MỤC LỤC

[**1.** **Giới thiệu** 2](#_Toc183390119)

[**2.** **Bộ dữ liệu maintance** 2](#_Toc183390120)

[**3.** **Cân bằng dữ liệu SMOTE - Synthetic Minority Over-sampling Technique** 3](#_Toc183390121)

[**4.** **Chuẩn hóa Minmax** 4](#_Toc183390122)

[**5.** **Mối tương quan giữa các đặc trưng** 5](#_Toc183390123)

[**6.** **Triển khai trên bộ dữ liệu** 6](#_Toc183390124)

[**7.** **Phân tích hiệu suất của các thuật toán** 7](#_Toc183390125)

[**i.** **Support Vector Classification (SVC)** 8](#_Toc183390126)

[**ii.** **K-nearest Neighbor (KNN)** 9](#_Toc183390127)

[**8.** **Phân tích độ nhạy của các thuật toán được đề xuất** 10](#_Toc183390128)

[**9.** **Thảo luận** 11](#_Toc183390129)

1. **Giới thiệu**

Các quy trình sản xuất và công nghiệp đóng vai trò cốt lõi trong việc tạo ra hàng hóa và dịch vụ cần thiết cho xã hội. Chúng bao gồm các chuỗi hoạt động có tổ chức nhằm biến nguyên vật liệu thành các sản phẩm hoàn chỉnh hoặc bán thành phẩm, được tối ưu hóa nhằm đạt hiệu suất cao, chất lượng ổn định và chi phí thấp nhất có thể.

AI4I, hay Artificial Intelligence for Industry, là một nền tảng và chương trình của Siemens được thiết kế nhằm thúc đẩy sự phát triển và ứng dụng AI trong các quy trình sản xuất và công nghiệp. AI4I cung cấp các công cụ, dữ liệu và nền tảng cần thiết để các công ty có thể triển khai các ứng dụng AI vào quy trình sản xuất của họ, giúp cải thiện hiệu suất, giảm thiểu lãng phí và tối ưu hóa chi phí.

AI4I tập trung vào các giải pháp AI như bảo trì dự đoán (predictive maintenance), phát hiện lỗi trong chuỗi sản xuất, tối ưu hóa quy trình và phân tích dữ liệu sản xuất theo thời gian thực. Những giải pháp này không chỉ giúp doanh nghiệp nâng cao chất lượng sản phẩm mà còn giúp cải thiện hiệu suất sử dụng thiết bị và tài nguyên.

Hiện nay, các ngành công nghiệp ngày càng ứng dụng công nghệ như AI, robot, IoT, và phân tích dữ liệu lớn để tối ưu hóa quy trình. Những công nghệ này không chỉ giúp tăng cường hiệu suất, mà còn cải thiện độ chính xác, giảm thiểu lãng phí và giúp quy trình trở nên linh hoạt hơn.

1. **Bộ dữ liệu maintance**

Bộ dữ liệu AI4I (AI for Industry 4.0) là một tập dữ liệu được xây dựng với mục tiêu hỗ trợ việc nghiên cứu và phát triển các giải pháp trí tuệ nhân tạo (AI) trong lĩnh vực công nghiệp, đặc biệt là công nghiệp sản xuất thông minh. Bộ dữ liệu này chứa các thông tin liên quan đến các quá trình sản xuất và hoạt động của máy móc trong môi trường sản xuất công nghiệp, bao gồm các chỉ số về trạng thái hoạt động, hiệu suất, và các yếu tố gây lỗi của thiết bị.

Bộ dữ liệu AI4I có các cột thông tin như sau:

* **UDI** (Unique Device Identifier): Mã định danh duy nhất của thiết bị, giúp nhận diện từng mẫu dữ liệu một cách riêng biệt.
* **Product ID**: Mã sản phẩm được sản xuất, có thể liên quan đến loại sản phẩm hoặc lô sản phẩm cụ thể.
* **Type**: Loại sản phẩm hoặc danh mục của sản phẩm, phân biệt các dòng sản phẩm khác nhau trong quá trình sản xuất.
* **Air Temperature [K]**: Nhiệt độ không khí xung quanh máy móc trong quá trình sản xuất, tính bằng độ Kelvin (K).
* **Process Temperature [K]**: Nhiệt độ tại điểm hoạt động của quá trình sản xuất, tính bằng độ Kelvin (K). Thường cao hơn nhiệt độ không khí do phát sinh nhiệt từ hoạt động máy móc.
* **Rotational Speed [rpm]**: Tốc độ quay của động cơ, tính bằng vòng/phút (rpm), thể hiện tốc độ hoạt động của máy móc.
* **Torque [Nm]**: Mô-men xoắn (torque), tính bằng Newton-mét (Nm), thể hiện lực quay của máy móc trong quá trình vận hành.
* **Tool Wear [min]**: Độ mòn của dụng cụ sản xuất, tính bằng phút (min), biểu thị thời gian công cụ đã được sử dụng trong sản xuất. Đây là thông số quan trọng để đánh giá tình trạng của công cụ và khả năng cần bảo trì hoặc thay thế.
* **Target**: Nhãn mục tiêu (target), xác định xem một sản phẩm có đạt tiêu chuẩn hay không. Đây có thể là biến nhị phân, cho biết sản phẩm đã qua kiểm tra chất lượng (đạt hoặc không đạt).
* **Failure Type**: Loại lỗi gặp phải (nếu có) trong quá trình sản xuất. Các giá trị trong cột này có thể bao gồm:
  + Không lỗi (No Failure)
  + Lỗi quá tải công cụ (Tool Wear Failure)
  + Lỗi nguồn nhiệt (Heat Dissipation Failure)
  + Lỗi công cụ (Power Failure)
  + Lỗi bôi trơn (Overstrain Failure)
  + Lỗi môi trường (Random Failures)

Bộ dữ liệu này có thể được dùng để xây dựng các mô hình AI và phân tích dữ liệu, nhằm dự đoán lỗi hoặc tối ưu hóa hiệu suất của máy móc. Các cột **Rotational Speed**, **Torque**, và **Tool Wear** có thể đóng vai trò quan trọng trong việc phân tích sự hao mòn của máy móc và dự đoán thời điểm bảo trì cần thiết. Các cột **Air Temperature** và **Process Temperature** giúp đánh giá sự ảnh hưởng của nhiệt độ đến hiệu suất hoạt động và tuổi thọ của thiết bị.

1. **Cân bằng dữ liệu SMOTE - Synthetic Minority Over-sampling Technique**

SMOTE (Synthetic Minority Over-sampling Technique) là một phương pháp cân bằng dữ liệu trong các bài toán phân loại mất cân bằng. Phương pháp này tạo thêm các mẫu tổng hợp từ lớp thiểu số (minority class) thay vì chỉ nhân bản ngẫu nhiên, giúp cải thiện hiệu suất của các mô hình học máy khi có sự mất cân bằng giữa các lớp.

**Cách hoạt động của SMOTE**

* **Chọn các điểm dữ liệu từ lớp thiểu số**: SMOTE bắt đầu bằng cách chọn ngẫu nhiên một điểm dữ liệu thuộc lớp thiểu số.
* **Xác định hàng xóm gần nhất**: Sau khi chọn một điểm từ lớp thiểu số, SMOTE tìm k hàng xóm gần nhất của điểm này trong cùng lớp thiểu số. Số lượng hàng xóm k thường được chọn là 5, nhưng có thể điều chỉnh tùy vào tình huống.
* **Tạo các điểm tổng hợp**: Một điểm ngẫu nhiên từ các hàng xóm gần nhất được chọn và SMOTE tạo một điểm tổng hợp mới bằng cách nội suy giữa điểm ban đầu và điểm hàng xóm. Quá trình nội suy được thực hiện bằng cách:

A group of symbols on a white background

Description automatically generated

trong đó  là một số ngẫu nhiên trong khoảng [0,1].

Kết quả là một điểm mới nằm trên đoạn thẳng nối giữa hai điểm, thuộc không gian của lớp thiểu số.

* **Lặp lại**: Quy trình trên được lặp lại cho đến khi đạt được số lượng mẫu mong muốn để cân bằng với lớp đa số.
* **Borderline-SMOTE**

Borderline-SMOTE (Borderline Synthetic Minority Over-sampling Technique) là một biến thể của thuật toán SMOTE (Synthetic Minority Over-sampling Technique), được sử dụng để cân bằng dữ liệu mất cân bằng trong các bài toán phân loại. Trong các bài toán này, số lượng mẫu của lớp thiểu số (minority class) thường nhỏ hơn rất nhiều so với lớp đa số (majority class), làm giảm độ chính xác và độ tin cậy của mô hình. Borderline-SMOTE tập trung vào việc tạo mẫu tổng hợp cho các điểm dữ liệu nằm gần biên giới giữa các lớp, nơi mà khả năng nhầm lẫn giữa các lớp thường cao hơn.

**Cách hoạt động của Borderline-SMOTE**

1. **Xác định các mẫu khó phân loại**: Đầu tiên, Borderline-SMOTE xác định các mẫu của lớp thiểu số nằm gần biên giới phân chia giữa lớp thiểu số và lớp đa số, cụ thể là các mẫu có nguy cơ bị nhầm lẫn cao hơn. Các mẫu này thường nằm gần các điểm của lớp đa số.
2. **Phân loại các điểm biên**: Những điểm dữ liệu thuộc lớp thiểu số gần kề với lớp đa số được chọn làm các điểm biên giới (borderline). Borderline-SMOTE xác định những điểm này bằng cách sử dụng một số tiêu chí như số lượng hàng xóm gần nhất thuộc lớp đa số. Những mẫu này có nguy cơ cao bị phân loại sai trong quá trình học của mô hình.
3. **Tạo các mẫu tổng hợp**: Sau khi xác định các điểm biên giới, Borderline-SMOTE tạo ra các điểm tổng hợp mới bằng cách nội suy giữa các điểm biên này với các hàng xóm gần nhất của chúng (thuộc lớp thiểu số). Điều này giúp tăng số lượng mẫu ở các khu vực quan trọng nhất (biên giới giữa các lớp) thay vì tăng số lượng mẫu trên toàn bộ không gian lớp thiểu số như SMOTE gốc.
4. **Chuẩn hóa Minmax**

Chuẩn hóa Min-Max (Min-Max Normalization) là một phương pháp được sử dụng trong tiền xử lý dữ liệu nhằm đưa các giá trị của thuộc tính về một khoảng nhất định, thường là từ 0 đến 1.

Giả sử có một tập dữ liệu với một thuộc tính x, và giá trị của thuộc tính này nằm trong khoảng từ ​ đến ​. Chuẩn hóa Min-Max đưa các giá trị của x về khoảng [0,1] bằng công thức:

A white background with black text

Description automatically generated

Chuẩn hóa Minimax đặc biệt hữu ích trong ML, nơi các đặc điểm có quy mô hoặc phạm vi lớn có thể chi phối quá trình phân tích và tạo ra kết quả không chính xác. Bằng cách áp dụng chuẩn hóa Minimax cho từng đặc điểm trong tập dữ liệu, chúng ta có thể đảm bảo rằng tất cả các đặc điểm đều có đóng góp như nhau cho quá trình phân tích, bất kể quy mô hoặc phạm vi ban đầu của chúng.

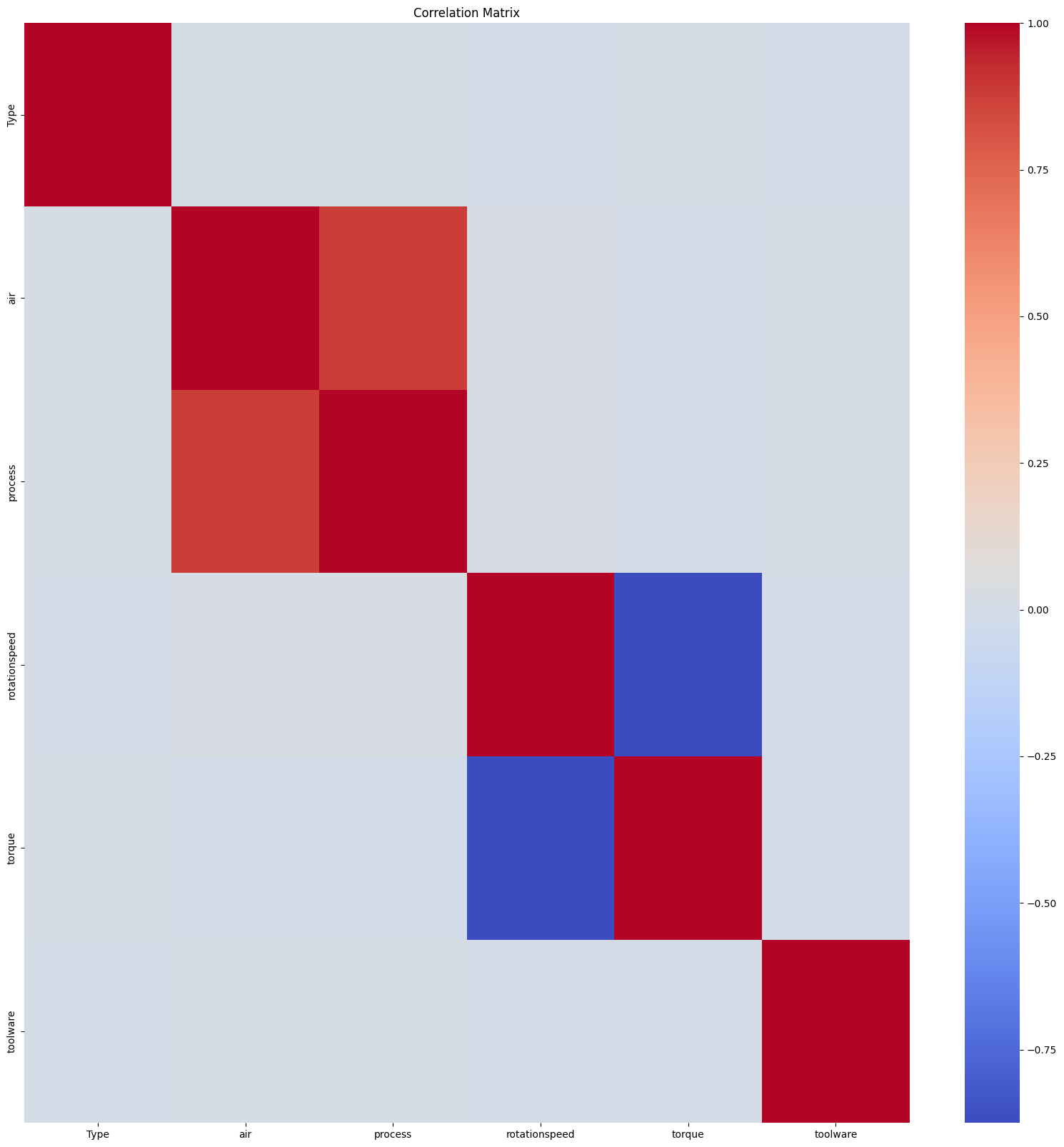
1. **Mối tương quan giữa các đặc trưng**

Ma trận tương quan là một công cụ thống kê được sử dụng để đo lường và thể hiện mức độ tương quan giữa các biến trong một tập dữ liệu.

Trong ma trận này:

* Màu đỏ đậm đại diện cho mối tương quan dương cao (gần +1), có nghĩa là khi một biến tăng, biến kia cũng có xu hướng tăng.
* Màu xanh đậm đại diện cho mối tương quan âm cao (gần -1), có nghĩa là khi một biến tăng, biến kia có xu hướng giảm.
* Màu xám hoặc màu nhạt gần như không có mối tương quan (giá trị gần 0).

Dưới đây là ma trận tương quan thể hiện mối tương quan giữa các đặc trưng của bộ dữ liệu:



Dựa trên ma trận tương qua, ta thấy:

* Biến **air** và **process** có mối tương quan dương cao, cho thấy nhiệt độ không khí và nhiệt độ quá trình có quan hệ chặt chẽ với nhau.
* Biến **rotational speed** và **torque** có mối tương quan âm, tức là khi tốc độ quay tăng, mô-men xoắn có xu hướng giảm.
* Biến **tool wear** có mối tương quan thấp với các biến còn lại, nên có thể độc lập hơn và không ảnh hưởng nhiều đến các biến kia.

1. **Triển khai trên bộ dữ liệu**

****

* Loại bỏ các đăc trưng tên gọi

A screenshot of a computer error

Description automatically generated

A white background with red text

Description automatically generated

* Loại bỏ đặc trưng Failure Type vì chênh lệch giữa các giá trị

A computer code with black text

Description automatically generated

* Chuyển các đặc trưng phân loại về dạng số

A close-up of a text

Description automatically generated

A screen shot of a computer code

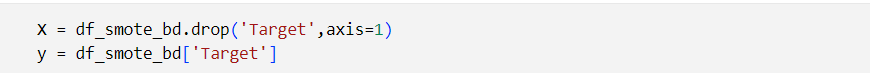
Description automatically generated

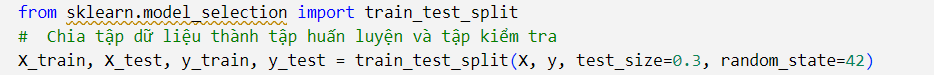
* Áp dụng BorderlineSMOTE để xử lí sự mất cân bằng dữ liệu

A screenshot of a computer code

Description automatically generated

* Áp dụng Chuẩn hóa Min-Max với những đặc trưng không phải đặc trưng phân loại





A screenshot of a computer

Description automatically generated

A screenshot of a computer program

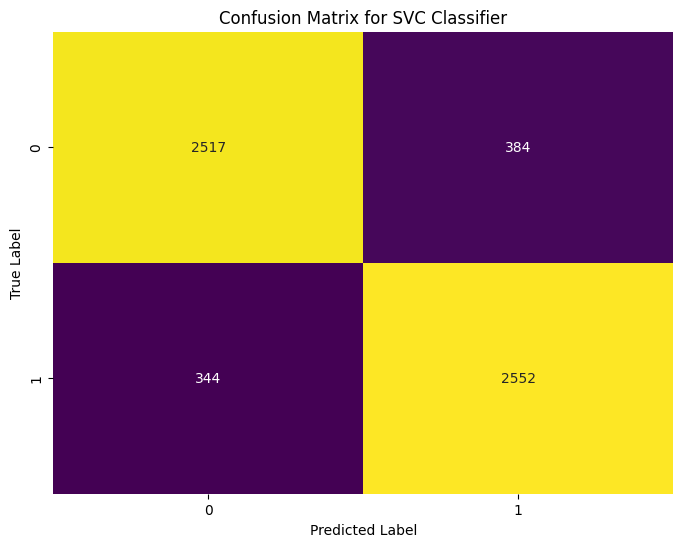
Description automatically generated

* Khởi chạy các mô hình

1. **Phân tích hiệu suất của các thuật toán**

Báo cáo sử dụng 2 thuật toán dựa trên ML, cụ thể là SVC và KNN để dự đoán lỗi của máy móc. Để kiểm tra hiệu quả của các thuật toán, báo cáo sử dụng bộ dữ liệu gồm 10000 mẫu máy móc gồm máy móc lỗi và máy móc bình thường. Các tập dữ liệu này đã cung cấp các điểm chuẩn quan trọng để đánh giá các thuật toán dựa trên ML. Tuy nhiên, chất lượng và tính đa dạng của dữ liệu đào tạo cũng như các tính năng và tham số của thuật toán có thể làm giảm khả năng xác định các ứng dụng độc hại của chúng. Các mô hình dựa trên ML được đề xuất đã chứng minh là một phương pháp hiệu quả để phát hiện các máy móc lỗi.

* 1. **Support Vector Classification (SVC)**

A graph of a line

Description automatically generated

A graph of a line

Description automatically generated

* + 1. Ma trận nhầm lẫn
    2. Biểu đồ ROC – AUC
    3. Biểu đồ Precison-Recall

Hiệu suất của mô hình SVC

| **Chỉ số** | **Precision** | **Recall** | **F1-score** | **Accuracy** |
| --- | --- | --- | --- | --- |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Giá trị** | 0.869 | 0.881 | 0.875 | 0.874 |

1. **Ma trận nhầm lẫn**:
   * Mô hình SVC đã phân loại chính xác phần lớn các mẫu trong tập dữ liệu, với 2517 và 2502 là số lượng dự đoán đúng cho mỗi lớp.
   * Tuy nhiên, vẫn còn có một số dự đoán sai, với 364 mẫu của lớp 0 bị dự đoán thành lớp 1 và 316 mẫu của lớp 1 bị dự đoán thành lớp 0.
2. **Biểu đồ ROC – AUC**:
   * Đường cong ROC gần sát với góc trên bên trái của biểu đồ, cho thấy mô hình có khả năng phân biệt tốt giữa các lớp.
   * Giá trị AUC đạt 0.95, là một con số cao, chứng tỏ mô hình có độ nhạy cao trong việc phân biệt các lớp.
3. **Biểu đồ Precision-Recall**:
   * Đường cong Precision-Recall cho thấy mô hình đạt độ chính xác và độ nhạy cao khi tỷ lệ Recall ở mức cao.
   * Diện tích dưới đường cong (AP = 0.94) cũng cho thấy khả năng của mô hình trong việc xử lý lớp thiểu số hoặc mất cân bằng trong dữ liệu.
4. **Hiệu suất của mô hình**:
   * Các chỉ số đánh giá của mô hình như Precision (0.869), Recall (0.881), F1-score (0.875), và Accuracy (0.874) đều cho thấy mô hình có hiệu suất tốt và ổn định.
   * Precision và Recall có giá trị khá gần nhau, điều này cho thấy mô hình có sự cân bằng giữa việc nhận diện đúng các mẫu và giảm thiểu dự đoán sai.

**Tổng kết:**

Mô hình SVC này đạt hiệu suất khá tốt trên tập dữ liệu hiện tại. Tuy nhiên, vẫn có thể tối ưu thêm để giảm số lượng mẫu bị phân loại sai, chẳng hạn bằng cách điều chỉnh các tham số của mô hình hoặc sử dụng các kỹ thuật tăng cường dữ liệu nếu có sự mất cân bằng giữa các lớp.

* 1. **K-nearest Neighbor (KNN)**

A graph of a line

Description automatically generatedA yellow and purple squares with numbers

Description automatically generated A graph with a line

Description automatically generated

* + 1. Ma trận nhầm lẫn
    2. Biểu đồ ROC – AUC
    3. Biểu đồ Precison-Recall

**Hiệu suất của mô hình KNN**

| **Chỉ số** | **Precision** | **Recall** | **F1-score** | **Accuracy** |
| --- | --- | --- | --- | --- |
| **Giá trị** | 0.943 | 0.987 | 0.964 | 0.963 |

1. **Ma trận nhầm lẫn**:
   * Mô hình KNN phân loại đúng phần lớn các mẫu trong tập dữ liệu, với 2727 và 2857 là số lượng dự đoán đúng cho mỗi lớp.
   * Tuy nhiên, vẫn có một số lượng nhỏ mẫu bị phân loại sai, với 174 mẫu của lớp 0 bị dự đoán thành lớp 1 và 39 mẫu của lớp 1 bị dự đoán thành lớp 0, cho thấy mô hình có hiệu suất tốt hơn ở lớp 1.
2. **Biểu đồ ROC – AUC**:
   * Đường cong ROC sát với góc trên bên trái của biểu đồ, cho thấy mô hình có khả năng phân biệt giữa các lớp khá tốt.
   * Giá trị AUC đạt 0.99, cho thấy mô hình KNN có độ chính xác cao trong việc phân loại và khả năng phân biệt hai lớp mạnh mẽ.
3. **Biểu đồ Precision-Recall**:
   * Đường cong Precision-Recall cho thấy mô hình duy trì độ chính xác và độ nhạy cao trong phần lớn các giá trị Recall.
   * Diện tích dưới đường cong Precision-Recall (AP = 0.98) cho thấy mô hình hoạt động hiệu quả trong việc phân loại các mẫu của lớp thiểu số.
4. **Hiệu suất của mô hình**:
   * Các chỉ số Precision (0.943), Recall (0.987), F1-score (0.964), và Accuracy (0.963) đều cho thấy mô hình KNN đạt hiệu suất rất cao và ổn định.
   * Giá trị Recall cao (0.987) cho thấy mô hình ít bỏ sót các mẫu dương tính, trong khi Precision cũng cao, cho thấy mô hình ít dự đoán sai các mẫu âm tính.

**Tổng kết:**

Mô hình KNN cho thấy hiệu suất rất tốt trên tập dữ liệu hiện tại, với độ chính xác và khả năng phân loại rất cao. Các chỉ số đều ở mức tốt, đặc biệt là Recall và F1-score, cho thấy mô hình có thể cân bằng giữa độ chính xác và độ nhạy.

1. **Phân tích độ nhạy của các thuật toán được đề xuất**

Để đánh giá độ chính xác dự đoán của các thuật toán dựa trên ML, báo cáo này đã sử dụng một số số liệu thống kê, bao gồm lỗi tuyệt đối trung bình (MAE), lỗi bình phương trung bình (MSE) và R bình phương (R2). Các lỗi dự đoán thu được giữa lớp mục tiêu và các giá trị dự đoán được trình bày trong bảng dưới đây. Phân tích được trình bày cung cấp những hiểu biết có giá trị về hiệu suất của các mô hình dựa trên ML, cho phép chúng tôi xác định và giải quyết mọi lỗi dự đoán.

| **Chỉ số** | **Mô hình SVC** | **Mô hình KNN** |
| --- | --- | --- |

|  |  |  |
| --- | --- | --- |
| **MAE** | **0.1256** | **0.0367** |

|  |  |  |
| --- | --- | --- |
| **MSE** | **0.1256** | **0.0367** |

|  |  |  |
| --- | --- | --- |
| **R-squared (R²)** | **0.4977** | **0.8530** |

* **MAE và MSE:**
  + Mô hình KNN có MAE và MSE thấp hơn SVC, nghĩa là sai số trung bình của KNN nhỏ hơn, dữ đoán chính xác hơn trong việc dự đoán giá trị thực tế.
* **R²-score:**
  + Hệ số R² của KNN cao hơn nhiều, cho thấy KNN giải thích được phần lớn biến thiên của dữ liệu tốt hơn.

**Tổng kết:**

* Mô hình **KNN** vượt trội hơn so với SVC trên cả ba chỉ số: MAE, MSE và R², thể hiện khả năng dự đoán chính xác và ổn định hơn trên tập dữ liệu hiện tại.
* Với R² cao (0.8530), KNN cho thấy mô hình này phù hợp hơn và có khả năng giải thích biến thiên của dữ liệu cao hơn nhiều so với SVC (0.4977).

1. **Thảo luận**

Trong việc dự đoán lỗi máy móc, Machine Learning (ML) đóng vai trò quan trọng giúp doanh nghiệp giảm thiểu rủi ro và chi phí liên quan đến sự cố thiết bị. Các thuật toán ML phân tích dữ liệu từ cảm biến như nhiệt độ, độ rung, áp suất và âm thanh để phát hiện các dấu hiệu bất thường báo trước sự cố. Theo nghiên cứu, việc áp dụng ML để dự đoán lỗi có thể giúp các công ty giảm tới 30-50% chi phí bảo trì và giảm tới 70% thời gian ngừng máy đột xuất. Điều này đặc biệt có giá trị trong các ngành công nghiệp như sản xuất, dầu khí, và năng lượng, nơi thời gian ngừng hoạt động đột ngột có thể gây thiệt hại hàng triệu đô la mỗi giờ. Bằng cách phát hiện sớm các vấn đề tiềm ẩn, ML giúp các doanh nghiệp chuyển từ bảo trì theo lịch trình sang bảo trì dự đoán (predictive maintenance), tối ưu hóa hiệu suất thiết bị và nâng cao tuổi thọ máy móc.

| **Chỉ số** | **Mô hình SVC** | **Mô hình KNN** | **Nhận xét** |
| --- | --- | --- | --- |
| **Precision** | 0.869 | 0.943 | KNN có độ chính xác cao hơn SVC, ít dự đoán sai hơn cho các mẫu dương tính. |
| **Recall** | 0.881 | 0.987 | KNN có Recall cao hơn, cho thấy khả năng nhận diện đúng các mẫu dương tính tốt hơn. |
| **F1-score** | 0.875 | 0.964 | KNN đạt F1-score cao hơn, cho thấy cân bằng tốt giữa Precision và Recall. |
| **Accuracy** | 0.874 | 0.963 | Độ chính xác của KNN cao hơn SVC, cho thấy hiệu suất tổng thể tốt hơn trên tập dữ liệu. |
| **MAE** | 0.1256 | 0.0367 | MAE của KNN thấp hơn, nghĩa là sai số trung bình của KNN nhỏ hơn, dự đoán chính xác hơn. |
| **MSE** | 0.1256 | 0.0367 | MSE của KNN cũng thấp hơn, cho thấy sai số bình phương nhỏ hơn so với SVC. |
| **R-squared (R²)** | 0.4977 | 0.8530 | KNN có R² cao hơn nhiều, nghĩa là giải thích tốt hơn phần lớn biến thiên của dữ liệu. |

**Tổng kết:**

* Mô hình **KNN** thể hiện hiệu suất vượt trội hơn so với **SVC** trên tất cả các chỉ số đánh giá, đặc biệt là ở Precision, Recall, F1-score, và R².
* Với độ chính xác cao hơn, sai số thấp hơn, và khả năng giải thích biến thiên dữ liệu tốt hơn, **KNN** là lựa chọn tối ưu cho bài toán này.
* Để có thể tăng độ chính xác của 2 mô hình, ta có thể sử dụng các cách như
  + Tối ưu hóa tham số
  + Lựa chọn các đặc trưng quan trọng