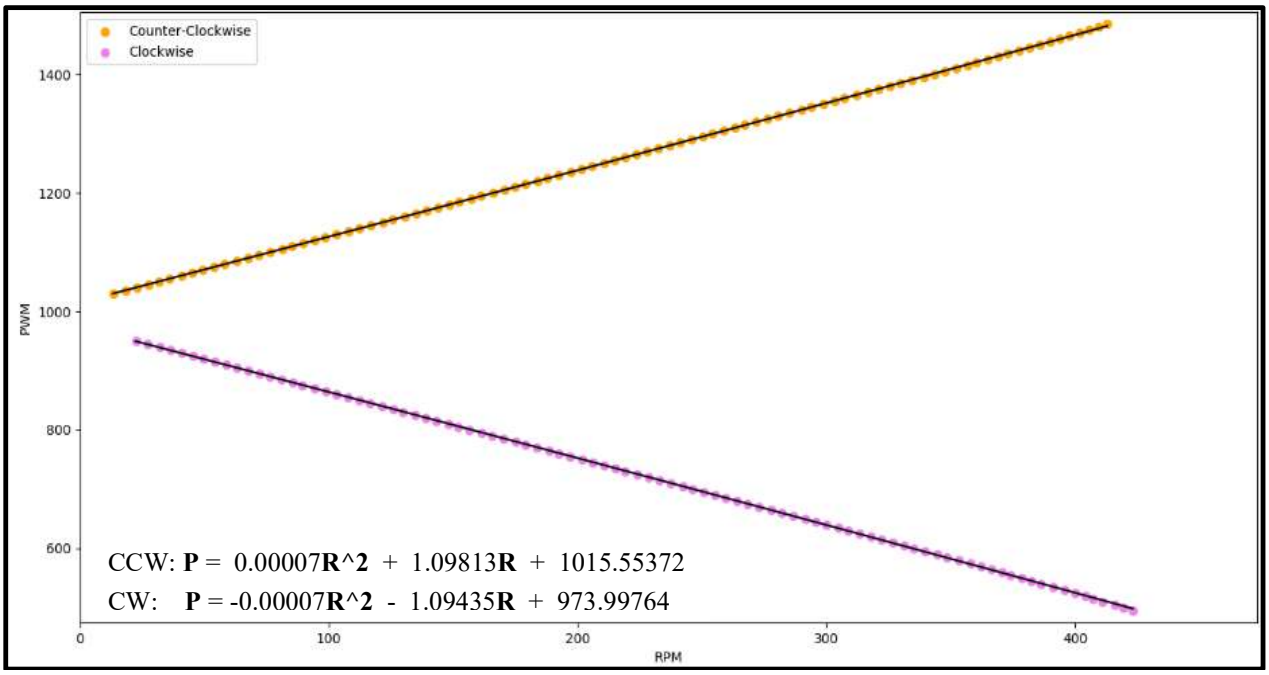
Figure 10



4.4 USER INTERACTION

Once the electronics are powered, the Raspberry Pi will immediately boot up and start the sensor analysis program. An audio alert informs the user when this occurs. The robot will be in a standby state until the user interacts with the handheld keyboard. Pressing <ENTER> will allow robot sensing and repositioning to begin, sounding another audio cue. Currently there are four audio signals in total that the user will need to be made aware of. A basic understanding of Python keyboard interrupts for stopping the running code, as well as knowledge of Linux commands for properly restarting the terminated program and for shutting off the Pi when not in use is also needed.

4.5 DRAWBACKS

The previous section brings attention to the fact that the user is expected to have some software knowledge in order to use the machine properly. The user also has no flexibility in terms of different modes or default settings to choose from, such as how far or close they can be to the robot before it moves. This could be an issue with individual height differences and sparring strategies.

4.6 FUTURE IMPROVEMENTS

Adding multithreading to our project so that modules can run synchronously should improve runtime and allow the robot to respond quicker to its surroundings. This is especially needed if multiple sensors are being used for distance tracking. Learning and implementing OpenCV will improve accuracy of user tracking if a camera is incorporated into the SAMAR system. The user experience drawbacks discussed in the last section will be addressed, though this may not be a priority until a final product is ready to be released.