# Modelling Brain Interfaces

By Alanna Manfredini

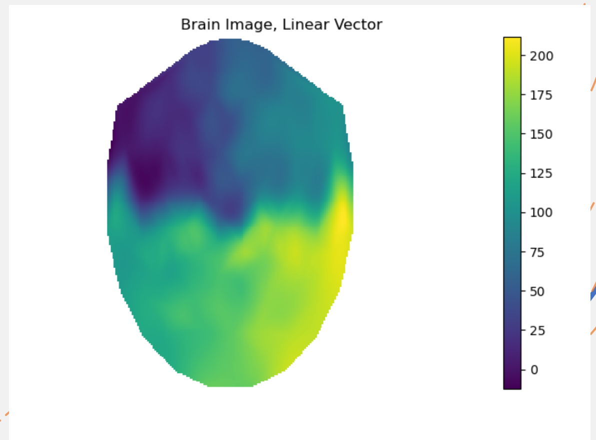
# Summary of Theory and Impact - BCI

The first insight into how the brain controls movement was made in the 19th century and showed that the cerebral cortex sent electrical signals from the brain to the spinal column. In fact the signals which enter the cerebral cortex appear there by way of the cerebellum. One of the oldest evolutionary sections of the brain (Evarts). The cerebellum is located in very close proximity to the brain stem at the back of the brain (Kazilek). In this experiment, values are plotted on the brain in order from the front left hand side of the brain to the back right side of the brain. It can be expected that the values from the cerebellum will be further back in the brain and thus correspond to the higher numbered electrodes. Using a model with the left hand side as class 1 and the right hand side as class 2, the electrode weights will be negative for values that are determined to be in class 1.

An incredible feature of the brain is that imagined movement and actual movement activate the same parts of the brain without delay. In fact, if one imagines performing an exercise rather than actually performing the exercise their muscle mass increases by 22% rather than 30%. This highlights that although imagined movement does not activate the brain and body to quite the same degree as actual movement, it is highly correlated (Decety). This allows us in an experiment to compare results from cerebellum activation from imagined vs actual movement.

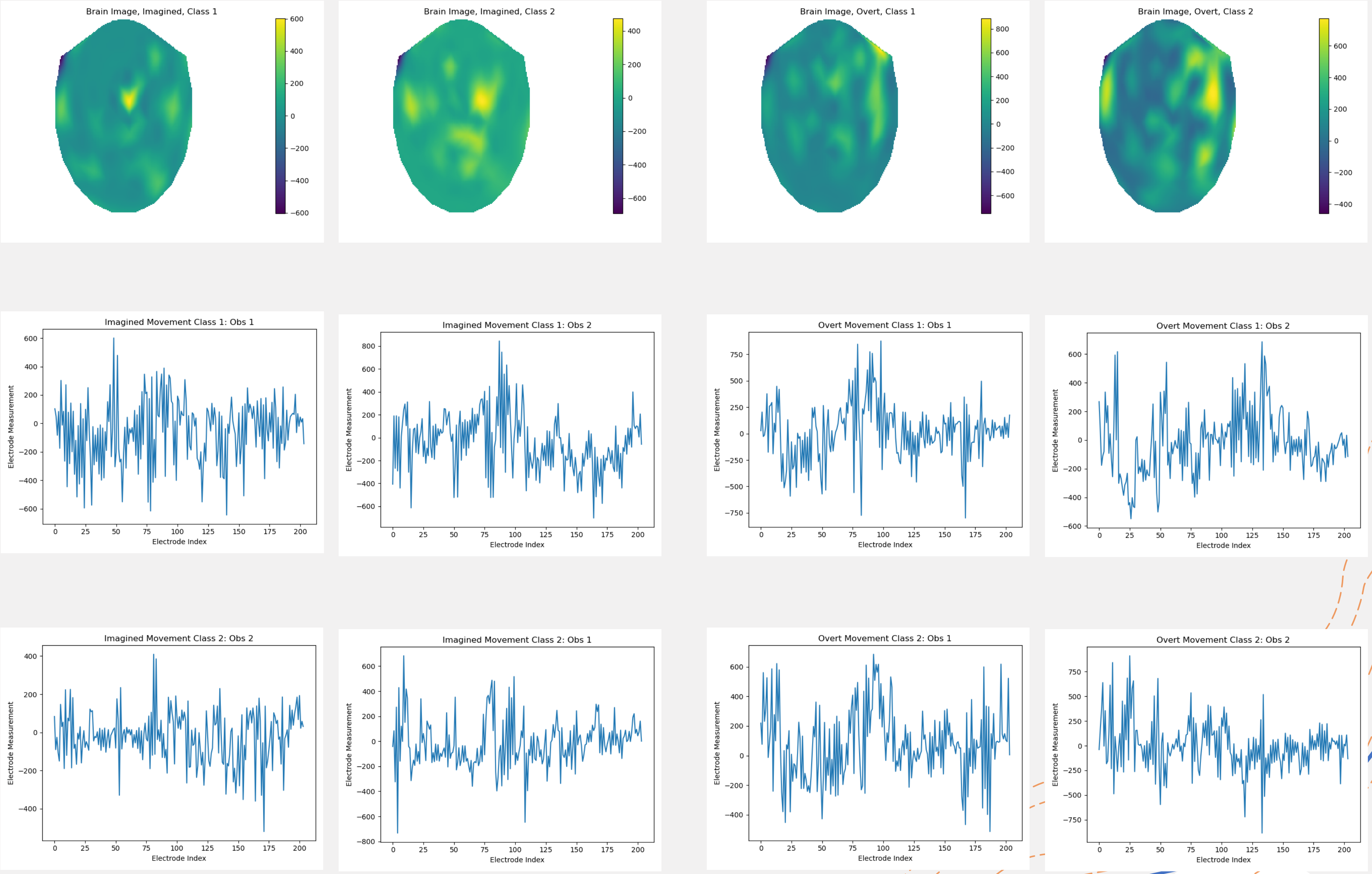
The importance of this experiment is because of the increased focus in understanding neuroscience, modelling the human brain in neural networks and using brain interfaces. By being able to accurately classify how someone wants to move their hands, if there is a problem with their spinal chord, a medical profession could scan their brain and use a mechanical substitute to complete their intended task even without the individual having control of their limbs. An modern example of brain computer interfaces is Neuralink. This company aims to improve the lives of people with paralysis by recording the electrical signals in the brain and translating them into the control of an external device. Much of the training is done by asking an individual to imagine moving their hand in certain motions, similar to this experiment ("Neuralink"). Although we do not know what type of machine learning models they are using, the applications of methods such as SVM to brain computer interfaces are currently being investigated and will have a massive impact on the future of medicine.

Image 2– Cerebellum Location (Kazilek)



# Data

The Overt and Imagined data were provided in two files each, one for the first class and one for the second class. Each class represents imagining movement in a hemisphere of the brain. Overt data is from test subjects physically moving their limbs in either the left or right direction. Imagined data is from test subjects who are imagining moving their limbs in a certain direction. Each dataset had 120 trials with the value from 204 electrodes in each trial. These 204 values come from 102 sensor locations with two electrodes at each position. The first row of plots depict the value of the maximum sensor reading for each position in the brain for two Imagined and two Overt trials respectively. The following eight plots visualise the true value of each of the sensors across the brain.



# Summary of Theory and Impact - SVM

Support vector machines (SVMs) are a type of machine learning model which classify discrete outcomes (classes); in the case of this experiment they aim to classify whether a person is thinking about moving their left or right arms. The SVM acts as a geometric way to think about dividing these few classes. If the data is in p dimensions, the two sets of data can be separated with a p - 1 hyperplane. As is pictured on the right, all of the data is in two dimensions and the hyperplane is in one dimension, or a line.

The solid line is the boundary and is defined by maximising the distance between the points on either side of the line. However, some datasets cannot be separated merely by a flat hyperplane in p-1 dimensions. Thus the data is projected into a higher dimensional space using a 'kernel' and the plane splits the higher dimensional space rather than the original data. This can be observed in plot b, c and d on the right. In image b one can imagine the orange data being projected out of the plane of the page and the blue data being projected into the plane of the page. Thus the plane of the page can split the two datasets and result in the yellow and green delineations.

An advantage of SVMs is that not all of the data is needed to define the model. For example, in image a, only the orange point closes to the boundary and the blue point closest to the boundary are needed for the boundary to take into account the distance from each. This means that considerably less data is needed to still have an accurate classification of the different classes. The dimensions of the data do not affect the classification because the hyperplane is merely in p - 1 space and not dependent on individual features in different dimensions. (Marc Peter Deisenroth et al.)

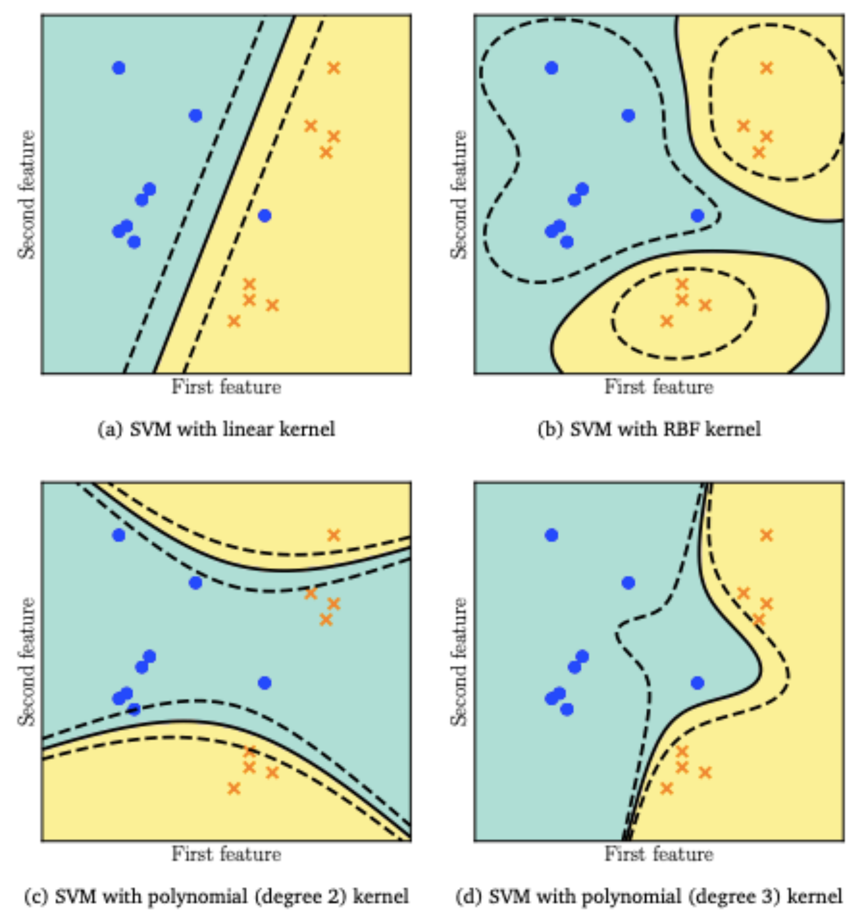



Image 2- SVMs with different kernels (Marc Peter Deisenroth et al.)

# What is Cross Validation

Cross validation is used in Machine Learning to ensure that the model created accurately predicts the data. Since, in a real life scenario, a scientist does not have data to test whether their model is accurate, the training data is split into various groups called folds. The model is then trained with all of the folds except one and then tested on the other folds. This is repeated until different models have been trained with all of the data and tested on all of the data (Müller and Guido).

This project uses a two level cross validation to ensure that the models are representative of the data and to ensure that the model is being tested with the best possible hyperparameters.

The data is first split into six folds: I, II, III, IV, V and VI. Fold I is set aside for training and a model is trained on Folds II, III, IV, V and VI. However, what is unique about two – level cross validation is that rather than just training a single model on II, III, IV, V and VI, multiple models are trained on these folds with different hyperparameters. In this second layer, II is set aside and a model is trained with the first hyperparameter, in this case 'C' = 0.0001, on III, IV, V and VI. This is then cross validated by testing on III, IV, V and VI respectively with different models trained with 'C' = 0.0001 and the other four respective folds.

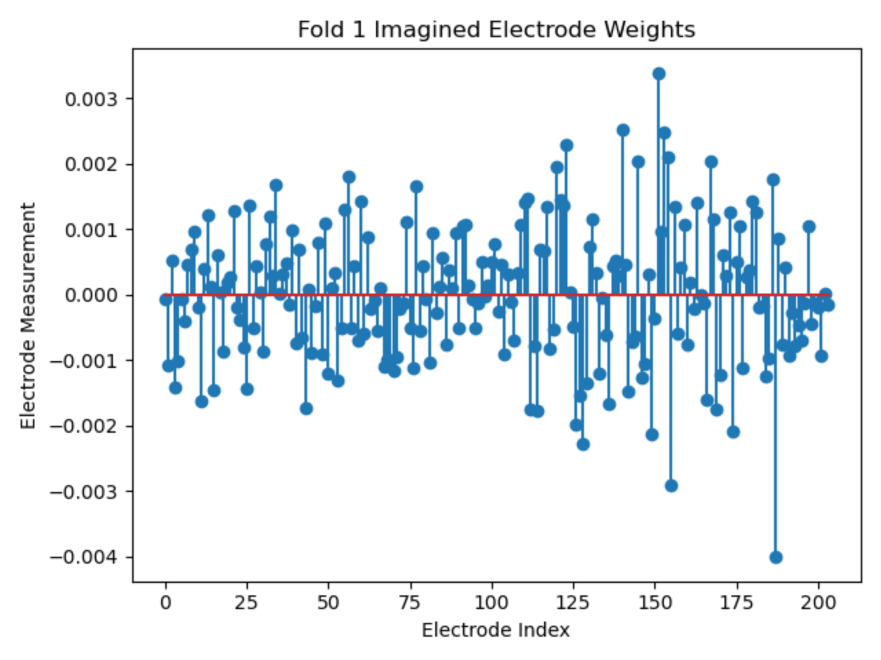
The advantage of using this method is that the hyperparameters are optimised at the same time the model is optimised and verified. Thus the most effective model can be chosen and used for predictions.

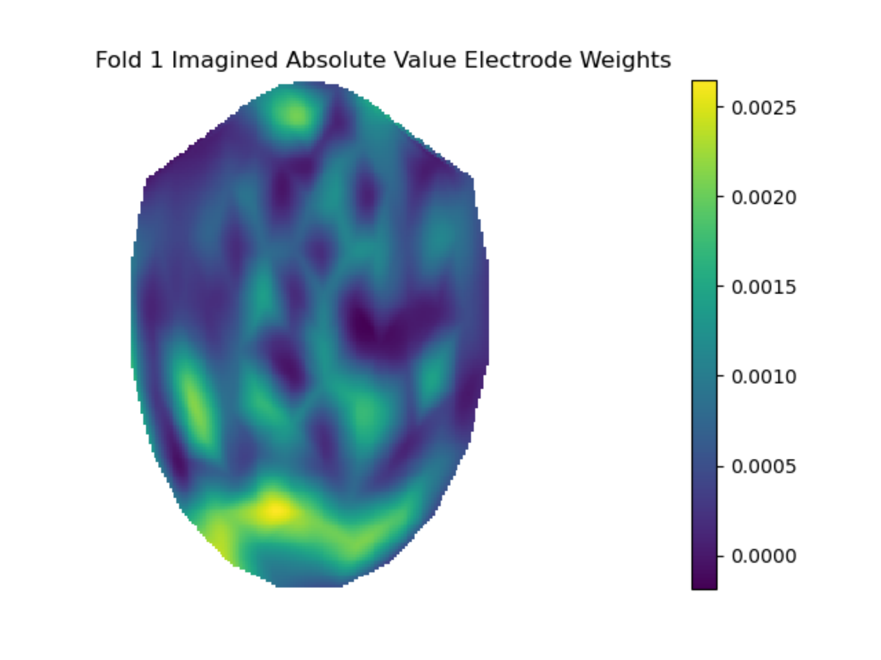
# Channel Weights - Imagined

A model for the Imagined data was created using the two level cross validation technique described in the previous slide. The first fold of the cross validation was taken and the weights the model assigned were visualised in different ways. The value of the weights of the electrodes are shown in the plot on the top right. The absolute value of these same values are plotted on the brain image below. The absolute value is used because positive values represent a high likelihood that the movement is class 1 movement, but a negative value is class 2 movement. For this experiment, it is important to consider both left and right movement and thus it is important to consider the maximum positive and maximum negative values. To do this, the absolute value of the magnitudes are plotted to show the parts of the brain that are most important to determining movement. Similarly, the electrodes with the six most important weights were determined in a table. It is clear to see that the highest peaks and lowest troughs on the first plot match the indices of the highest magnitude electrodes as expected.

Table 1 – Weights of Six Highest Magnitude Electrodes

| Ranking | Weights | Index |
|---|---|---|
| 1 | 0.00400469 | 187 |
| 2 | 0.00338697 | 151 |
| 3 | -0.00290085 | 155 |
| 4 | 0.00252529 | 140 |
| 5 | 0.00247934 | 153 |
| 6 | 0.00229667 | 123 |



# Channel Weights - Overt

The same plots as were on the previous page were repeated for the Overt data. Similarly to the Imagined data, the back of the brain had the electrode values that were the most important for determining movement. Interestingly, by comparing with the Imagined brain plot of the electrode weights, the highest Imagined weights were on the order of 0.002 but for the Overt data the highest weight was only 0.0016. This is interesting because throughout the analysis it was clear that the Overt model was more accurate than the Imagined model, as is expected because more neurons are expected to fire when someone is physically moving their limbs rather than just thinking about moving them. Therefore the higher weight shows the model's reaction to noise by weighting important electrodes higher to avoid bias. The most important electrode weights were located at the back of the skull, in the same location as readings from a cerebellum, as expected.

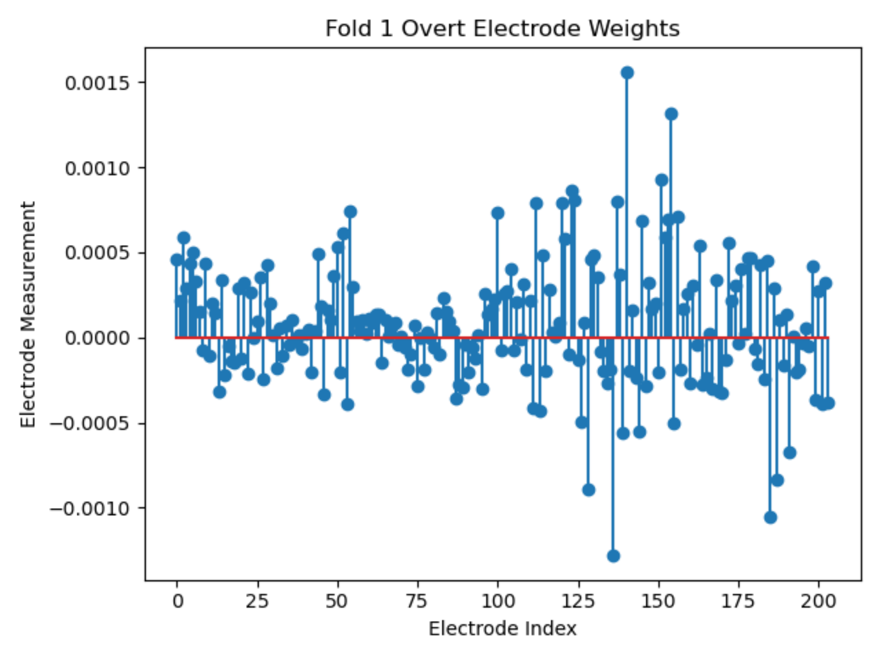
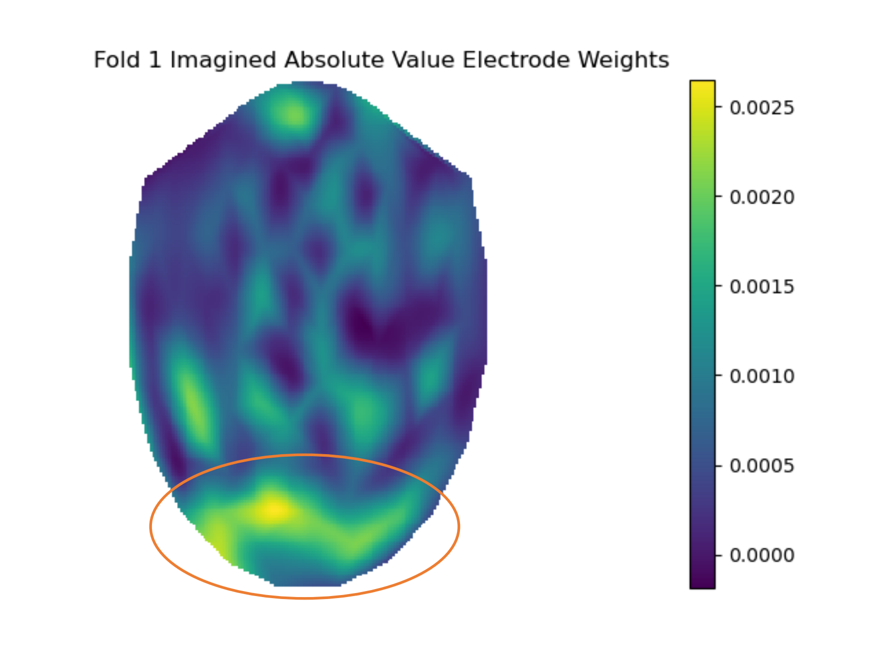
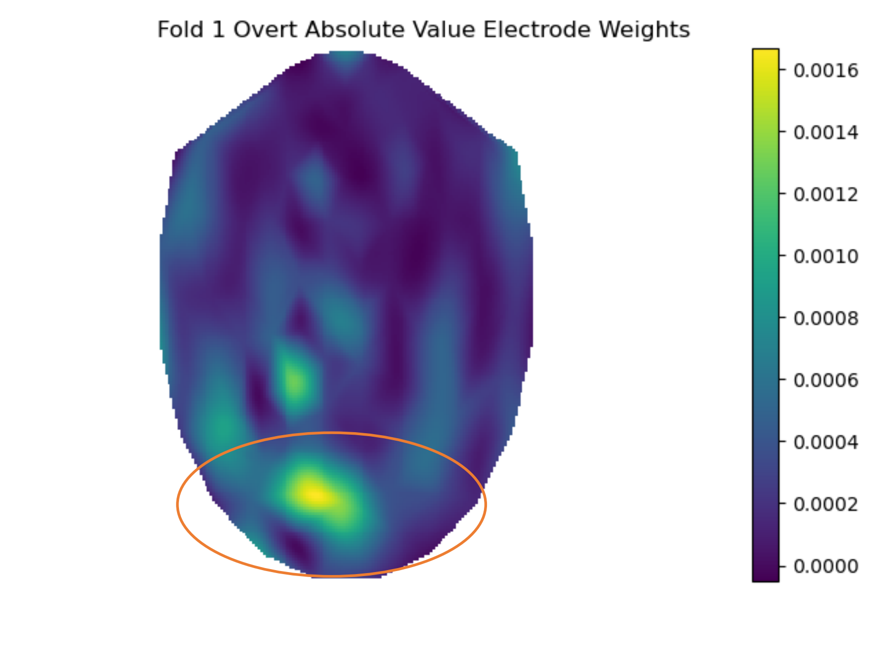



Table 2 – Weights of Six Highest Magnitude Electrodes

| Ranking | Weights | Index |
|---|---|---|
| 1 | 0.00155942 | 140 |
| 2 | 0.00131596 | 154 |
| 3 | -0.00128183 | 136 |
| 4 | -0.00105102 | 185 |
| 5 | 0.00092672 | 151 |
| 6 | -0.00088954 | 128 |



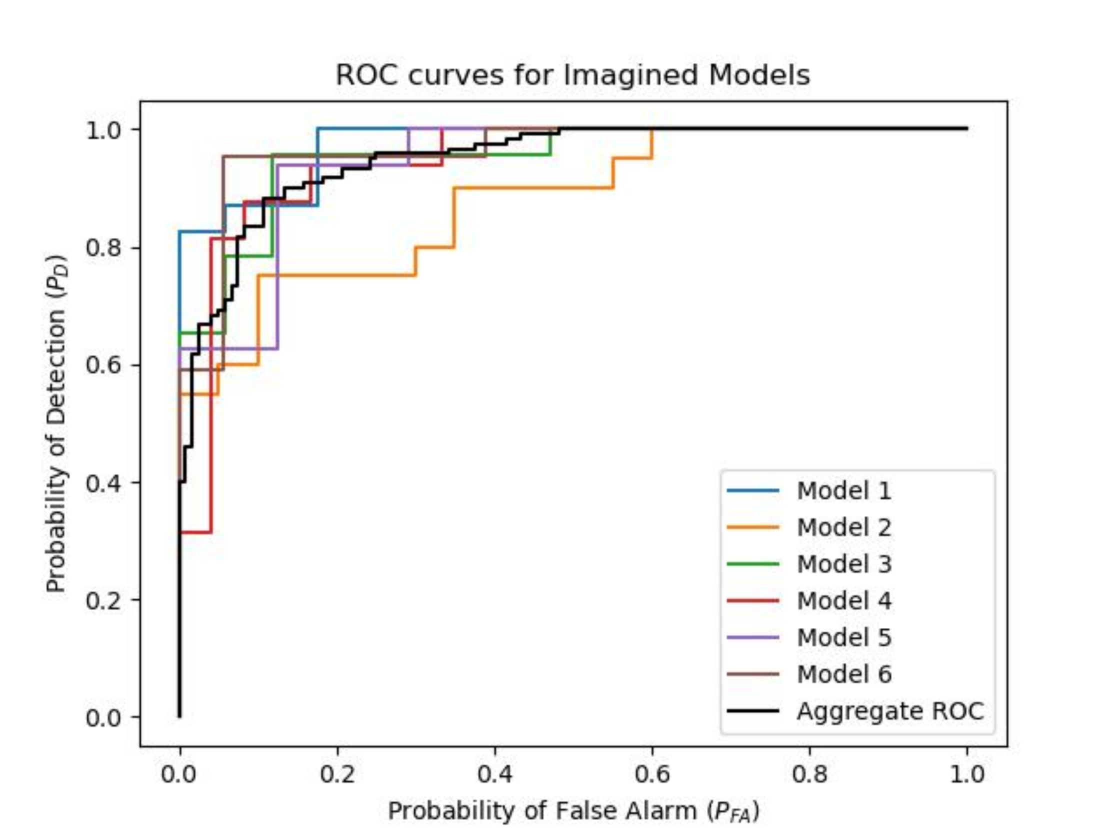
# Imagined ROCs

In the two level cross validation, an optimal model was chosen for each of the six folds. This model was tested on the respective testing fold and the results were plotted on an ROC. The decision statistics for all of the test scenarios were then aggregated to determine the average accuracy of each model. Various models performed differently on their testing data, however the accuracy stayed comfortably between 0.835 and 0.88. As is clear from visual analysis the aggregate ROC was representative of the CV models.

It is important to note that although there appear to be discrepancies such as Model 2 appearing to be the worst on the ROC plot but having an accuracy of 0.85, (which is much higher than model 3's accuracy of 0.835) this is not the case because the ROC was created from testing on the top level cross validation fold, but the accuracy score was created from the inner CV fold using the "best_score_" accuracy value from GridSearchCV.

Table 3 – Model Accuracy Scores

| Model | Accuracy |
|---|---|
| 1 | 0.865 |
| 2 | 0.85 |
| 3 | 0.865 |
| 4 | 0.835 |
| 5 | 0.875 |
| 6 | 0.88 |
| Aggregate | 0.86 |



# Overt ROCs

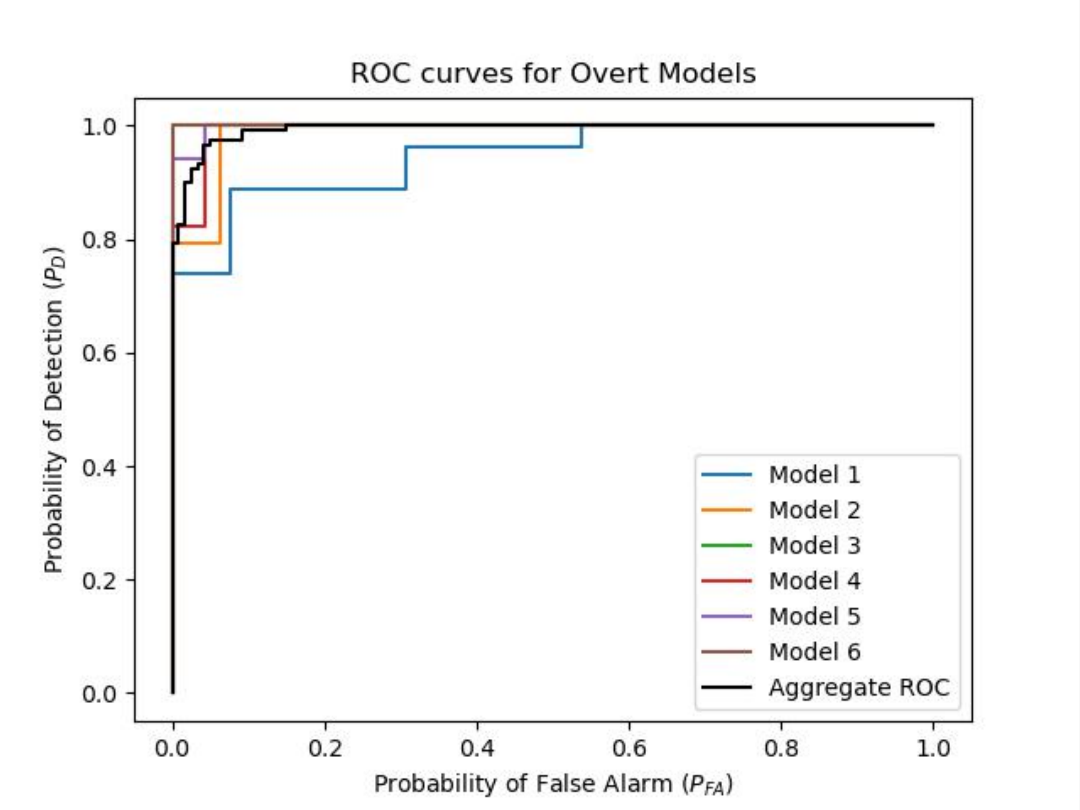



Table 4 – Model Accuracy Scores

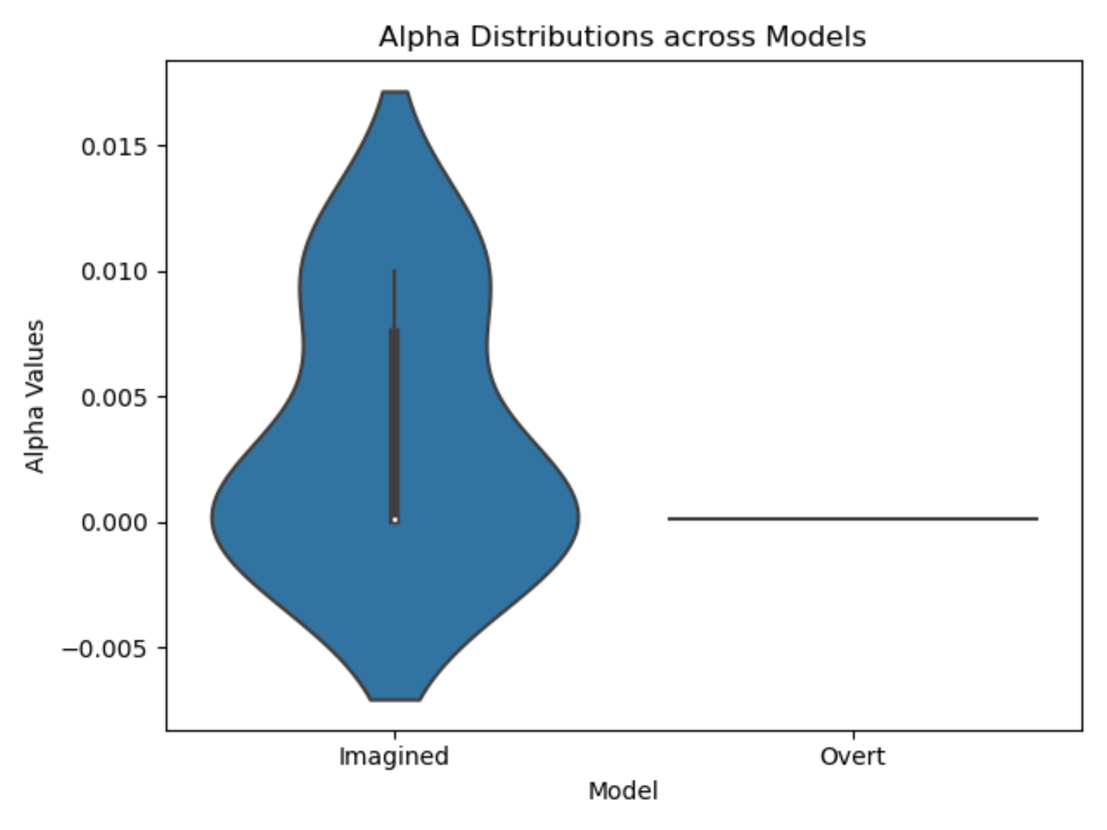
| Model | Accuracy |
|---|---|
| 1 | 0.94 |
| 2 | 0.92 |
| 3 | 0.92 |
| 4 | 0.935 |
| 5 | 0.95 |
| 6 | 0.925 |
| Aggregate | 0.932 |

As was expected from looking at prior literature, the ROC curves for the Overt data were much higher and more consistent than the Imagined data. When someone moves their arm there is a much more predictable set of neurons firing than when someone thinks about moving their arm. Because of this, it is easier to get an accurate model and thus any predictions are more accurate.

Additionally, because the data is more clear for all the Overt trials, it is easier to predict Overt movement, and thus Overt predictions are more consistent across folds.

For the Overt ROC, it is possible to tell that the aggregate accuracy and the aggregate ROC are in the middle of the other models as expected. The spread of accuracy values for the Overt test was only 0.03 further highlighting that it is a reliable model.

# Regularisation Parameters




During the GridSearchCV process, the regularisation constant was optimised for the model of each fold. The strength of the regularisation is inversely proportional to the C values (sklearn.svm.svc). The possible values that the algorithm tested were C = [0.0001, 0.01, 1, 100, 10000]. As can be seen in the table below, the C value chosen for all of the Overt models was 0.0001. Since a smaller C value causes a more restricted model wherein individual datapoints have less influence (Müller and Guido), it clearly shows that the Overt model had more clear delineations between the left and right hemispheres and thus needed less information to accurately classify the data. On the other hand, a few of the Imagined C's were of a much higher order of magnitude than the Overt C's, which supports the postulate that the Imagined model must be more complex to accurately pick up on the delineations between different classes over the surrounding noise; it is imperative that the noise does not change the datapoint and thus they should be made to have less influence.

Table 5 – Regularisation Constant for CV Models

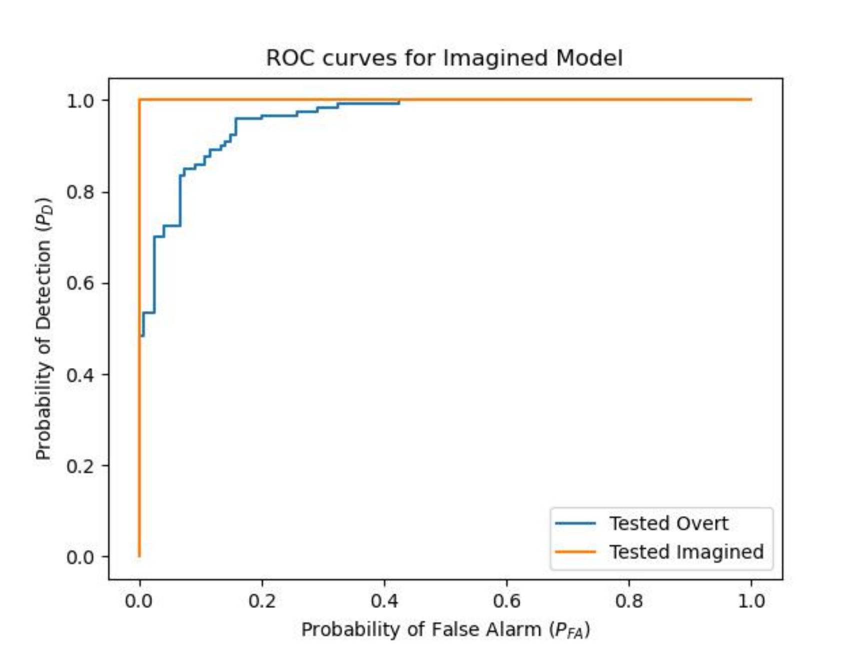
| Model No. | Imagined C | Overt C |
|---|---|---|
| 1 | 0.0001 | 0.0001 |
| 2 | 0.01 | 0.0001 |
| 3 | 0.01 | 0.0001 |
| 4 | 0.0001 | 0.0001 |
| 5 | 0.0001 | 0.0001 |
| 6 | 0.0001 | 0.0001 |

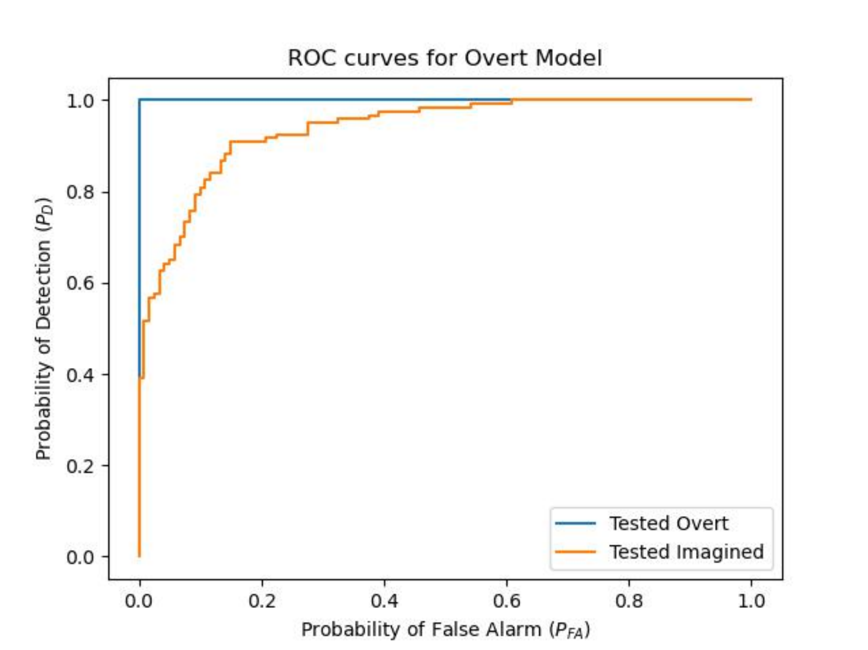
# ROC Compare

In the plots on the right, a model is trained using the optimal regularisation parameter and then it is tested on various datasets. When the data is tested on the same data it is trained on, incestuous training, the ROC is perfect. When the data is trained on the Imagined data and tested on Overt data, the accuracy of the model is higher than when the model is trained on the Overt data and tested on the Imagined. This is due to the inherent noise in the Imagined dataset.

Because the neurons fire less specifically when someone imagines moving their arm rather than actually moving their arm, there is more noise within the data. As was previously discussed, since the Overt model has a very small C it is a more simple model than the Imagined model. Thus, when applied to the more noisy Imagined data, the model performs worse. On the other hand, since the Imagined model is trained on more noisy data, it must be fitted more closely to the true boundary between classes to correctly classify. Thus, when it is tested on the less noisy Overt data it is more likely to correctly classify.

If I were able to design the training data to have a different quality, I would increase the noise slightly. Clearly the models are still performing well with cross testing, but by including noise in the original model it results in a more complex model that divides the boundaries between classes more accurately. Of course, increasing the noise during training too much may result in a faulty model, so a balance must be made to account for the bias- variance trade-off.



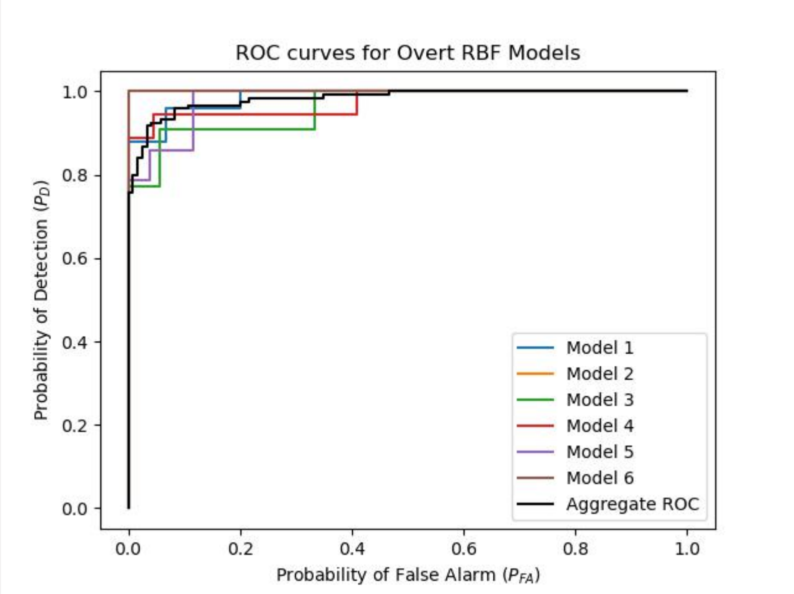



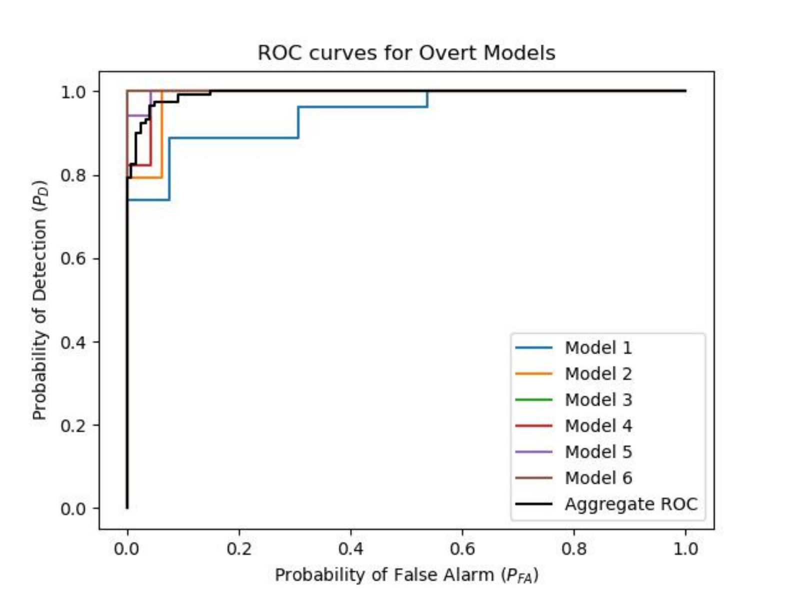
# Explorations

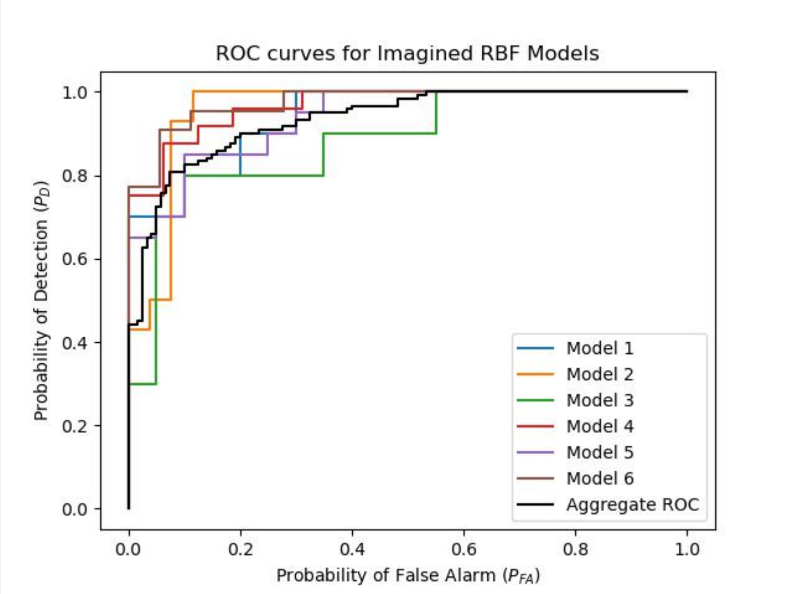
To further explore Support Vector Machines, an RBF model was used. A Radial Basis Function model transforms the data into a higher dimension to separate any data which would have a nonlinear boundary into two sections that can be separated by a linear hyperplane. This is explained in further detail on the SVM slide.

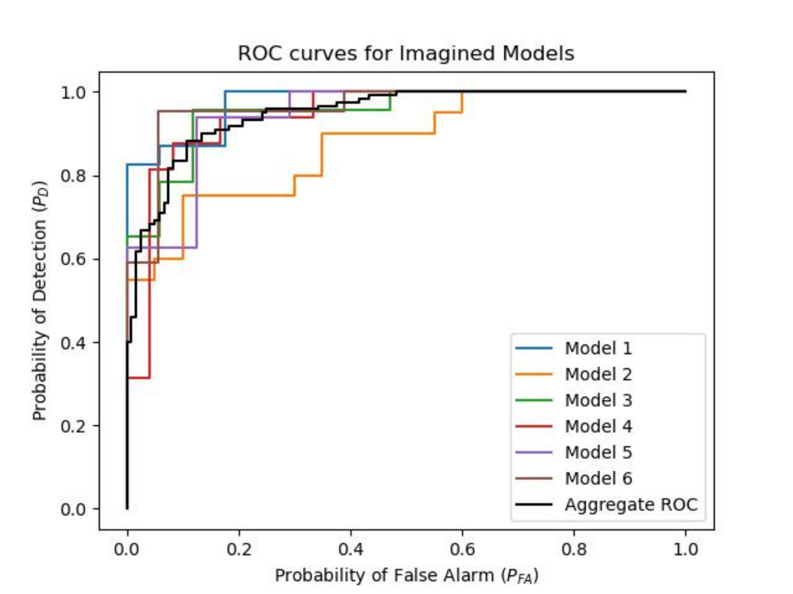
Clearly the RBF did not improve the model much, although the runtime for the model creation was much shorter. The reasoning behind why the RBF did not improve the model is because the data was already so clearly linearly divisible.

In summary, the RBF was an unnecessary model to run, but if further experiments were run with more complicated demands from the brain, perhaps such as moving the same limb one direction or another, the data may be more closely packed and a non linear kernel could be effective.



# Future Work and Conclusions

In the future it would be interesting to further explore choosing certain models from the cross validation and testing with them. Additionally, using other similar modelling methods such as RBF could be interesting to explore. On first hypothesis, however, since the data is clearly linearly separable, it is not necessary to define the data by its centre of mass, since the boundary is the more accurate representation of the division. RBF’s are better at characterising data that does not have a linear kernel.

In conclusion, this experiment was very interesting, because it was possible to identify the clear link between imagined and actual movement, and possible to characterise which way an individual was moving based on their brain signals. The location of the cerebellum was very obvious since the most highly weighted sensors were all located at the back of the skull. The linear SVM model chosen was very effective at characterising the different classes, but would perform better if trained on imagined data due to the increased noise.

# Collaborations

*Who did you share and debate ideas with while working on this project?*

I did almost all of the project entirely alone. During class discussions I worked with Matthew Blume and Joanna Peng and we cogitated about the different ways to display the change in alpha, because they were so similar between the six models for both experiments.

*Who did you share code with while working on this project?*

I did not share code with anyone during this project.

*Who did you compare results with while working on this project?*

To ensure that my results were of the correct order of magnitude I asked Joanna Peng whether her weights were of the order of 0.01. I also asked her whether her ROC's with the best model were perfect. During one of the checkpoints I discovered that my ROC which wasn't perfect was supposed to be perfect and this was because I had my data transposed.

*Who did you help overcome and obstacle and vice versa while working on this project?*

Ben Matz and I discussed what the aim was for the cross validation, because I was unsure whether the data should be split once manually and then once using GridSearchCv or whether GridSearchCv already split the data into two levels. We also discussed whether the optimal method for choosing the model to do the Imagined Overt Cross Testing was by choosing the model with the best accuracy score and or by training a new model with the best regularisation parameters decided by the cross validated models.

I helped Aryan Mathur decide how to create an aggregate ROC by using the decision statistics from previous models in the cross validation

# References


- Decety, Jean. "Do Imagined and Executed Actions Share the Same Neural Substrate?" *Cognitive Brain Research*, vol. 3, no. 2, Mar. 1996, pp. 87–93, https://doi.org/10.1016/0926-6410(95)00033-x.
- Evarts, Edward V. "Brain Mechanisms of Movement." *Scientific American*, vol. 241, no. 3, Sept. 1979, pp. 164–179, https://doi.org/10.1038/scientificamerican0979-164. Accessed 15 Nov. 2019.
- Kazilek. "Parts of the Brain | Ask a Biologist." *Asu.edu*, 5 May 2011, askabiologist.asu.edu/parts-of-the-brain.
- Marc Peter Deisenroth, et al. *Mathematics for Machine Learning*. Cambridge, United Kingdom ; New York, Ny, Cambridge University Press, 2020.
- Müller, Andreas C, and Sarah Guido. *Introduction to Machine Learning with Python : A Guide for Data Scientists*. Beijing, O'reilly, 2017.
- "Neuralink." *Neuralink.com*, 2021, neuralink.com/approach/.

# Packages


- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*, 357–362. https://doi.org/10.1038/s41586-020-2649-2
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95.
- McKinney, W., & others. (2010). Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 51–56).
- Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., … SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2
- Waskom, M., Botvinnik, Olga, O'Kane, Drew, Hobson, Paul, Lukauskas, Saulius, Gemperline, David C, … Qalieh, Adel. (2017). *mwaskom/seaborn: v0.8.1 (September 2017)*. Zenodo. https://doi.org/10.5281/zenodo.883859