**Identification of railway track events**

Switches are an essential, safety-critical part of railway infrastructure. Compared to open tracks, their complex geometry results in increased dynamic loading on the track superstructure caused by passing trains, leading to higher maintenance costs. To improve efficiency, condition monitoring methods specific to railway switches are necessary. A common approach to monitoring the track superstructure is to measure the acceleration caused by vehicle-track interaction. Local interruptions in the wheel-rail contact, caused by factors such as defects or track discontinuities, appear in the data as transient impact events.

In this assignment, these transient events are investigated using an ***experimental setup of a railway switch equipped with track-side acceleration sensors.*** The analysis uses frequency and waveform data to understand the origins of these impact events and their implications for monitoring local track discontinuities and defects with wayside acceleration sensors in practice.

**Dataset:**

The data comprises three files, each containing extracted features from 1D acceleration sensor signals. These features include:

**Statistical Features:** mean, std, max, min, range, skewness, kurtosis, rms, crest\_factor, variance, zero\_crossings.

**Frequency Features:** dominant\_freq, spectral\_energy, spectral\_centroid, spectral\_bandwidth, spectral\_flatness.

Each file includes a column labelled event, which serves as the ground truth for the presence of transient events.

**Tools Required:**

Python programming language.

Libraries such as numpy, matplotlib, and other relevant tools for data preprocessing, visualization, and machine learning.

**Task to pass with Grade 3:**

1. **Data preprocessing:**
   * Load the data from all three files.
   * Combine the three datasets into a single unified dataset.
   * Remove the columns start\_time, axle, cluster, tsne\_1, and tsne\_2 from the dataset.
   * Replace all normal events with 0 and all other events with 1.
2. **Data transformation:**

Normalize the dataset.

**Task to pass with Grade 4:**

1. **Dataset splitting:**

Split the data into training and testing sets in an 80/20 ratio.

1. **Cross-Validation:**  
   Perform k-fold cross-validation (e.g., 5-fold) on the training set to evaluate model stability.
2. **Comparison task:** Compare between the 80/20 train-test split and k-fold cross-validation using SVM (Support Vector Machine). Train an SVM model using both methods and evaluate its performance. Discuss the differences in accuracy, consistency of results, and generalization ability.

**Task to pass with Grade 5:**

Implement feature selection algorithms to identify and retain the most relevant features, improving model performance by reducing noise and dimensionality.

Steps to complete the task:

1. **Research and understand various feature selection techniques, such as:**
   * Filter methods (e.g., Pearson correlation, chi-square test).
   * Wrapper methods (e.g., recursive feature elimination).
   * Embedded methods (e.g., LASSO, feature importance in tree-based models).
2. **Implement at least four feature selection algorithms in this project, applying them to the dataset.**

**The report**

You have to submit a report on this assignment. The report should contain:

1. Heading
2. Name of the participants
3. Problem description
4. Method of solution
   1. Steps to implement event detection.
5. Your observations and reflections

The assignment will be considered complete, when the student has demonstrated the working code to the teacher, uploaded the code to GitHub repository, and submitted the report on canvas.