu sau khi giảm chiều bằng cách sử dụng biểu đồ phân tán, trong đó mỗi điểm dữ liệu được biểu diễn bởi 2 thành phần chính và được tô màu theo nhãn tương ứng.

## 2.3 Kết Quả

Dữ liệu sau khi giảm chiều bằng PCA đã cho thấy một phân phối rõ ràng của các lớp dữ liệu trong không gian 2 chiều. Mặc dù dữ liệu đã được giảm chiều xuống chỉ còn 2 chiều, nhưng chúng ta vẫn có thể nhận biết được sự phân tách giữa các lớp dữ liệu khác nhau.

Biểu đồ phân tán của dữ liệu sau khi giảm chiều cung cấp một cái nhìn tổng quan về sự phân bố của các lớp dữ liệu trong không gian chiều thấp hơn, giúp chúng ta hiểu rõ hơn về cấu trúc và tính chất của tập dữ liệu MNIST.



## 2.4 Kết Luận

Trong báo cáo này, chúng tôi đã giới thiệu phương pháp giảm chiều dữ liệu của tập dữ liệu MNIST bằng cách sử dụng PCA. Qua quá trình này, chúng tôi đã hiểu được cách dữ liệu được biểu diễn trong không gian chiều thấp hơn và làm thế nào phương pháp này có thể giúp giảm độ phức tạp tính toán và trực quan hóa dữ liệu một cách hiệu quả.

# 3. Sử dụng mô hình Naïve Bayes

## 3.1 Giới thiệu mô hình Naïve Bayes

* **Mô hình Naïve Bayes trong Học Máy:** Mô hình Naïve Bayes là một phương pháp phân loại trong học máy, dựa trên định lý Bayes. Phương pháp này được sử dụng để dự đoán lớp của một mẫu dựa trên các đặc trưng của nó. Mặc dù đơn giản, nhưng Naïve Bayes thường cho kết quả tốt và hiệu quả trong nhiều tình huống thực tế.
* **Cơ sở Toán học của Mô hình Naïve Bayes:** 
  + Xác suất có điều kiện: Naïve Bayes dựa trên xác suất có điều kiện để dự đoán lớp của một mẫu dữ liệu dựa trên các đặc trưng của nó. Sử dụng định lý Bayes, chúng ta có thể tính xác suất của mỗi lớp dựa trên các đặc trưng của mẫu.
  + Giả định về độc lập: Một trong những giả định chính của Naïve Bayes là tất cả các đặc trưng đều độc lập với nhau khi đã biết lớp. Mặc dù giả định này thường không hoàn toàn chính xác trong thực tế, nhưng nó giúp giảm độ phức tạp của mô hình và làm cho quá trình tính toán đơn giản hơn
* **Ưu điểm và Nhược điểm**:
  + Ưu điểm: Mô hình Naïve Bayes có tính đơn giản, dễ triển khai và hiệu quả trong nhiều tình huống. Nó đặc biệt hữu ích khi có ít dữ liệu huấn luyện và khi các đặc trưng có mối quan hệ độc lập với nhau.
  + Nhược điểm: Naïve Bayes có giả định về độc lập giữa các đặc trưng, điều này có thể không phù hợp với thực tế trong một số trường hợp. Ngoài ra, nếu có dữ liệu thiếu hoặc nhiễu, mô hình có thể cho kết quả không chính xác.

## 3.2 Phân loại sử dụng Mô hình Naïve Bayes

* Để thực hiện việc phân loại, chúng tôi đã xây dựng một chương trình sử dụng mô hình Naïve Bayes.
* Xây dựng chương trình: Đầu tiên, chúng tôi đã sử dụng thư viện Scikit-learn để xây dựng và triển khai mô hình Naïve Bayes. Thư viện này cung cấp một loạt các công cụ và thuật toán phổ biến để thực hiện các tác vụ học máy, bao gồm cả Naïve Bayes.
* Huấn luyện mô hình: Sau đó, chúng tôi đã sử dụng tập dữ liệu huấn luyện để huấn luyện mô hình Naïve Bayes. Trong quá trình này, mô hình đã học các quy tắc và mẫu từ tập dữ liệu, từ đó có thể dự đoán nhãn của các ảnh chữ số viết tay.
* Kiểm tra với tập validation: Cuối cùng, chúng tôi đã sử dụng tập dữ liệu validation để kiểm tra hiệu suất của mô hình. Bằng cách này, chúng tôi đánh giá khả năng tổng quát hóa của mô hình, tức là khả năng dự đoán đúng nhãn cho các dữ liệu mà mô hình chưa từng thấy trước đó.
* Kiểm tra và đánh giá hiệu suất: Sau khi huấn luyện, chúng tôi sử dụng tập kiểm tra để đánh giá hiệu suất của mô hình. Các chỉ số được sử dụng để đánh giá bao gồm độ chính xác (accuracy), ma trận nhầm lẫn (confusion matrix), recall, precision và F1-score.

## 3.3 Kết quả và thảo luận

Sau khi huấn luyện mô hình **Naïve Bayes**, chúng tôi đã kiểm tra hiệu suất trên tập kiểm tra và thu được các kết quả sau:

* **Độ chính xác**: Mô hình đạt được độ chính xác 83.55% trên tập validation, Điều này cho thấy mô hình đã học và tổng quát hóa tốt trên dữ liệu mới, đồng thời cung cấp một cơ sở đáng tin cậy cho việc phân loại dữ liệu ảnh chữ số viết tay.
* **Ma trận nhầm lẫn**: Ma trận nhầm lẫn được sử dụng để đánh giá hiệu suất của mô hình. Nó cho chúng ta biết được số lượng các dự đoán đúng và sai cho mỗi lớp. Đây là ma trận nhầm lẫn thu được từ mô hình Naïve Bayes:
  + Confusion Matrix:

[[ 901 0 5 4 3 11 16 0 32 0]

[ 0 1083 8 11 0 1 7 0 54 2]

[ 17 12 875 23 12 4 29 14 58 1]

[ 1 11 31 852 1 17 6 9 44 32]

[ 3 3 10 0 678 0 22 2 27 165]

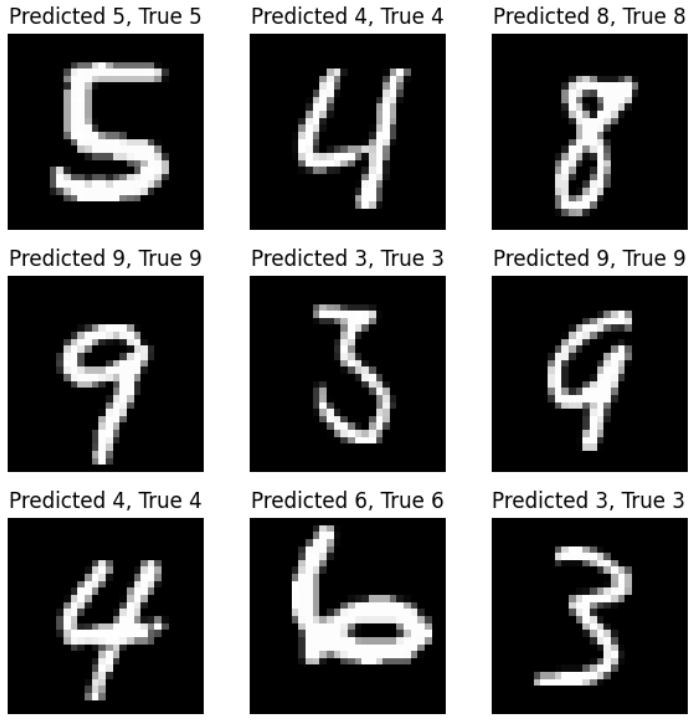
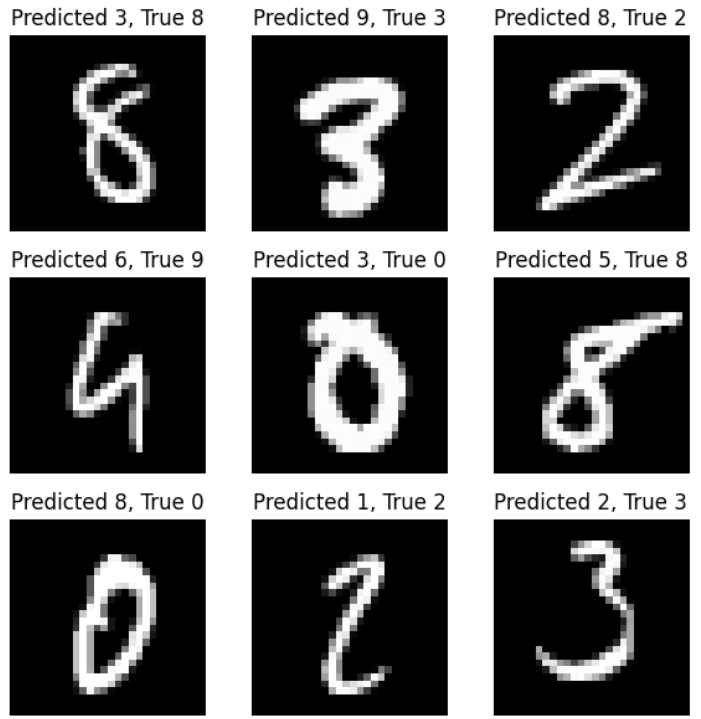
[ 23 9 5 99 20 568 20 7 71 38]

[ 11 14 11 0 2 26 926 0 18 0]

[ 2 22 13 6 18 0 3 872 37 85]

[ 1 26 13 57 13 33 8 18 783 30]

[ 7 6 4 7 66 9 0 19 31 846]]

* **Recall, precision, F1-score**: Các chỉ số này đều đạt mức trên 80%, cho thấy sự cân bằng tốt giữa việc xác định đúng các lớp và việc tránh nhầm lẫn. Cụ thể:
  + Recall (tỷ lệ khôi phục): 83.84%
  + Precision (độ chính xác): 84.54%
  + F1-Score: 83.90%
* Một số dự đoán của mô hình:
  + Dự đoán đúng:  
    
  + Dự đoán sai:  
    
  + Trong một số trường hợp, mô hình Naïve Bayes có thể dự đoán sai vì một số lý do sau:   
    Đặc trưng không phản ánh đầy đủ thông tin: Trong một số trường hợp, các đặc trưng được sử dụng để đào tạo mô hình không phản ánh đầy đủ thông tin để phân loại dữ liệu. Điều này có thể dẫn đến sự hiểu lầm hoặc sự khó khăn trong việc phân loại các trường hợp phức tạp

Mô hình quá đơn giản: Mô hình Naïve Bayes có giả định về sự độc lập giữa các đặc trưng, điều này có thể làm cho mô hình quá đơn giản để hiểu được mối quan hệ phức tạp giữa các đặc trưng.

Các kết quả trên cung cấp thông tin quan trọng về hiệu suất của mô hình Naïve Bayes trong việc phân loại dữ liệu ảnh chữ số viết tay, giúp chúng tôi hiểu rõ hơn về khả năng dự đoán của mô hình trên dữ liệu thực tế.

## 3.4 Kết Luận

* Trên cơ sở các kết quả và thảo luận trên, chúng tôi kết luận rằng mô hình Naïve Bayes là một lựa chọn hợp lý và hiệu quả trong việc phân loại dữ liệu ảnh chữ số viết tay. Mặc dù có một số trường hợp dự đoán sai, nhưng mô hình vẫn có khả năng tổng quát hóa tốt trên dữ liệu mới và cung cấp một cơ sở đáng tin cậy cho việc phân loại. Đồng thời, các kết quả và thảo luận cũng cung cấp thông tin quan trọng để hiểu rõ hơn về hiệu suất và giới hạn của mô hình.

# 4. Sử dụng mô hình ANN

## 4.1 Giới thiệu mô hình

MLPClassifier là một trong những thuật toán học máy quan trọng thuộc họ Mạng Nơ-ron Nhân tạo(Artificial Neural Networks - ANN). MLPClassifier là một mạng nơ-ron dạng feedforward (mạng truyền thẳng), có thể bao gồm một hoặc nhiều lớp ẩn (hidden layers). Mỗi lớp ẩn chứa một số lượng nơ-ron nhất định, và mỗi nơ-ron trong một lớp được kết nối với tất cả các nơ-ron trong lớp tiếp theo, một kiến trúc được gọi là "fully connected" hoặc "dense". MLPClassifier hoạt động dựa trên nguyên tắc lan truyền tiến (forward propagation) và lan truyền ngược (backward propagation). Trong lan truyền tiến, dữ liệu đầu vào được truyền qua các lớp, áp dụng các hàm kích hoạt (activation functions) để tạo ra đầu ra cuối cùng. Để huấn luyện mạng, lan truyền ngược được sử dụng để điều chỉnh các trọng số thông qua một quá trình gọi là "backpropagation", với mục tiêu giảm thiểu hàm mất mát (loss function). Quá trình này thường sử dụng các thuật toán tối ưu như Stochastic Gradient Descent (SGD) hoặc Adam

## 4.2 Phương pháp

1. Chuẩn hóa dữ liệu: Chúng tôi chuẩn hóa các giá trị pixel bằng cách sử dụng StandardScaler để đảm bảo dữ liệu có giá trị trung bình bằng 0 và độ lệch chuẩn

bằng 1

2. Giảm số chiều bằng PCA: Sau khi chuẩn hóa, chúng tôi áp dụng PCA để giảm số chiều của dữ liệu từ 784 xuống còn 100. PCA là một kỹ thuật được sử dụng để giảm số chiều bằng cách chọn những thành phần chính (principal components) có phương sai lớn nhất.

Giảm số chiều về 100 bằng phương pháp phân tích thành phần chính (PCA) trước khi huấn luyện mô hình ANN (Artificial Neural Network) là một chiến lược phổ biến trong học máy và khoa học dữ liệu, nhất là khi làm việc với dữ liệu hình ảnh. Sau đây là các lý do chính tại sao việc giảm số chiều trước khi huấn luyện ANN có thể hữu ích:

a. Giảm độ phức tạp và tránh overfitting:

Dữ liệu hình ảnh thường có rất nhiều chiều (chẳng hạn, 28x28 pixels cho dữ liệu MNIST, tương đương 784 chiều). Với số lượng chiều lớn như vậy, mô hình ANN có thể trở nên quá phức tạp và dễ bị overfitting, nghĩa là mô hình học quá mức chi tiết của dữ liệu huấn luyện và hoạt động kém trên dữ liệu chưa thấy trước đó (validation).

PCA giúp giảm số chiều và giữ lại các thành phần quan trọng nhất của dữ liệu. Điều này làm cho mô hình ANN trở nên đơn giản hơn và giảm nguy cơ overfitting.

b. Tăng tốc độ huấn luyện và tính toán:

ANN là loại mô hình phức tạp với nhiều tham số. Khi số chiều dữ liệu đầu vào lớn, số lượng tham số của mô hình tăng lên, dẫn đến việc huấn luyện chậm và đòi hỏi nhiều tài nguyên tính toán.

Giảm số chiều giúp giảm khối lượng tính toán, tăng tốc độ huấn luyện và đòi hỏi ít tài nguyên hơn.

c. Giảm nhiễu (noise) và cải thiện hiệu suất:

Dữ liệu với nhiều chiều có thể chứa nhiều nhiễu (noise). PCA giúp loại bỏ các chiều ít thông tin, tập trung vào các chiều mang nhiều thông tin nhất.

Bằng cách tập trung vào các thành phần chính, mô hình ANN có khả năng hoạt động hiệu quả hơn trên dữ liệu với ít nhiễu hơn.

c. Huấn luyện mô hình ANN: Chúng tôi sử dụng MLPClassifier từ thư viện scikit-learn để xây dựng mô hình ANN với một lớp ẩn gồm 100 nơ-ron. Mô hình được huấn luyện với tập dữ liệu đã giảm số chiều.

d. Kiểm tra và đánh giá hiệu suất: Sau khi huấn luyện, chúng tôi sử dụng tập kiểm tra để đánh giá hiệu suất của mô hình. Các chỉ số được sử dụng để đánh giá bao gồm độ chính xác (accuracy), ma trận nhầm lẫn (confusion matrix), recall, precision và F1-score.

## 4.3 Kết quả và thảo luận

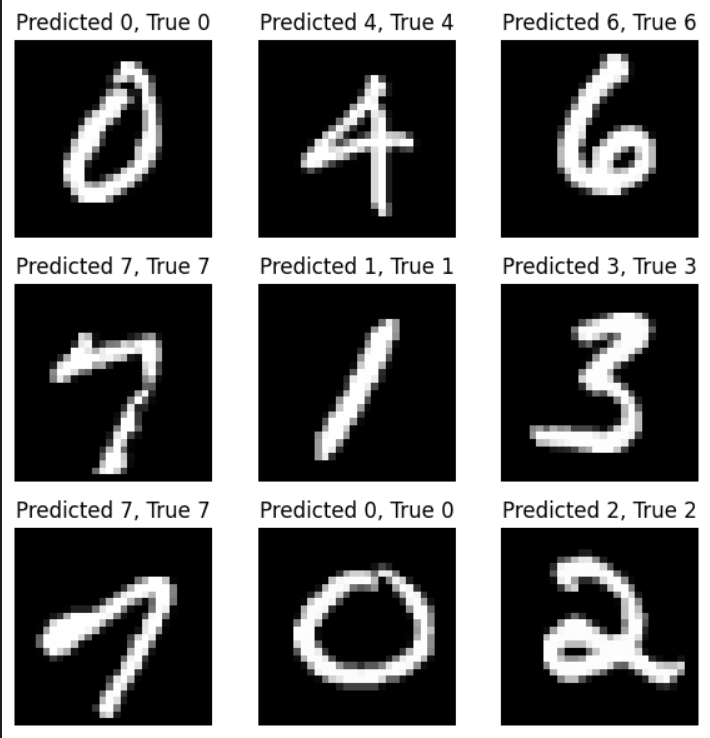
Sau khi huấn luyện mô hình ANN, chúng tôi đã kiểm tra hiệu suất trên tập kiểm tra và thu được các kết quả sau:

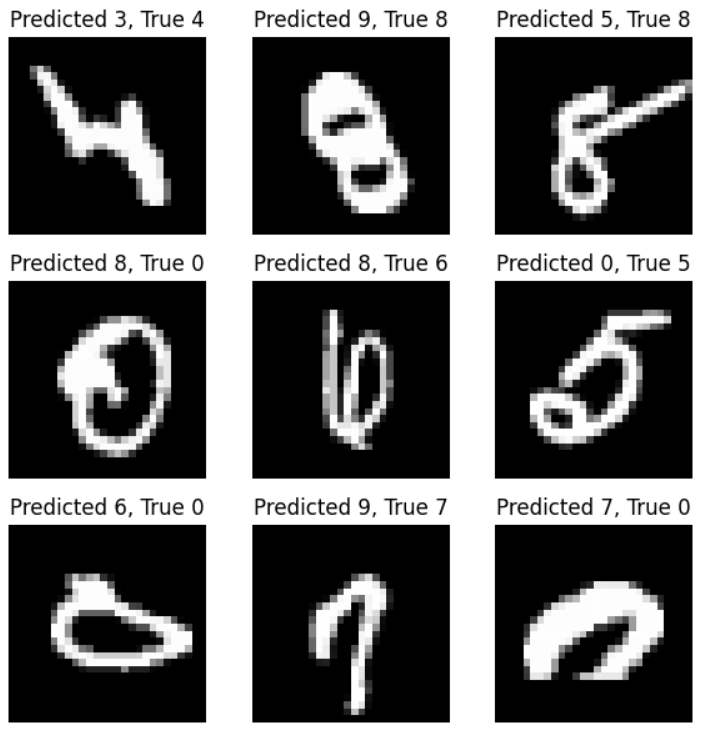
* **Độ chính xác**: Độ chính xác tổng thể của ANN là 96.93%. Đây là một mức độ chính xác cao, cho thấy mô hình đã học được các mẫu dữ liệu một cách tốt và có thể phân loại chính xác hầu hết các hình ảnh trong tập kiểm tra.
* **Ma trận nhầm lẫn**: Ma trận nhầm lẫn cho phép chúng ta xem các lỗi nhầm lẫn cụ thể giữa các lớp. Dưới đây là một số quan sát:
  + Lớp 0 và 1: Mô hình hoạt động rất tốt, không có hoặc rất ít lỗi nhầm lẫn.
  + Lớp 2, 3, 4: Có một số lỗi nhầm lẫn, nhưng số lượng không đáng kể.
  + Lớp 5, 6, 7, 8, 9: Xuất hiện nhiều nhầm lẫn hơn so với các lớp khác. Ví dụ, lớp 5 có số lỗi cao hơn so với các lớp khác, với 14 lần nhầm lẫn.
* **Recall, precision, F1-score**: Recall, Precision, và F1-Score đều đạt 96.93%, phản ánh sự cân bằng tốt giữa việc nhận diện đúng các lớp và tránh nhầm lẫn. F1-Score, là sự cân bằng giữa recall và precision, cho thấy hiệu suất tổng thể của mô hình.

Nhìn chung, mô hình ANN đã thể hiện hiệu suất cao trong việc phân loại ảnh MNIST sau khi giảm số chiều bằng PCA. Việc giảm số chiều giúp tăng tốc quá trình huấn luyện và giảm nguy cơ overfitting, trong khi vẫn duy trì độ chính xác cao.

Một vài dự đoán của mô hình:

* Dự đoán đúng:



* Dự đoán sai:
* 
* Các trường hợp dự đoán sai là do đặc điểm của chữ số viết tay (quá xấu, khó nhận dạng được, hoặc các số được viết gần giống nhau). Nói chung mô hình ANN đã làm rất tốt việc nhận dạng chữ số viết tay trên bộ dữ liệu MNIST

## 4.4 Kết luận

Việc sử dụng PCA để giảm số chiều kết hợp với ANN là một phương pháp hiệu quả để phân loại ảnh. Trong bối cảnh bộ dữ liệu MNIST, mô hình ANN đã đạt được độ chính xác và các chỉ số hiệu suất cao. Các kết quả này cho thấy tiềm năng của việc sử dụng các kỹ thuật học sâu và giảm số chiều trong các bài toán phân loại ảnh phức tạp hơn.