MobGap-Plus:
An HPC-Powered ML Pipeline for Digital Mobility: Bridging Python & R-Shiny

Lead Supervisor: V. Lanfranchi, School of Computer Science, The University of Sheffield.
Email Address: v.lanfranchi@sheffield.ac.uk
Named dRTP/Co-Supervisor: Robert Chisholm

Background and aims:
Digital Mobility Outcomes (DMOs), such as gait speed and cadence, are vital biomarkers for monitoring disease progression and treatment efficacy. While the mobgap library provides a validated framework for extracting these outcomes from Inertial Measurement Unit data, a significant technical gap remains: translating signal processing into a user-friendly clinical decision-support tool. The goal of MobGap-Plus is to develop an integrated web pipeline that bridges this gap:

1. Ingests and converts raw sensor formats into mobgap-compatible structures to processes raw signals into DMOs
2. Provides interactive tables and graphical visualisations for clinical review (building on an existing visualisation platform developed in Ruby on Rails).
3. Utilizes the extracted DMOs as input features for Machine Learning models (e.g., classifying gait quality or predicting clinical scores).
4. Connects the Python processing backend to an existing Shiny R dashboard via API.

This project involves a workflow for which High-Performance Computing (HPC) is essential. It will leverage HPC to demonstrate reproducible workflows, using containerisation (e.g., Apptainer/Docker) to ensure the pipeline is portable and scalable for the following tasks:

- Feature engineering to generate DMOs from 100Hz+ IMU signals, and interoperability is computationally expensive.
- For longitudinal studies involving hundreds of participants, the volume of data reaches Big Data proportions. HPC allows for parallelisation, to improve efficiency.
- For hyperparameter tuning and model validation, GPU-accelerated clusters (like N8 Bede) will enable rapid training and deployment of ML classification models
- HPC will provide a robust backend environment for the "Python-R bridge" with the required memory and throughput to host a seamless and low-latency service.

Candidate profile:
The ideal candidate should have interest in Digital Health and Research Software Engineering. Essential skills: Proficiency in Python and libraries such as Pandas, NumPy, and Scikit-learn; Understanding of how to build and query API; Understanding of time-series data; Competency with Git/GitHub for collaborative code management and documentation.

Output:
In the 8 week internship and working closely with researchers from the Pervasive Computing and RSE groups, the intern will:

1. Produce a functional Web Prototype that ingests raw IMU data to output validated DMOs
2. Embed a trained model that uses DMOs as features to indicate disease progression
3. Create a documented API and brief technical summary of the speedups achieved by parallelising the mobgap pipeline on the N8 cluster