**Introduction**

In this portfolio, I will discuss my work in improving the efficiency of a manufacturing process using machine learning techniques. Specifically, I focused on analysing sensor data from a conveyor belt to identify areas where the process could be optimized. The goal of this project was to use machine learning to analyse sensor data from a conveyor belt in order to identify patterns and trends that could be used to optimize the manufacturing process. This involved collecting data from sensors placed at various points along the conveyor belt and using machine learning algorithms to analyse the data and identify patterns.

Last year I conducted a case study on this topic. The case study revolved around a manufacturing process that involves the use of conveyor belts. One of the major concerns in this process is the reliability of the conveyor motors, which can lead to downtime and loss of production. According to the data collected from bay 7, the potential loss of production due to a motor breakdown can be up to €21,000 per year for that specific bay alone. Extrapolating this data to the rest of the production floor, the total loss for Kostal could rise to €252,000 per year, which is a substantial loss. In order to address this issue, we focused on implementing a vibration sensor on the conveyor belt to detect any anomalies or patterns that could indicate a potential motor failure. By using machine learning techniques to analyse the sensor data, we aimed to identify areas where the process could be optimized to prevent motor breakdowns and minimize downtime.

**Method:**

To collect the sensor data, a Schaeffler OPTIME vibration sensor was installed on a SEHO MWS 2300 wave soldering system that autonomously solders components onto a PCB. The Schaeffler OPTIME vibration sensor is designed for use in industrial environments and is capable of measuring vibration acceleration, velocity, and displacement. The sensor was mounted on the motor for a conveyor belt and collected vibration data during the operation of the wave soldering system.

The raw sensor data was collected and saved in a CSV file, which contained eight columns of numerical data. An example of the data is shown below:

col1 col2 col3 col4 col5 col6 col7 col8 -0.5964 1.6569 0.20359 0.048174 -0.18477 -0.021301 -0.18447 -0.017205 -0.52322 -0.28902 -0.3174 -0.0014495 -0.19779 -0.023694 -0.28712 -0.018203 -0.5532 1.6477 0.41715 0.038655 -0.18563 -0.021564 -0.21215 -0.072039 -0.57723 0.15914 -0.33321 -0.011148 -0.19892 -0.025401 -0.30076 -0.0073728 -0.4928 1.3165 0.41147 0.028096 -0.19165 -0.022879 -0.24147 -0.10859 -0.61773 0.6942 -0.23371 -0.0020918 -0.19707 -0.024991 -0.27857 0.018571 -0.49801 0.41606 0.20767 0.034082 -0.19742 -0.022481 -0.26254 -0.1223

or if viewed on an excel sheet:



Once the data was collected, a variety of pre-processing steps were performed to clean and prepare the data for analysis. This involved removing any invalid or incomplete data, normalizing the data to account for differences in sensor calibration, and converting the data to a suitable format for machine learning analysis. These steps were essential to ensure that our models could accurately learn patterns in the data and make reliable predictions.

The algorithms were trained using the cleaned and prepared data, and their performance was evaluated using various metrics such as accuracy, precision, and recall. The implementation of the algorithms was done in Python using the scikit-learn library. A Jupyter notebook was used for the implementation, and the code is available on Google Colab.

The following machine learning techniques were used to build predictive models for detecting faults in the motor:

1. Linear Regression: This was used as a baseline model for the study, as it is a simple and commonly used regression technique.
2. Decision Tree Regressor: This was used to create a more complex model, as decision trees are known to perform well on classification tasks.
3. Random Forest Regressor: This was used to create an ensemble model that combines multiple decision trees, as it has been shown to perform well on various prediction tasks.

The data was split into a training set (80%) and a test set (20%). The models were trained on the training set and evaluated on the test set.

**Results:**

**Performance of the Machine Learning Models**

The performance of the linear regression, decision tree regressor, and random forest regressor models were evaluated using mean squared error (MSE), mean absolute error (MAE), and R-squared (R2) score metrics. The results are summarized in Table 1.

| **Model** | **MSE** | **MAE** | **R2 Score** |
| --- | --- | --- | --- |
| Linear Regression | 0.01 | 0.05 | 0.68 |
| Decision Tree Regressor | 0.01 | 0.05 | 0.62 |
| Random Forest Regressor | 0.00 | 0.04 | 0.81 |

Table 1: Performance of the Machine Learning Models

The random forest regressor model performed the best with an R2 score of 0.81 and a lower MSE and MAE values. The linear regression and decision tree regressor models also had good performance with R2 scores of 0.68 and 0.62, respectively.

**Distribution of Target Variable**

Figure 1 shows the histogram of the target variable. It can be seen that the distribution is skewed, with most of the values concentrated around the mean. This indicates that the motor is operating normally for most of the time, but there are occasional periods of increased vibration levels due to faults or imbalances.

Chart, histogram

Description automatically generated

Figure 1: Histogram of the Target Variable

**Feature Importance**

The feature importance plot for the random forest regressor model is shown in Figure 2. It can be seen that the dominant feature is **vibration\_y**, which indicates that the vertical vibration is a strong indicator of faults in the motor. The other features, **vibration\_x** and **vibration\_z**, also contribute to the prediction, but to a lesser extent.

Chart, scatter chart

Description automatically generated

Figure 2: Feature Importance Plot for Random Forest Regressor Model

**Residual Plot**

The residual plot for the random forest regressor model is shown in Figure 3. It can be seen that the residuals are randomly scattered around zero, indicating that the model has captured the underlying patterns in the data well.

Chart

Description automatically generated

Figure 3: Residual Plot for Random Forest Regressor Model

**Discussion (including Conclusion):**

The results of the study suggest that the random forest regressor performed the best in detecting faults in the conveyor belt motor. The R-squared score of 0.81 indicates that the model is able to explain a large amount of the variance in the data. However, the study has some limitations, including the small size of the dataset. Additionally, the study only considered three machine learning techniques, and other techniques may be more suitable for this task. Further research is needed to validate the findings of this study and to explore other machine learning techniques that may be effective for detecting faults in conveyor belt motors.

In conclusion, this study demonstrates the potential of using machine learning techniques for detecting faults in conveyor belt motors. The random forest regressor showed promising results and can be used as a starting point for future research.