Stress Detection and Alleviation through Sensors

CIS/CEN4914 – Fall 2018 – Presentation #2

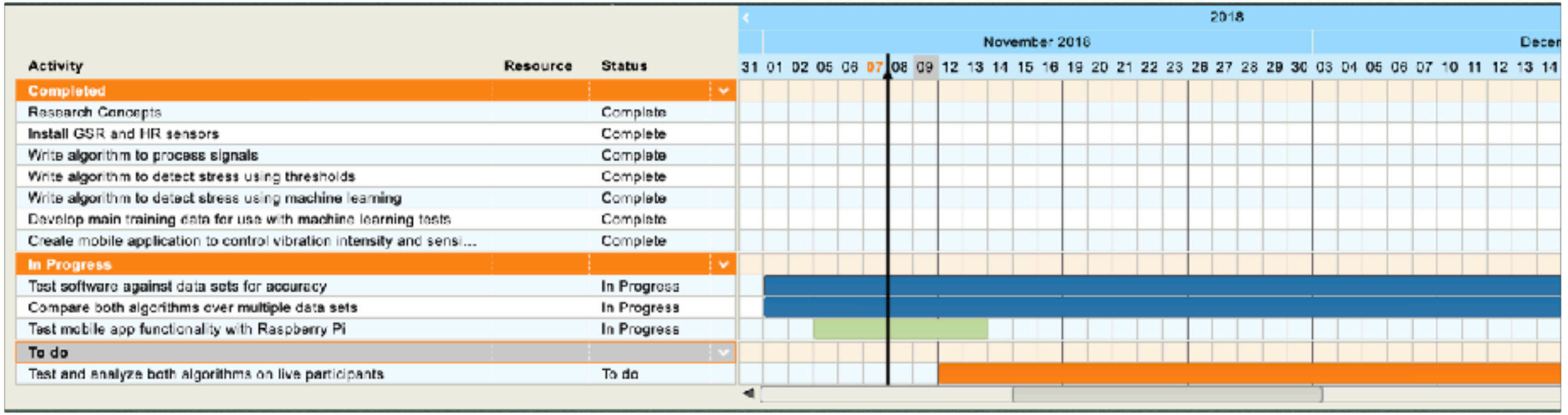
SYLVIA LUJO

OLIVIA LEUNG

Project Overview

- Focus
  - Manage a user's stress autonomously through use of a device
  - Develop software that can accurately detect stress
- Motivation
  - Stress is increasingly prevalent in today's society
  - Current stress management methods not always effective
- Technical Approach
  - Develop and test algorithms to detect stress through sensors
  - Build device hardware necessary for algorithms to be effective

GANTT Chart



Activity	Resource	Status
Completed		
Research Concepts		Complete
Install GSR and HR sensors		Complete
Write algorithm to process signals		Complete
Write algorithm to detect stress using thresholds		Complete
Write algorithm to detect stress using machine learning		Complete
Develop main training data for use with machine learning tests		Complete
Create mobile application to control vibration intensity and sensi...		Complete
In Progress		
Test software against data sets for accuracy		In Progress
Compare both algorithms over multiple data sets		In Progress
Test mobile app functionality with Raspberry Pi		In Progress
To do		
Test and analyze both algorithms on live participants		To do

Literature

- Salai, Mario, et al. “Stress Detection Using Low Cost Heart Rate Sensors”, *Journal of Healthcare Engineering* (2016).
- Perala C.H., Sterling B.S. “Galvanic Skin Response as a Measure of Soldier Stress”, *Army Research Laboratory ARL-TR-4114* (2007).
- Agarwal, A., et al. “Acupressure for Prevention of Pre-Operative Anxiety: a Prospective, Randomised, Placebo Controlled Study.” *Anaesthesia*, Blackwell Publishing Ltd (2005).
- Healey JA, Picard RW. "Detecting stress during real-world driving tasks using physiological sensors". *IEEE Transactions in Intelligent Transportation Systems* 6(2):156-166 (2005).

Algorithm Flowcharts

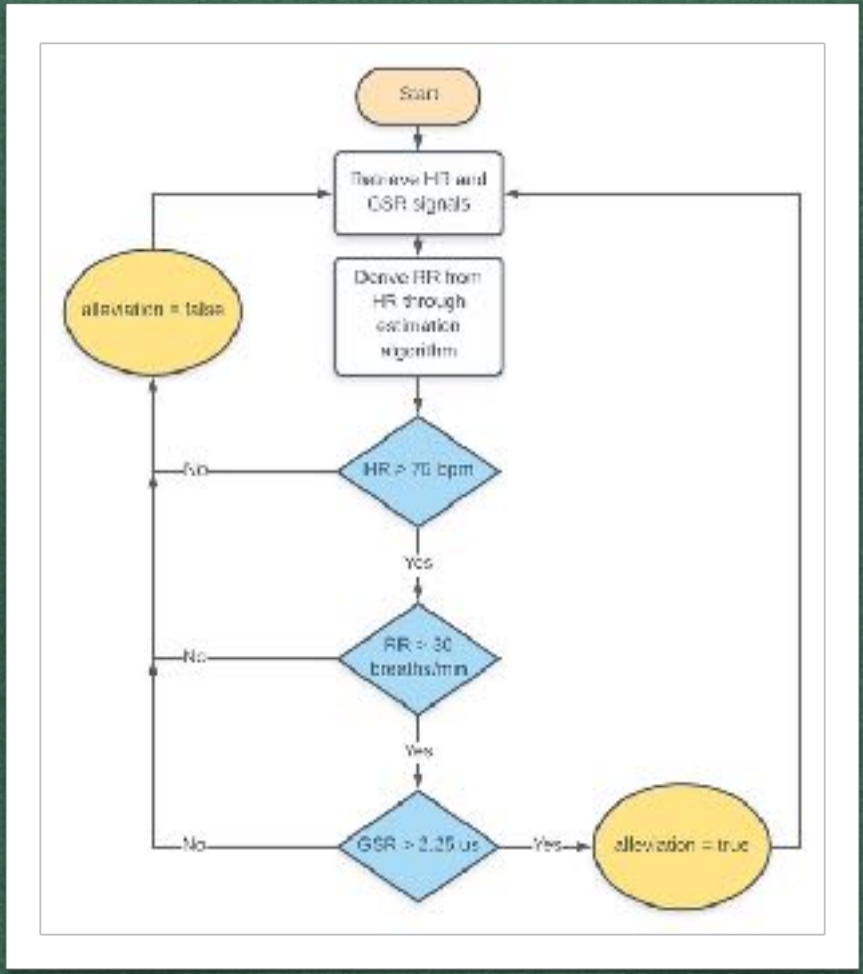


Figure 1. Model of algorithm process based on user signals passing certain thresholds indicative of stress.

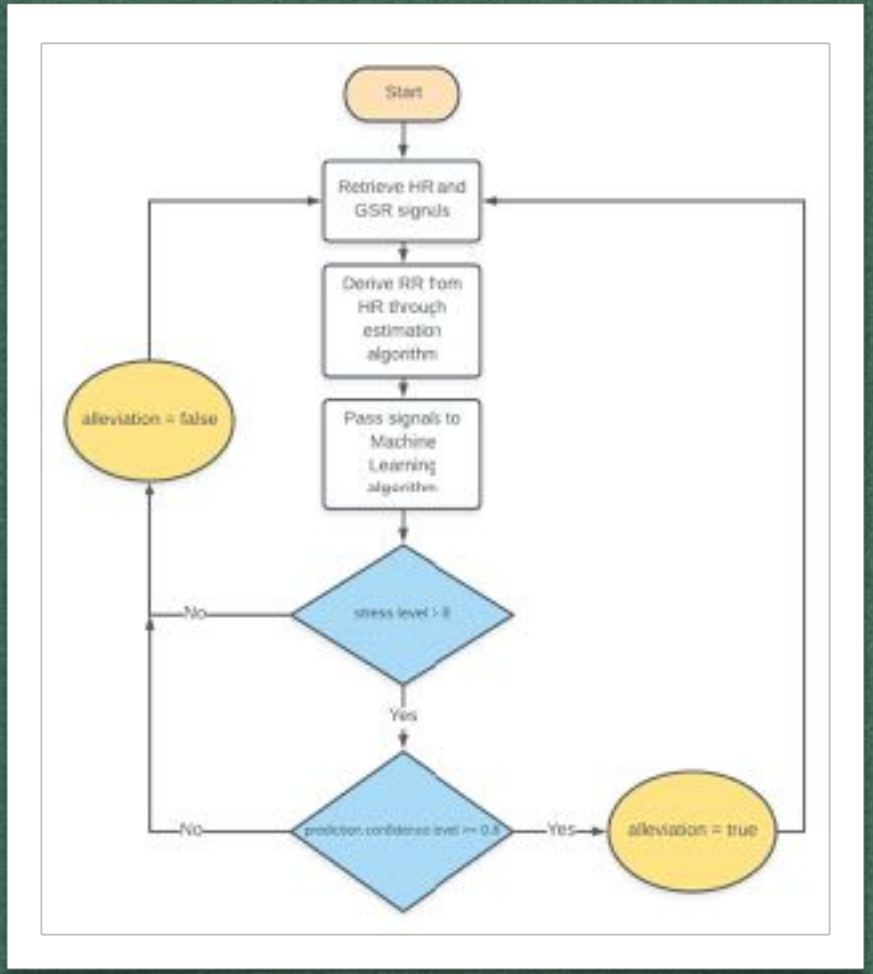


Figure 2. Model of algorithm process based on machine learning.

Comparison: Threshold vs. Machine Learning

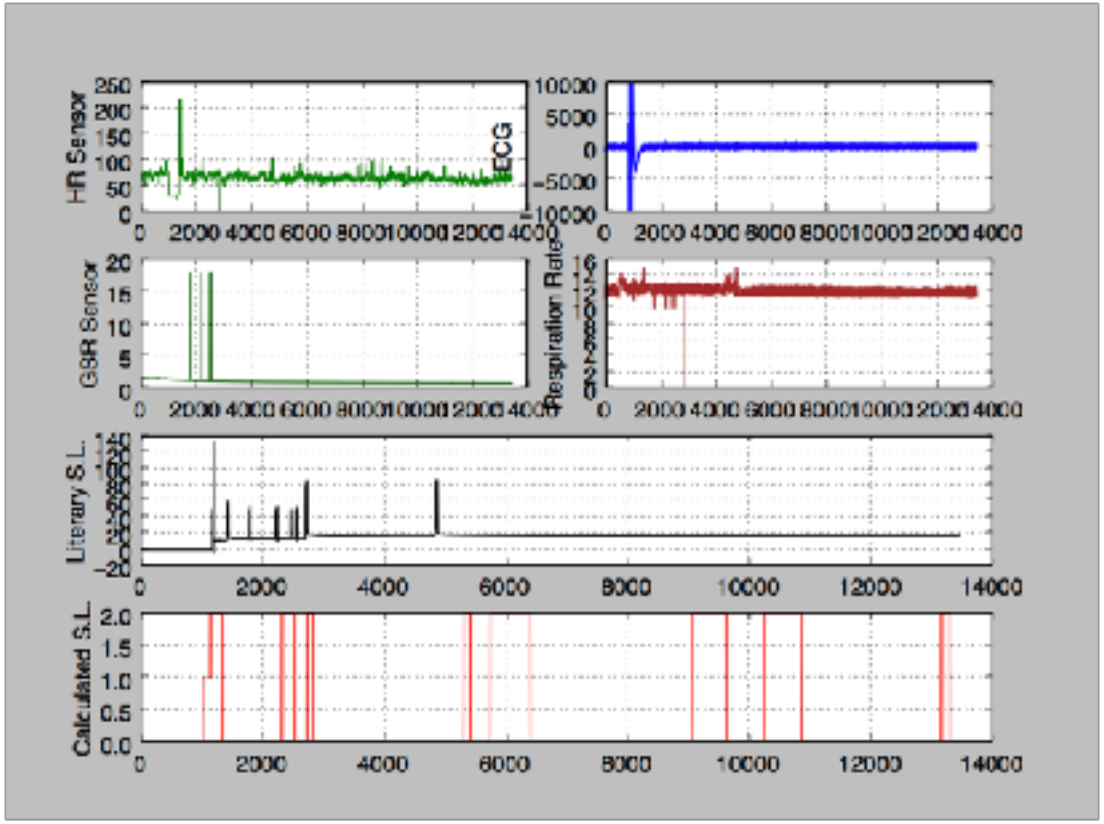


Figure 3. Run of threshold algorithm on a data set with a pre-defined stress level for each data point for comparison purposes.

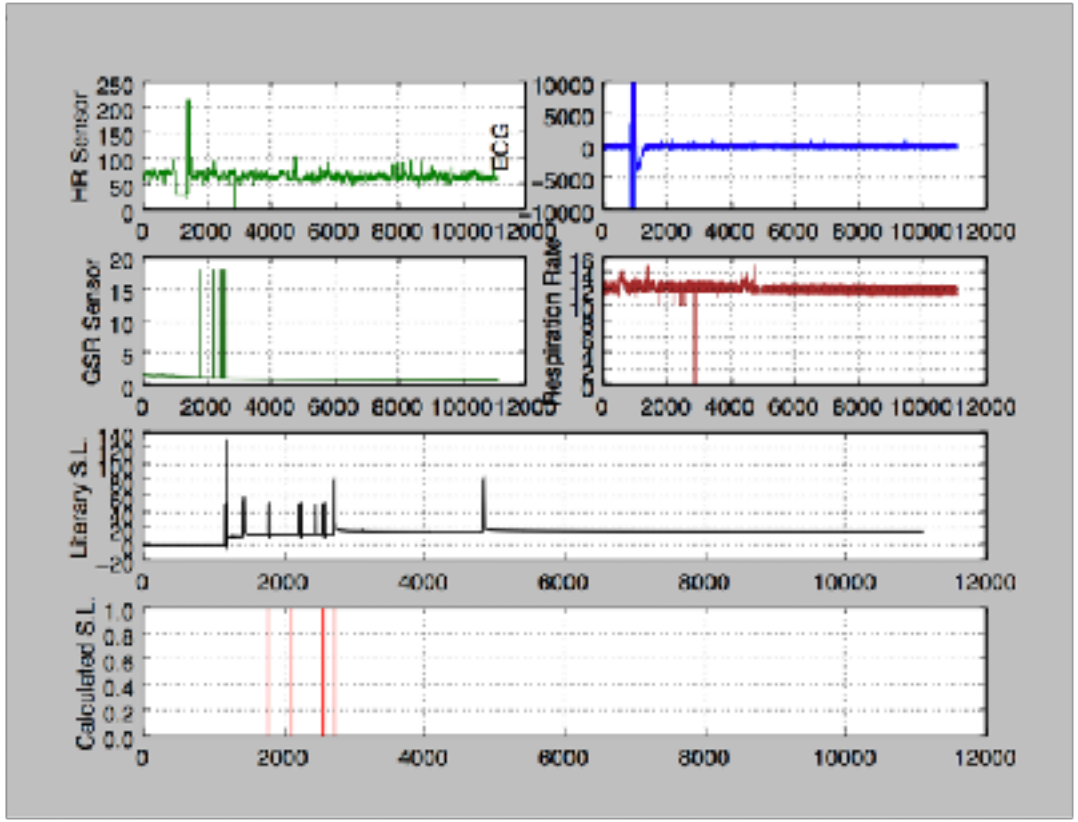


Figure 4. Run of machine learning algorithm on a data set with a pre-defined stress level for each data point for comparison purposes.

Comparison: Machine Learning Variations

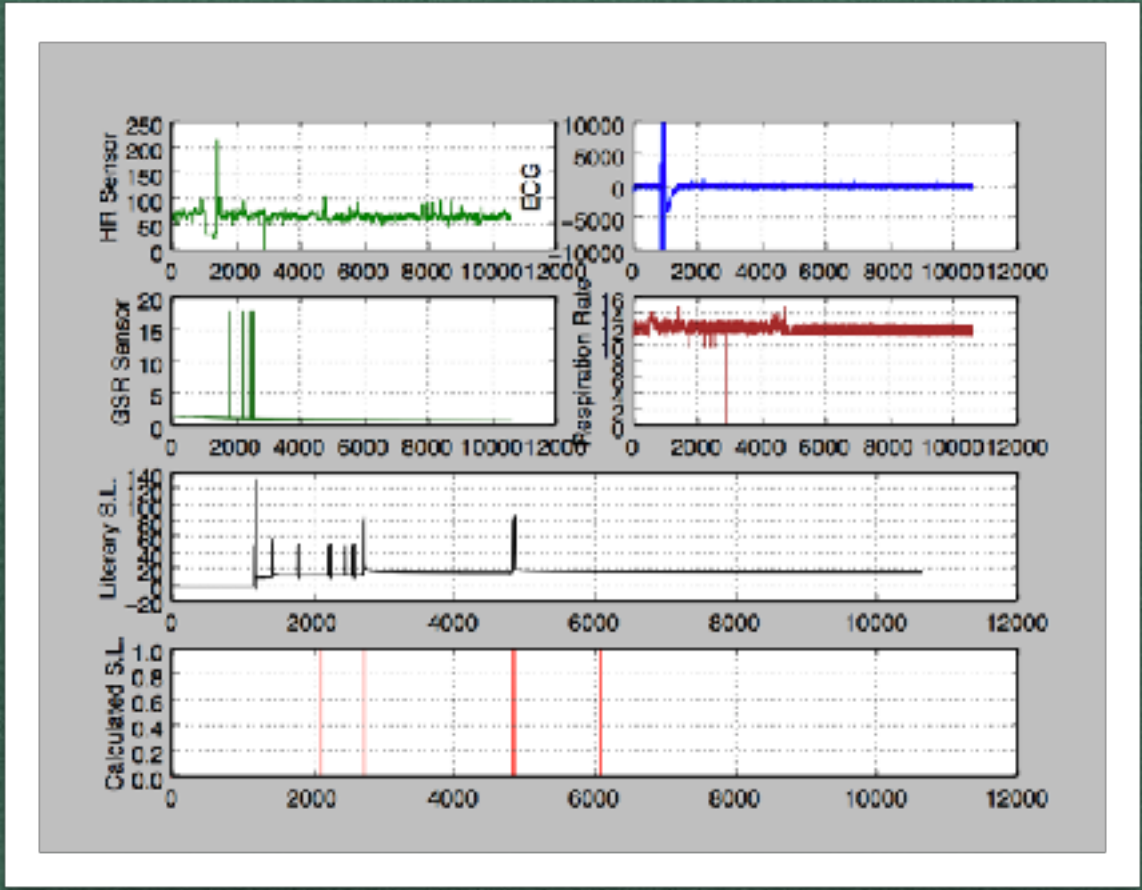


Figure 5. Run of machine learning algorithm against data set. Algorithm trained to recognize stress on a scale of 1-5.

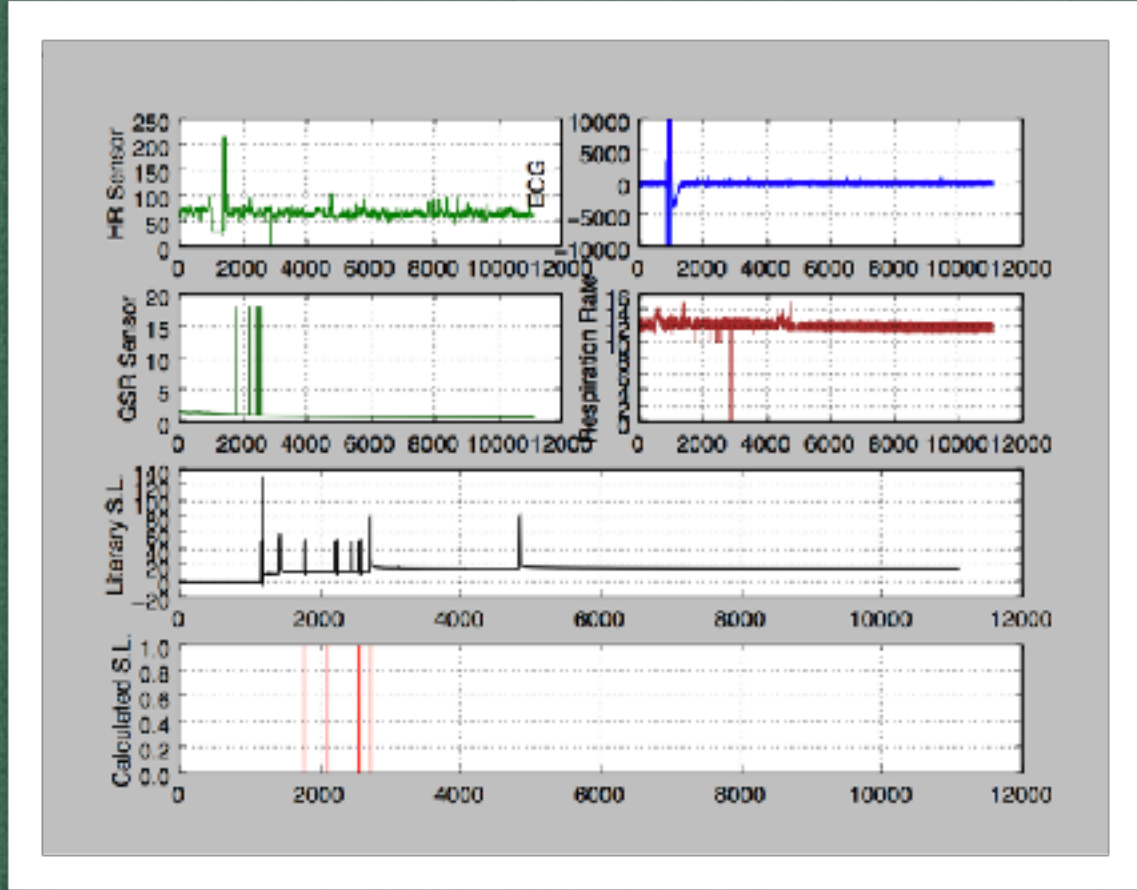


Figure 6. Run of machine learning algorithm against data set. Algorithm trained to recognize stress on a scale of 1-10.

Companion Mobile Application

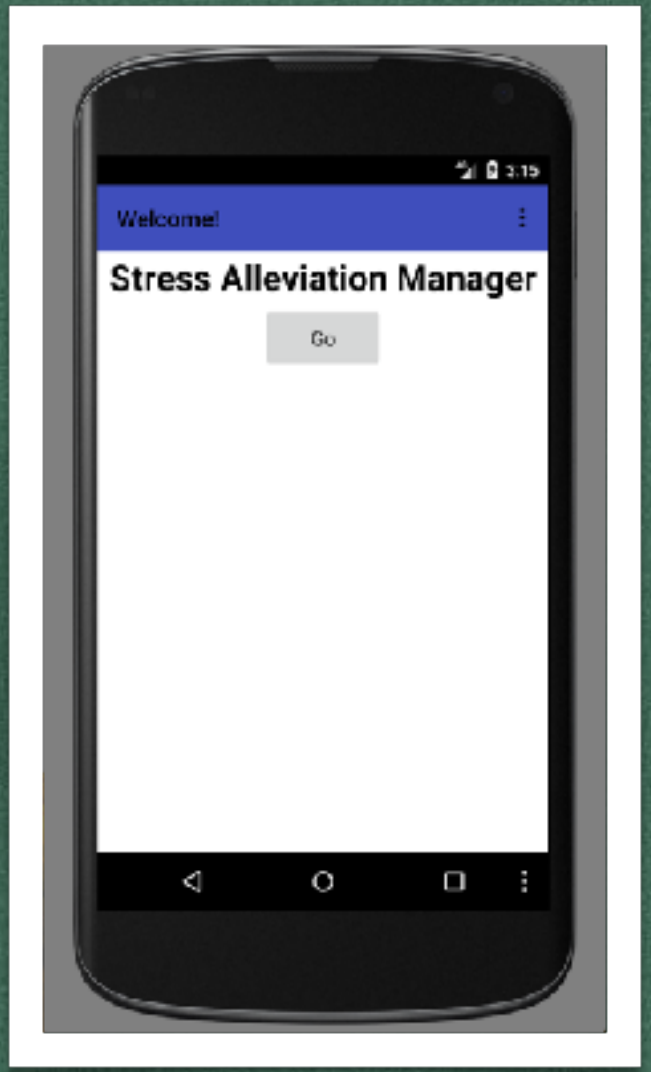


Figure 7. Companion application start screen.

Figure 8. Companion application bluetooth connections screen.

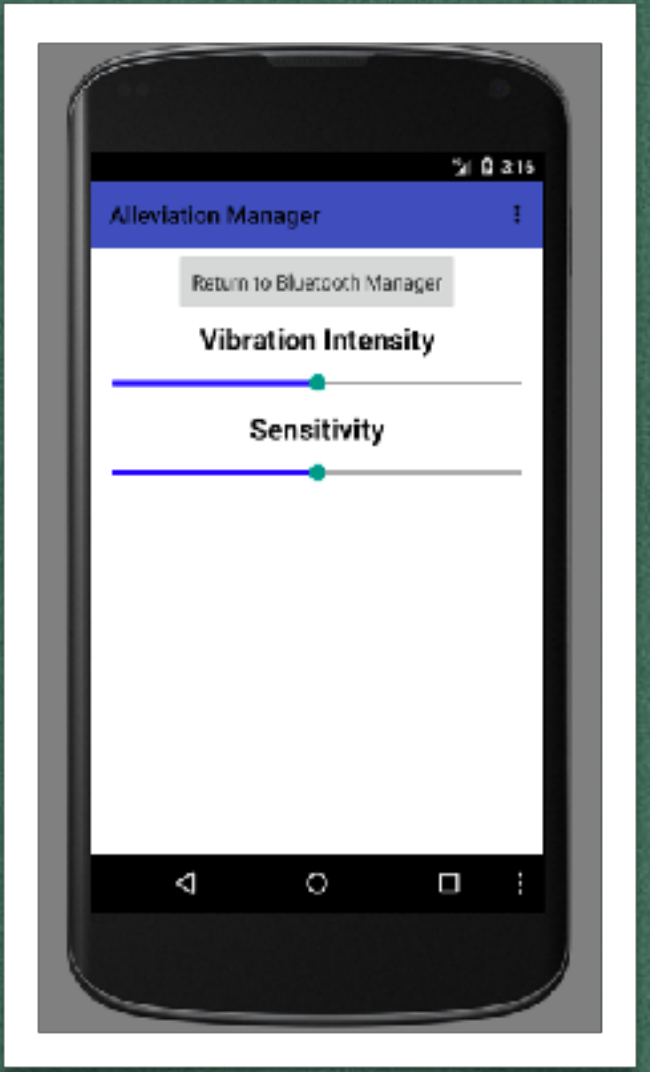


Figure 8. Companion application screen for alleviation settings.

Work to Be Performed

- Continue testing algorithms
  - Determine accuracy and sensitivity of each algorithm
- Determine training set that allows for best algorithm performance
  - Train algorithm to accurately detect stress with different ranges of sensitivity
- Test mobile application alongside Raspberry Pi
  - Test connectivity and functionality of app in correlation to the device