EPQ Project Proposal

Problem:

The number of elderly people is increasing and will continue to increase, at least for the near future. As a result, a major concern is their health and safety, especially when they are living alone. Falling is an extremely dangerous matter and is a common occurrence.

Solution:

Fall detection, if a sensor detects someone has fallen over, a relative or the emergency services will be notified.

Literature Review:

- Possible methods of fall detection:
    - Wearable IMU devices (Inertial Movement Unit)
        - Often has gyroscope and/or accelerometer
    - Camera
        - Measures motion of target
        - Attached to wall
        - A device worn on body
    - Gyroscope and accelerometer data
        - Measures rotation of
        - From Phone
        - From Watch
- Possible methods to analyse data:
    - Suitable languages:
        - Python
        - JavaScript
        - SQL
        - C family
    - Mathematical and Statistical Methods:
        - Regression Analysis: relationship between dependent and independent variable
        - Discriminant Analysis: what makes two groups different
        - Dispersion Analysis: dispersion of data, how it is spread
    - Visualize data using graphs
    - Machine Learning and Artificial Intelligence: Classification
        - Linear and Logistic Regression (Supervised)
        - Classification and Decision Trees (Supervised)
        - Naive Bayes (Supervised)
        - Neural Network (Reinforcement)
        - Random Forest (Supervised)

Proposed Solution:

The proposed solution is to use the gyroscope and accelerometer data of a phone to measure movement and implement fall detection. The phone was chosen because it is the most common device, and also, is the cheapest – (assuming they already have a phone) people do not need to invest in any expensive watches or other wearable devices and can download an application for free or with significantly lower expense. Also, the phone was the most accessible device when doing this project.

Objective:

- Build a mobile application that can access the gyroscope and accelerometer data of phone:
    - Gyroscope and accelerometer in 3 dimensions (X, Y, Z)
    - Numerical to sufficient degree of accuracy (3+ significant figures)
    - Store collected data in CSV file
- Collect sufficient amount of data:
    - Different ways of falling
    - Repeats, resulting in consistent data
- Analyse the data and produce an algorithm for fall detection:
- Implement and test the algorithm:
    - High probability of true positives and true negatives using new fall data
    - Low probability of false positives and false negatives using new fall data

Resources:

- There is no financial budget, but spending should be kept to a minimum. The time limit is a year, the length of EPQ course. Other assets include:

Phone	Gyroscope Accelerometer
Computer	IDE with Python and C#
Secondary data sources	Kaggle Published papers (Google Scholar)

Tasks:

- Task 1: Build application to get gyroscope and accelerometer:
    - 1.1: Learn how to create application on Visual Studios using Xamarin
    - 1.2: Access gyroscope and accelerometer data of phone
    - 1.3: Save any data collected in an appropriate format
        - 1.3.1: Must be accessible – external storage
- Task 2: Collect data *(Needs to be done in an ethical and safe manner)*:
    - 2.1: Ask for volunteers and participants
    - 2.2: Use phone running application to get gyroscope and accelerometer data
    - 2.3: Collect walking motion as control variable
    - 2.4: Participants fall over to get required data

        - 2.5.1: Fall over in different ways – forwards, backwards, left, right
    - 2.5.2: Tests repeated multiple times (current estimate: 3)
- Task 3: Process data:
    - 3.1: Learn how to use Python's Data Science libraries
    - 3.2: Import sensor data from phone to computer
    - 3.3: Experiment with and analyse data
        - 3.3.1: Look for patterns
        - 3.3.2: Identify when someone has fallen over based on value and shape of data
        - 3.3.3: Create ML algorithm to identify when someone has fallen based on 'new' data
- Task 4: Evaluate performance of algorithm and effectiveness of data:
    - 4.1: Implement fall detection algorithm
        - 4.1.1: Record results
        - 4.1.2: Probability of false positives and false negatives
    - 4.2: Optimize ML algorithm if needed
    - 4.3: Check against similar papers
    - 4.4: Use secondary sources of data for testing

Risk Assessment:

Risk	Likelihood	Mitigation
Unable to build application to collect primary data	Low	Use application available on App Store or other sources.
Unable to collect accurate data	Medium Low	Use more secondary data.
Unable to collect a sufficient amount of data	Low	Use more secondary data.
Data collection experiment considered unsafe for participant	Medium	Ensure participants are safe. For example, make them fall onto soft mat, teach them the correct falling technique, inform them of the safety risks.
Machine Learning algorithm cannot make any useful observations	Medium High	Use mathematical observations and interpretations. Alternatively, try a different ML algorithm.
No useful observations whatsoever from primary and/or secondary data	Medium High	Try reading similar papers published online to see what observations they made.