**Brett Wyton – 45423229 – COMP255 Assignment 1 – Project Movement**

**Introduction:**

The purpose of this project is to extract specific human activities from a series of given datasets. The data set is recorded by a range of accelerometers and gyroscopes located on the body. They are located on the wrist, chest, hip and ankle. The accelerometer measures the linear movement of the x, y and z co-ordinates of each unit and the gyroscope measures the rotation of each of the devices.

**SCRUM Sprint 1:**

The first sprint will be focusing on the loading of data sets and the visualisation of the data. Signal data will be filtered in the attempt to reduce the micro movements that will get picked up by the sensors.

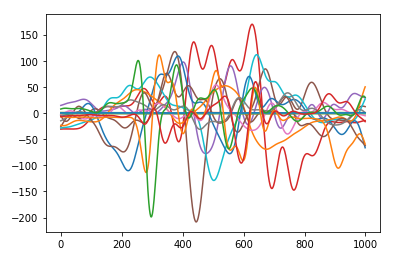
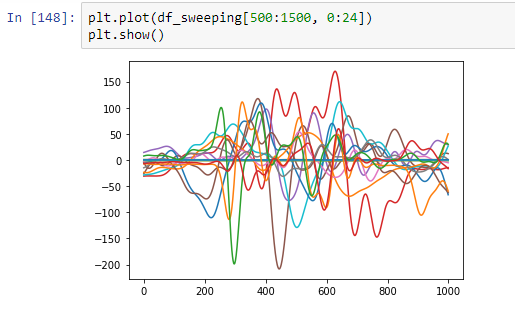
The three most useful technologies used for achieving this were the; pandas, matplotlib and SciPy libraries. All three working in tandem allowed for the reading of the data files, plotting the data itself and removing noise signal from the plotted data respectively. The activity that I chose to compare was the “sweeping activity”. In the case of the first data set, I recorded the unfiltered signal from the wrist accelerometers and then compared it to the filtered (using a lowpass filter) wrist accelerometer. I repeated this for all the accelerometers (the first four plotted data comparisons) and then used the same method for the gyroscopes. After the individual comparisons between devices were made, in the interest of a complete picture of the noise cancelling process, I compared filtered and non-filtered data plots from all the recording from both the first and the last data sets.



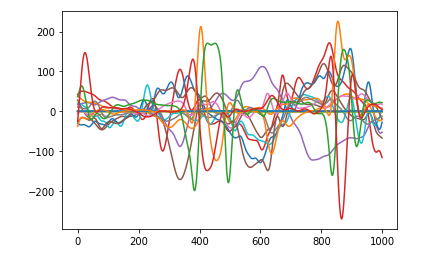
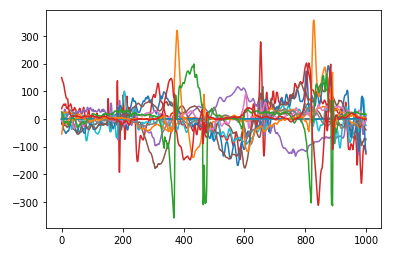
(Wrist accelerometer data plots)



(Ankle gyroscope data plots)

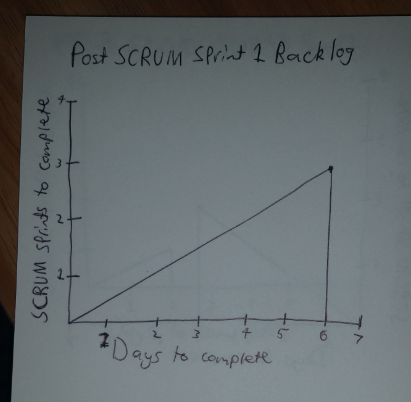


(Unfiltered sweeping data vs filtered sweeping data for dataset 1)



(Unfiltered sweeping data vs filtered sweeping data for dataset 19)

**SCRUM Burndown Chart:**

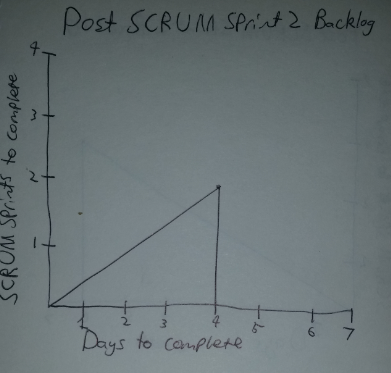


**SCRUM Sprint 2:**

The second sprint will focus on feature engineering. The now cleaned sensor data can have features extracted. The KNN machine learning model can also be applied here to help recognise human activity patterns.

The most useful technology for this section was the use of the Numpy library. This library is useful for making multi-dimensional arrays and matrices which are incredibly useful for holding data in an easy to read and understand way.

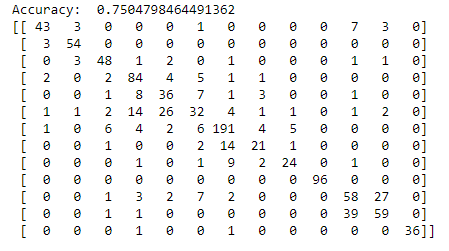
**SCRUM Burndown Chart:**



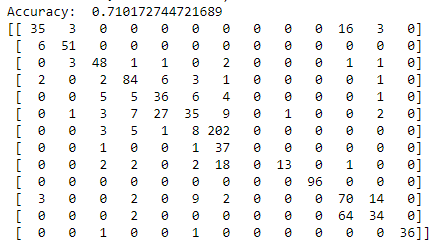
**SCRUM Sprint 3:**

In this SCRUM Sprint the models need to be tested for their accuracy. Both a KNN and an SVM machine learning algorithm will be utilised to test the accuracy of the test data.

In this SCRUM Sprint, the KNN and SVM models were used to test the training data. Both models had a confusion matrix created to test their accuracy. The KNN confusion matrix was created with an accuracy of ~75% whereas the SVM model only had an accuracy of ~71%. Bicycling on an ergometer of 50 W and 100W were the two most confused activities. This is most likely due to how similar some of the results are in that it is a similar activity regarding movement.

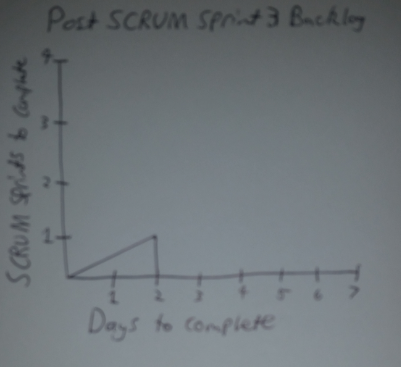


(KNN confusion matrix).



(SVM confusion matrix).

**SCRUM Burndown Chart:**



**SCRUM Sprint 4:**

This SCRUM Sprint will focus on code refactoring. Much of the code that was given was longwinded and difficult to trace. This sprint is designed to held it become more legible to those who may read the source code after it’s been written.

The method for completing this SCRUM Sprint is self-explanatory. Through directly reading through the code and understanding which different parts achieved different outcomes, adding comments to help increase clarity on what how the program works was simple enough.

**SCRUM Sprint Backlog:** Completed.

**Evaluation:**

The accuracy of the results is much lower than desired for this project. If the machine learning model can only accurately decipher 70 – 75% of the time that the activity is what it says it is, then that is not a successful experiment. The machine learning model needs to become more accurate by using more testing data to help increase these number much closer to around the 90 – 95% accuracy mark.

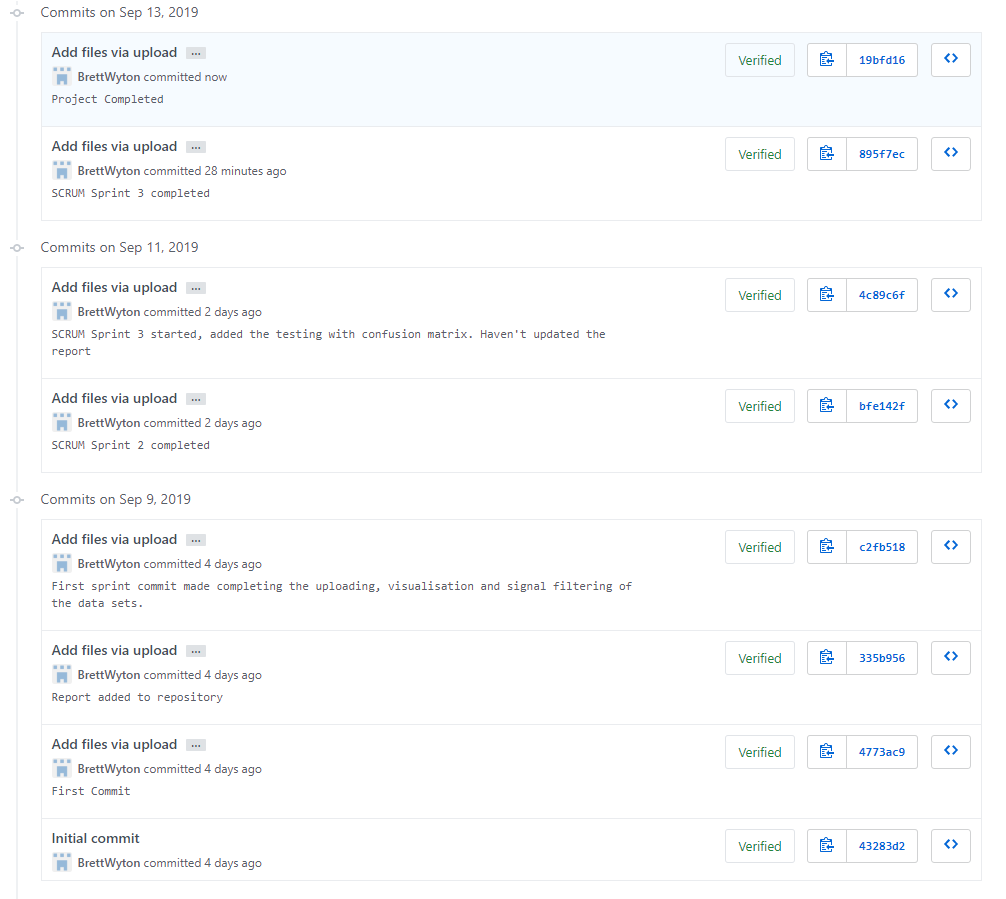
**Challenges:**

This assignment was incredibly difficult to understand what was being asked. As majority of the code was given, it did not feel like an assignment as it was not required to create the functions ourselves. The assignment document was also difficult to understand as the marking criteria did not make it clear where the most marks could be gained. The feature engineering SCRUM Sprint was the most difficult to implement. Thee was a lot of information in the source code to unpack and that made it incredibly difficult to alter to increase the amount of sensors that could be recorded for the test data (a task that I was unable to succeed with in this assignment). The version control system (GitHub) was also a difficult one to implement as for the majority of my time programming I have used Bitbucket and SourceTree for my version control and as such, adapting to the different system was difficult to the point where it was much easier and less of an issue to simply upload all the files at different stages of the development, than attempt to create a repository and push the files from the command line.

**Conclusion:**

The results from the machine learning algorithms were ultimately unsuccessful in my eyes. Being only 70 – 75% accurate in it’s recognition of the different activities is too low and must be further increased for the project to be considered a success.

**Version Control Commit History:**



**References:**

<https://github.com/>

<https://jupyter.org/install>