MSc Data Analytics (DAA)

Project Proposal

Predicting visual stimuli from MEG recordings of human brain activity using a Convolution Neural Network

April 2015

Shreekanth PAVAR

Department of Computer Science and Information Systems

Birkbeck College

University of London

CONTENTS

[1] Presentation of the Problem - Aims & Objectives

The aim of this project is to predict the category of a visual stimulus presented to a human test subject using MEG (*Magnetoencephalography* **[1]**) recordings of their brain activity. This is known as *Brain (or Neural) Decoding* **[2]**.

Magnetoencephalography is a non-invasive technique for measuring the electrical activity of neurons in the brain **[3]**. It is particularly useful because:

1. it allows brain activity to be recorded over time on a millisecond by millisecond basis
2. it shows where in brain this activity occurs

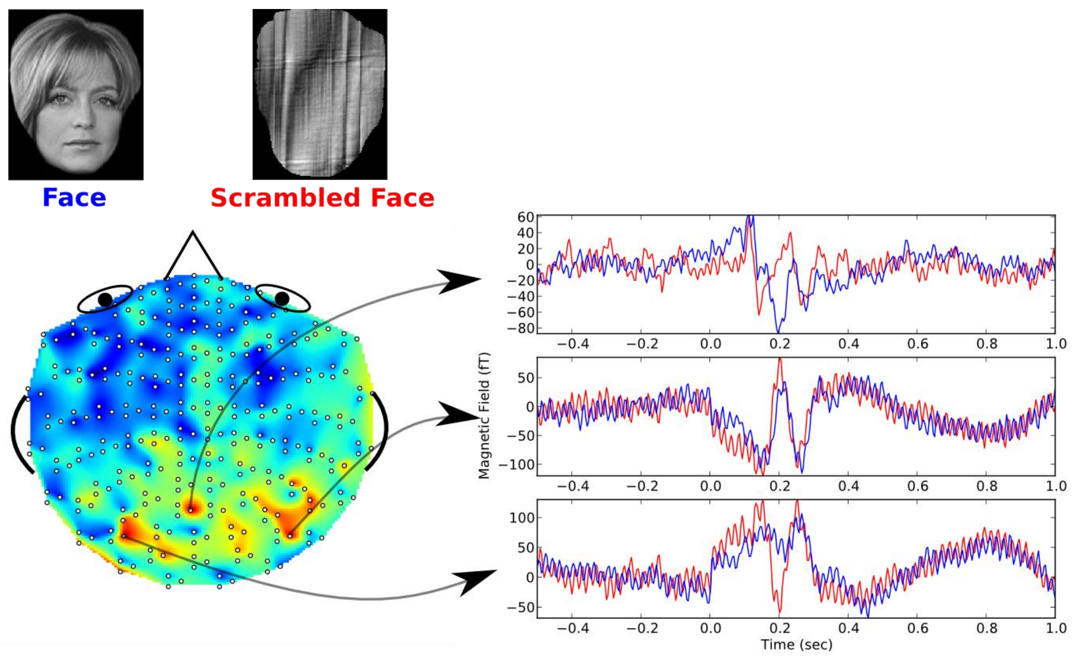
The stimuli consisted of 2 different images of human faces:

1. "unscrambled" faces - where normal facial features (such as eyes, nose, mouth etc) are recognisable.
2. "scrambled" faces - where the normal facial features are no longer recognisable.

If a relationship can be found between the pattern of the recorded signal - known as the *event-related brain potential* (ERP) **[4]** - and the category of the stimulus, this would aid in understanding how the brain recognises human faces - also known as *Face Perception* **[5]**.

In the terminology of *Machine Learning* **[6]**, this type of categorisation is a form of *Supervised Learning* **[7]** known as *(Statistical) Classification* **[8]** and the predictive model produced is called a *Classifier*.

The data to be used in this project was collected as part of a neuroscientific experiment which aimed to integrate data obtained using a variety of techniques (including MEG recordings) from multiple different subjects **[9]**. It was made publically available as part of a *Kaggle* competition **[10]** which provided the inspiration for this project (i.e. to solve the "inverse problem" of predicting a stimulus from the concurrent brain activity). The images below are taken from the competition webpage.



Approximately 580-590 randomised *trials*, each consisting of an image and the associated MEG recording, were performed for 23 different human subjects. Brain activity was captured using a SQUID array which recorded 306 time series at 1KHz of the magnetic field induced by the electric currents flowing through the neurons. This device took measurements for 1 second per stimulus but started recording 0.5sec early making each MEG "trace" 1.5 seconds in length.

These recordings were then separated into datasets for TRAINING and TESTING. The training dataset contains *labelled* data and consists of 9414 trials from 16. This labelling was done manually "by hand" using a human to perform the categorization between scrambled and unscrambled faces. The related class labels were "Face" (*class 1*) or "Scrambled Face" (*class 0*)**.** This will be used to create a model (i.e. the *classifier*) which will be evaluated using *unlabelled* test data comprising 4058 MEG recordings from 7 different subjects.

According to the *Efficient Coding Hypothesis* **[11]**, if the brain is presented with a variety of different stimuli, the neurons adapt to their statistical properties, encoding those which occur most frequently. Hence, it should also be possible to decode these neural responses to reconstruct the stimulus that was observed.

This project aims to perform *Neural Decoding* using MEG recordings of brain activity in response to two different visual stimuli. It assumes that experimental subjects are able to "recognise" a human face where the process of recognition involves the firing of a specific group (or network) of neurons inside the brain which is recorded in the MEG data**.**

A novel approach using a *Deep* (i.e. consisting of one or more "hidden" layers) *Convolutional Neural Network (CNN)* **[12]** as the Classifier will be attempted. CNNs are currently being applied to problems of image recognition **[13]** (including face perception) so the results of this project could potentially contribute to current research.

The labelled *training data* will be used to train the CNN by minimising the classification error of the outputs by modifying the input weights via *Back-Propagation* using *Stochastic Gradient Descent* as the optimization method **[14]**. The unlabelled *test data* will be used to evaluate the *Classification Accuracy* (i.e. what fraction of trials are correctly classified as either "face" or "scrambled face") of this approach. This will then be compared with the accuracy of the alternative approaches used by the winners of the Kaggle competition.

In Machine Learning, *Qualitative* (as opposed to *Quantitative*) or *Categorical Classification* is typically performed using *Logistic Regression* **[15]** which predicts a binary response based on one or more predictor variables (features) using a probabilistic approach.

All three winning entries in the *Kaggle* competition:

1. Used some form of logistic regression combined with other algorithms (i.e. an *Ensemble Method* **[27]**)
2. Identified the inter-variability of the data between subjects as a significant problem which needed addressing and came up with a solution to deal with it **[23]**.
3. Decided to train the classifier iteratively, incorporating test data with predicted labels into the training data in successive iterations.

The rationale behind the above choices and the individual approaches they chose to employ are described in the next section as part of the *Background Research* carried for the project.

[2] Background Research

Cognitive Neuroscience **[16]** is an interdisciplinary branch of both Psychology and Neuroscience concerned with understanding the mechanisms behind the mental processes in the brain. Prior to the advent of *Digital Imaging* this was necessarily accomplished via invasive *Neurosurgery*, however imaging techniques now allow the brain to be studied non-invasively. *Functional Neuroimaging* **[17]** includes techniques that quantifies some metric of brain activity such as;

1. Functional Magnetic Resonance Imaging (fMRI)
2. Positron Emission Tomography & Single Positron Emission Computed Tomography (PET/SPECT)
3. Electroencephalography (EEG)
4. Magnetoencephalography (MEG)

Also refer to [http://psychcentral.com/lib/types-of-brain-imaging-techniques/0001057]
and [http://blog.ketyov.com/2010/07/what-can-we-measure-using-neuroimaging.html]

[2.1] Comparison of fMRI with EEG and MEG

fMRI and PET/SPECT differ from EEG and MEG in that they do not measure neuronal activity directly. Instead, they measures changes in blood flow through the brain.

EEG and MEG both directly measure neural activity in the brain. EEG achieves this by placing electrodes on the scalp to measure the electrical signals produced by large groups of neurons firing together. In contrast, MEG measures the magnetic fields (electromagnetically) induced by this electrical activity using sensitive magnetometers known as *Superconducting Quantum Interference Devices* (SQUIDS) **[18]**.

An important advantage of both EEG and MEG over fMRI is that they can detect changes in electrical activity at the millisecond-level and record these as time series data. Additionally, they can both be recorded simultaneously so that data from these complementary techniques can be combined **[19]**.

However, they both suffer from low signal to noise ratio making it difficult to extract the relevant ERPs. Additionally, EEG suffers from poor spatial resolution **[20]**. In comparison, because magnetic fields can penetrate the bone making up the skull, the spatial resolution of MEG is comparable to that of fMRI while still maintaining the superior temporal resolution of EEG. Also, through the use SQUIDs, MEG can measure the same neural signatures as EEG but with less background "noise" **[21]**. However, like EEG, the signal is also biased toward cortical sources.

[2.2] The Efficient Coding Hypothesis and Neural Encoding vs. Decoding

The *Efficient Coding Hypothesis*, put forward by H Barlow in 1961 **[11]**, proposed that the pattern of electrical activity observed in neurons (i.e. electrically excitable nerve cells) in response to stimulus was not random but instead somehow representative of external stimuli in the outside world. Barlow interpreted the electrical impulses as a code which represented the sensory information provided by the stimulus.

This *Neural Encoding* of sensory information in the neurons implies a relationship between stimulus and response, which can be used to understand how the brain responded to different types of stimulus in order to construct models which attempt to predict responses to other stimuli.

The equivalent "inverse problem" refers to the reverse mapping of using a known pattern of response to infer he stimulus which provoked it. This *Neural Decoding* is the aim of this project, however, Poldrack **[22]** cautions that this kind of "reverse inference" is only deductively valid if there is a one-to-one mapping between specific patterns of neural activity and their associated stimuli, but this is not always the case (i.e. Correlation does not imply Causation).

[2.3] Ensemble Methods and Stacked Generalisation

As there are differences in both brain anatomy and function between humans, some variability is expected between the data of different subjects and consequently, also between the training and testing datasets **[23]**.

Olivetti et al. **[24]** surveyed of the scientific literature on the problem of decoding across subjects for inferential purposes. They proposed a solution which combines a *Transductive Transfer Learning* (TTL) approach with an *Ensemble Method* **[25]** known as *Stacked Generalization (or "Stacking")* **[26]** to deal with the variability across experimental subjects.

Ensemble methods aim to combine multiple learning algorithms to obtain better predictive performance than they produce individually. This "ensemble" typically yields better classification results when there is significant diversity between the algorithms used **[27]**. Unlike other more widely used ensemble techniques, such as "Bagging**"** and "Boosting", "Stacking" allows for classification models of different types to be combined, possibly non-linearly **[28]**.

[2.4] Methodology used by the three winning *Kaggle* competition entries

The winners of the *Kaggle* competition provided documentation describing their methodology and results which was published on the competition webpage **[10]**.

The first place entry, submitted by Alexandre Barachant, was implemented in *MATLAB*. He performed the classification twice. Initially, supervised learning (with labelled data) was carried out using the *lasso* function in MATLAB **[29]** to perform regularized least-squares logistic regression. This was then followed by unsupervised learning (with unlabelled data) using an iterative algorithm similar to that used in *K-means Clusterin*g **[30],** in order to achieve TTL. To avoid the problem of inter-subject variability in the data, the second step was performed independently on each subject. A co-variance matrix was used to store the extracted features which were manipulated using Riemannian geometry.

The second place entry, submitted by Huttunen et al., was implemented in *Python*. It used a hierarchical combination of logistic regression and *Random Forests* **[28]**. They also opted for iterative learning in order to achieve TTL and also trained their classifier independently for each subject for the same reasons as Barachant above. Of particular relevance to this project, Huttunen et al initially experimented with using CNNs as a classifier, which they found to be effective but unstable, with small parameter changes resulting in large differences in accuracy. Moreover, there was a large bias in the predictions, hence they chose to discard this approach in favour of their submitted method, however their findings were educational and will be compared with our own.

The third place entry, submitted by Nathan Hammes, was implemented in *MATLAB.* He also used the *lasso* function for logistic regression but combined this with two Support Vector Machines (SVMs) to perform a 3 step pooled classification.

[2.5] Deep Learning using Convolution Neural Networks (CNNs)

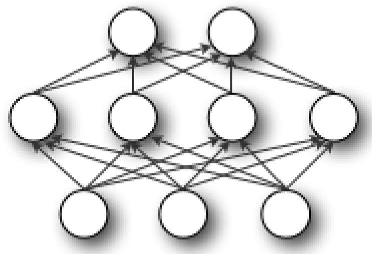
In machine learning, a *Convolutional Neural Network* (or CNN) is a specific type of *multi-layer, Feed Forward, Artificial Neural Network* **[31]**. They were inspired by the biological visual cortex (in animals) and tailored for computer vision tasks by Yann LeCun in early 1990s **[32]**.

Convolution **[33]** is a mathematical operation (similar to *cross-correlation*), which can be considered as applying a function repeatedly across the output of another function to produce a third function which is a measure of their overlap. In the context of neural networks, it means to apply a filter (via the weights assigned to the inputs) over a dataset (via the inputs).

The *Single-Layer Perceptron* (SLP) **[34]** is an algorithm for linear, binary classification. In the context of artificial neural networks, it represents an artificial neuron with a *linear* activation function. *Multi-Layer Perceptrons* (MLPs) are non-linear modifications of the SLP which can classify data that is not linearly separable. They consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. They are another name for Artificial Neural Networks (ANNs).

An ANN can be viewed as a logistic regression classifier where the input is first transformed using a learnt non-linear transformation. This transformation projects the input data into a space where it becomes linearly separable. It is performed in an intermediate layer, referred to as a "hidden" layer.

An (ANN) with a single hidden layer can be represented graphically as follows:



Deep Learning entails the use of a neural network with one more of these "hidden" layers. They are considered to be "hidden" because the nodes (i.e. neurons) in these layers are not accessible from outside the network and can only communicate with other nodes in the adjacent layers.

CNNs can be considered as variations of ANNs specifically designed to require minimal amounts of pre-processing prior to training. As with most neural networks they are trained using the back-propagation algorithm, however they differ in their network architecture. Unlike an ANN, all the layers in a CNN are NOT fully connected but instead contain localised connections and tied weights followed by some form of pooling.

There are three different types of layers:

1. *Convolutional*: which take inputs from a localised subset of the nodes in the previous layer and performs a convolution operation on them by applying a filter via the weights on these inputs.
2. *Pooling*: which subsample the outputs of the convolutional layer.
3. *Fully-Connected*: These layers are identical to those found in ANNs with every neuron in the previous layer being connected to every neuron in this layer.

Hence, unlike ANNs, CNNs do not assume the location of the input data within the dataset is irrelevant and this layer structure results in translation invariant features.

[3] Plan for Developing the Solution

[3.1] Critical issues to be addressed concerning the MEG data

There are two important problems that must be addressed when using MEG recordings for Statistical Inference:

Firstly, MEG data is known to be highly subject specific, so creating a statistical model with good generalisation ability for multiple subjects is difficult as there is considerable variation between the responses of different people to the same stimulus. This is further exacerbated by differences in external environmental conditions as reproducing an identical experimental (lab) setup is problematic. Olivetti et al.'s proposed solution combines a Transductive Transfer Learning (TTL) approach with the use of Ensemble Methods **[24]**.

Secondly, MEG data is noisy because it records ALL brain activity in the range of the measuring device, including unrelated signals from neurons which are not firing in response to the stimulus. Removing or reducing this noise makes it easier to extract relevant patterns of firing as classification features. This can be achieved via signal processing with appropriate filters prior to creating the classification model.

To deal with the noise in the MEG data. *Digital Signal Filtering* will be performed during pre-processing to improve the signal to noise ratio and *Wavelet Transforms* **[35]** will be employed to extract salient features. To tackle the inter-variability between data from different subjects, *Stacked Generalisation* will be applied during the training and testing of the classifier.

However, if problems are encountered with this approach (or if it takes too long!) then, as a "fallback", the pre-processed data from the second winning competition entry will be used to train the classifier (i.e. the pre-processing stage will be "skipped") This is because Huttunen et al. implemented their solution in Python so their code can be easily re-used and if necessary, incorporated.

[3.2]Rationale for using Python as the Programming Language

All the winning entries to the Kaggle competition chose to use either *MATLAB* or *Python*. In particular, two of them utilised the *lasso* function in MATLAB **[30]** to perform regularized least-squares logistic regression. Also, both *MATLAB* and *Python* wrappers exist for *Caffe [http://caffe.berkeleyvision.org/]* which is a deep learning framework (implemented in C++ for use with both CPUs or GPUs) which includes a library for CNNs.

It was decided to implement the solution in *Python*, although *MATLAB* is a viable alternative. There were several reasons for choosing *Python*.

Firstly, *Python* is general purpose language whereas both *MATLAB* and *R* are specialised for numerical computation and statistical data analysis. Hence, *Python* is more widely used and comparatively easy to learn while both *MATLAB* and *R* remain relatively complex programming languages to master.

Secondly, being able to use a single language for both development and analysis has significant benefits. The programming syntax and paradigm remain consistent throughout the code and there is no need to interface between the multiple languages needed for different parts of a project.

Thirdly, and perhaps more importantly from an analysis perspective, data that has been pre-processed in a particular format does not need to be reformatted for use by another language. This data "wraggling" or "munging" is a time consuming, yet relatively unproductive, requirement so minimising the requirement for it by using a single programming language constitutes a significant advantage in terms of reducing the time and effort spent during pre-processing.

Fourthly, another consideration is that Python is open-source and hence free to use whereas MATLAB is proprietary requiring the purchase of a licence. An academic licence is available, but ONLY for use in the DCSIS computing labs. Hence, in the interests of maintaining flexibility (i.e. not being restricted to working in only one location) Python is preferable.

Finally, *Python* is currently widely used within the Machine Learning so there are numerous software tools and libraries available including:

- **MNE** [*http://martinos.org/mne/stable/index.html*]
  - A software package for pre-processing of EEG & MEG data
- **Sci-Kit Learn** [*http://scikit-learn.org*]
  - A library specifically designed for Machine Learning
- **Theano** [*http://deeplearning.net/software/theano/*]
  - A library specifically optimised for GPU computation
- **PDNN** [*http://www.cs.cmu.edu/~ymiao/pdnntk.html*]
  - A deep learning toolkit developed under the Theano environment
- **PyBrain** [*http://pybrain.org/*]
  - A modular Machine Learning library which includes Neural Networks
- **convnet** [*http://libccv.org/doc/doc-convnet/*]
  - An implementation of deep convolutional networks mainly for image recognition and object detection (based on Alex Krizhevsky's work on ImageNet Classification **[13]**).
  There is also a C++/CUDA implementation of this library for use on GPUs called **cuda-convnet** [*https://code.google.com/p/cuda-convnet/]*

[NOTE: However, *MATLAB* can be used as a "fallback" option if necessary as a licence is available and it also has similar libraries and toolkits for working with Neural Networks]

This project intends to use **MNE** to pre-process the MEG training data and **convnet** to build a CNN classifier. For the time series data in the MEG recordings, a one dimensional CNN would be appropriate.

- Why use CNNs?
- Why use Wavelet Transforms?

[3.3] Project Timetable and associated Gantt Chart

The project will be carried out over the 14 week period between Mon 8 Jun - Mon 14 Sep and can be considered to comprised of 5 major sub-tasks with associated milestones:

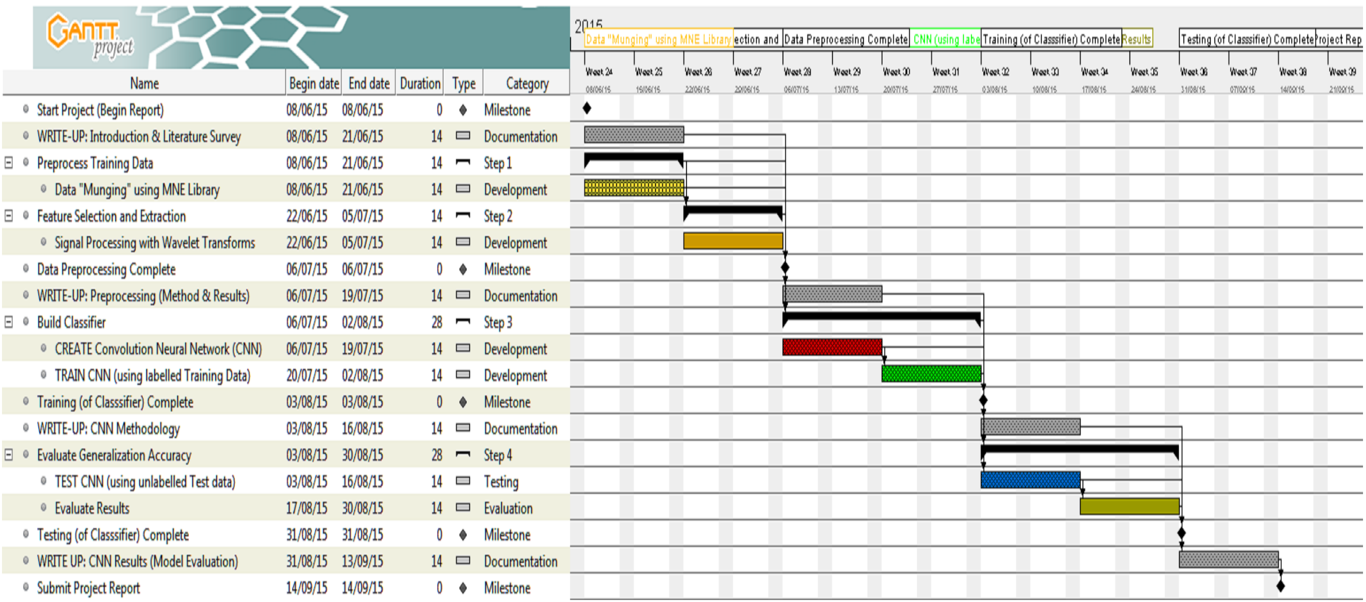
1. Preprocessing of the "raw" MEG recordings to reduce noise (using the MNE toolkit)
2. Feature Extraction using Wavelet Transforms
3. Building the CNN Classifier Model (using the labelled training data)
4. Evaluating the Model Generalisation Ability (using the unlabelled test data)
5. Write-up of the methodology and results in a project report (to be submitted on Mon 14 Sept)

Tasks 1 & 2 and 3 & 4 are expected to overlap as they are closely related. In particular, an "iterative learning" approach to training the classifier will require that predicted labels are used in successive training iterations.

The building and training of the CNN is the primary objective of this project and it is anticipated that this will consume the majority of the available time and effort (i.e. it constitutes the "critical path"). For this reason, one third of the total implementation time (i.e. 4 weeks) is allocated for tasks 1 & 2 whereas two thirds (i.e. 8 weeks) is allowed for tasks 3 & 4.

Also, while fallbacks are not incorporated into the Gannt chart below, the progress of 1 & 2 will be assessed after 3 weeks (i.e. at the end of June) and if necessary the fallback option (of using processed data from the second *Kaggle* competition entry) will be exercised.

The write-up will be performed concurrently with other tasks during the 14 weeks but the last 2 weeks are dedicated specifically to completing the report as it is the only (assessed) deliverable from the project.



[4] References

[1] Hämäläinen M, Hari R, Ilmoniemi RJ, Knuutila J, Lounasmaa OV (1993), *"Magnetoencephalography-theory, instrumentation, and applications to noninvasive studies of the working human brain"*, Reviews of Modern Physics 65 (2): 413–97, Bibcode:1993RvMP...65..413H, doi:10.1103/RevModPhys.65.413

[2] Smith K, *"Brain decoding: Reading minds",* Nature 502, 428–430 (23 October 2013) | doi:10.1038/502428a

[3] http://ilabs.washington.edu/what-magnetoencephalography [Accessed 13 April 2015]

[4] Luck SJ, (2005) *"An Introduction to the Event-Related Potential Technique"*, The MIT Press, ISBN 0-262-12277-4

[5] Eimer M, *"Event-related brain potentials distinguish processing stages involved in face perception and recognition"*, Clinical Neurophysiology, 111 (2000), pp. 694–705

[6] Mohri M, Rostamizadeh A, Talwalkar A (2012) *"Foundations of Machine Learning",* The MIT Press ISBN 9780262018258.

[7] Alpaydin E, (2010), *"Introduction to Machine Learning"*, MIT Press. p. 9. ISBN 978-0-262-01243-0.

[8] Mitchell T, (2006), *"The discipline of machine learning",* (Technical Report CMUML-06-108), Carnegie Mellon University

[9] Henson RN, Wakeman DG, Litvak V, Friston KJ (2011), *"A parametric empirical Bayesian framework for the EEG/MEG inverse problem: generative models for multi-subject and multi-modal integration"*, Frontiers of Human Neuroscience 5:76, doi: 10.3389/fnhum.2011.00076

[10] *DecMeg2014 - Decoding the Human Brain*, https://www.kaggle.com/c/decoding-the-human-brain [Accessed 13 April 2015]

[11] Barlow H, *"Possible principles underlying the transformation of sensory messages"*, Sensory Communication, MIT Press

[12] Convolutional Neural Networks (LeNet) http://deeplearning.net/tutorial/lenet.html [Accessed 13 April 2015]

[13] Krizhevsky A, Sutskever I, and Hinton GE, *"ImageNet Classification with Deep Convolutional Neural Networks"* , NIPS 2012

[14] Amari S, *"Backpropagation and stochastic gradient descent method",* Neurocomputing, Volume 5, Issues 4–5, June 1993, Pages 185-196, ISSN 0925-2312, http://dx.doi.org/10.1016/0925-2312(93)90006-O.

[15] Freedman DA, (2009), *"Statistical Models: Theory and Practice"*, Cambridge University Press, p. 128.

[16] Gazzaniga MS, Ivry R, Mangun GR, "*Cognitive Neuroscience: The Biology of Mind (IV edition)"*, W.W. Norton. 2013

[17] Bunge SA, Khan I, *"Cognition: An Overview of Neuroimaging Techniques"*, Encyclopedia of Neuroscience, 1063-1067 (2009)

[18] Cohen D, *"Magnetoencephalography: detection of the brain's electrical activity with a superconducting magnetometer"*, Science 1972, 175:664-66

[19] Freeman WJ, Ahlfors SP, Menon V, *"Combining fMRI with EEG and MEG in order to relate patterns of brain activity to cognition"*, International Journal of Psychophysiology, Volume 73, Issue 1, July 2009, Pages 43-52, ISSN 0167-8760, http://dx.doi.org/10.1016/j.ijpsycho.2008.12.019

[20] Agrawal P, *"A Probe Into Decoding Brain Activity Using fMRI and MEG"*, Electrical Engineering and Computer Science University of California, Berkeley

[21] Cohen D, Cuffin DN, *"Demonstration of useful differences between the magnetoencephalogram and electroencephalogram"*, Electroencephalography and Clinical Neurophysiology, 1983;56:38-51

[22] Poldrack RA, *"Can cognitive processes be inferred from neuroimaging data?"*, Trends in Cognitive Sciences, vol. 10, no. 2, pp. 59 – 63, 2006

[23] Zaremba W, *"Modeling the variability of EEG/MEG data through statistical machine learning"*, Computer Vision and Pattern Recognition. Ecole Polytechnique X, 2012

[24] Olivetti E, Kia SM, Avesani P, (2014), *"MEG decoding across subjects",* Pattern Recognition in Neuroimaging, 2014 International Workshop on, pp.1-4, 4-6 June 2014, doi: 10.1109/PRNI.2014.6858538

[25] Opitz D, Maclin R, (1999), *"Popular ensemble methods: An empirical study"*, Journal of Artificial Intelligence Research 11: 169–198. doi:10.1613/jair.614

[26] Wolpert DH, (1992), *"Stacked Generalization"*, Neural Networks, Volume 5, Issue 2, Pages 241-259.

[27] Kuncheva L, and Whitaker C, *"Measures of diversity in classifier ensembles"*, Machine Learning, 51, pp. 181-207, 2003

[28] Sewell M, (2008), *"Ensemble Learning"*, http://machine-learning.martinsewell.com/ensembles/ ensemble-learning.pdf, (unpublished) [Accessed 13 April 2015]

[29] *lasso - MATLAB* documentation http://uk.mathworks.com/help/stats/lasso.html [Accessed 13 April 2015]

[30] Hartigan JA, Wong MA, (1979) *"Algorithm AS 136: A K-Means Clustering Algorithm"*, Journal of the Royal Statistical Society, Series C 28 (1): 100–108. JSTOR 2346830

[31] Svozil D, Kvasnicka V, Pospichal J, *"Introduction to multi-layer feed-forward neural networks"*, Chemometrics and Intelligent Laboratory Systems, Volume 39, Issue 1, November 1997, Pages 43-62, ISSN 0169-7439, http://dx.doi.org/10.1016/S0169-7439(97)00061-0

[32] LeCun Y and Bengio Y, *"Convolutional Networks for Images, Speech, and Time-Series",* in Arbib, M. A. (Eds), The Handbook of Brain Theory and Neural Networks, MIT Press, 1995

[33] Bracewell R, *"Convolution and Two-Dimensional Convolution."*, Ch. 3 in The Fourier Transform and Its Applications, New York: McGraw-Hill, pp. 25-50 and 243-244, 1965

[34] Rosenblatt F, (1957), *"The Perceptron--a perceiving and recognizing automaton"*, Report 85-460-1, Cornell Aeronautical Laboratory

[35] Georgoulas GK, Stylios CD, Groumpos PP, *"Feature Extraction and Classification of Fetal Heart Rate Using Wavelet Analysis and Support Vector Machines"*, International Journal of Artificial Intelligence Tools 06/2006; 15(3):411-432. DOI:10.1142/S0218213006002746