Translational Neuromodeling - Report

Linus Rüttimann Elia Torre Saurabh Vaishampayan
ETH Zürich University of Zurich

June 1, 2023

Abstract

This paper presents the work performed in the final project of the Translational Neuromodeling course. The purpose of our project is that of developing a computational approach to model EEG data measured during auditory mismatch negativity (MMN) studies in healthy subjects to which either receptor agonist or receptor antagonist drugs were administered. In order to do so, the data from (Weber et al., 2022) was pre-processed and several different computational models developed [Naive Classification, Laplacian Eigenmaps, Dynamic Causal Modeling (DCM), NeuralHGF and Convolutional Recurrent Attention Model (CRAM)]. The objective of such models is that of inferring which drug has been administered to a specific subject starting from the subject EEG measurements.

I. Introduction

The main focus of this paper is that of presenting the modeling techniques we developed to classify the EEG recordings of healthy subjects to which receptor agonist or antagonist drugs were administered in a auditory mismatch negativity (MMN) setting. In the following work, we will explain the dataset under analysis and the pre-processing perfomed on it. In particular, three different techniques to treat standard and deviant trials are discussed. Then, we developed five different modeling techniques starting from simpler techniques such as naive classification without embedding and with laplacian embedding. Passing by multiple Dynamic Causal Modeling (DCM) embedding architectures. To reach more complex neural network architectures such as a custom Convolutional Recurrent Attention Model (CRAM), and a novel technique to invert and simulate Hierarchical Gaussian Filter (HGF): NeuralHGF.

II. Dataset

The dataset consisted out of EEG recordings from two auditory mismatch negativity (MMN) studies (Weber et al., 2022). In study 1 (antagonist branch) participants were given Biperiden (a selective muscarinic M1 receptor antagonist), amisulpride (a selective Dopaminergic D2/3 receptor antagonist) or placebo. In study 2 (agonist branch) subjects were given Galantamine (an acetylcholinesterase inhibitor), levodopa (a Dopamine precursor) or placebo. In both studies subject were presented with a sequence of 1800 auditory stimuli of two types. The presentation probability of the two stimuli was varying over the experiment. This was done to systematically test whether auditory mismatch responses under varying levels of environmental stability are sensitive to diminishing and enhancing cholinergic vs. Dopaminergic function.
For all data analysis in this report we polled the data from the two studies which resulted in an unbalanced dataset of $N = 51$ placebo samples, $N = 22$ Biperiden samples, $N = 24$ amisulpride samples, $N = 26$ Galantamine and $N = 26$ levodopa samples ($N_{total} = 149$). The two studies were following exactly the same experimental protocol but data was collected independently. We decided to pool the data because of the small sample sizes and because we were interested in comparing agonist with antagonist samples. It should be noted thought that training a classifier on the pooled data bears the risk that the classifier learns to discriminate signal differences caused by slight differences in experimental conditions between the two studies rather than differences stemming from the the different drugs.

III. Pre-Processing of data and definition of conditions

All data analysis in this report was performed on the pre-processed EEG data as provided by the course instructors. The following pre-processing steps were applied on the this data:

- Rereferencing to linked mastoid.
- High-pass filter 0.1 Hz
- Downsampling to 250 Hz
- Low-pass filter 30 Hz
- Eye blink correction
- Epoching between -100 and 452 ms relative to auditory stimulus
- Rejection of bad trials (>75 μV)

We defined trial conditions by averaging the epoched trials in three different approaches that are specified in the following subsections. For each subject, trials that fulfilled the condition criteria specified below were averaged to form one average trial per condition.

i. First Approach: Classic Standard vs. Deviant (StdVsDev)

This approach follows the standard and deviant definition in (Weber et al., 2022). It defines two conditions as follows:

- Deviant: every change after at least 5 repetitions of a tone (N=119).
- Standard: every 6th repetition of a tone (N=106).

ii. Second Approach: Classic Standard vs. Deviant with Stability (StdVsDevStb)

In (Weber et al., 2022) divide the experiment in stable and volatile phases. Stable phases are phases were the probability of hearing tone 1, p, is constant for 100 or more trials and volatile phases are all other phases. We slightly changed this definition and defined stable phases as phases were p did not change for the at least the last 50 trials. We made this change because for the definition of trial conditions only the subjects inference of environmental stability matters and in (Weber et al., 2022) definition it is not possible for the subject to know that it is in a stable phase at the beginning of a stable phase. We then divided the deviant and standard conditions in stable and volatile sub-conditions as follows:

- Deviant, stable: every change after at least 5 repetitions of a tone in a stable phase (N=52).
- Deviant, volatile: every change after at least 5 repetitions of a tone in a volatile phase (N=67).
- Standard, stable: every 6th repetition of a tone in a stable phase (N=47).
- Standard, volatile: every 6th repetition of a tone in a stable phase (N=59).

iii. Third Approach: Extended Standard vs. Deviant (ExtStdVsDev)

The first and second approach are very data inefficient because only data from 225 trials out of 1800 trials is included in the classification. In order to increase data efficiency we define two deviant and two standard conditions in this approach. This results in the use of 396 tones:

- Deviant3-5: every change after 3-5 repetitions of a tone (N=91).
- Deviant6+: every change after at least 6 repetitions of a tone (N=105).
- Standard5: every 6th repetition of a tone (N=106).
- Standard12+: every at least 12th repetition of a tone (N=94).

IV. Modeling Approaches

- Classification without Embedding
- Laplacian Eigenmaps Embedding
- Dynamic Causal Modeling (DCM)
- NeuralHGF
- (Extra): Convolutional Recurrent Attention Model (CRAM)

V. Classification without Embedding

The first modeling technique is a naive classification approach meant to give a baseline estimate of prediction accuracy. In particular, the data derived from the steps described in the previous section is further processed as follows:

1. Standard and Deviants are defined according to the first approach described in the Pre-Processing section to obtain 106 standard trials and 119 deviant trials.

2. Standard and deviant trials are averaged to obtain a data set of shape 149 subjects, 63 channels, 139 samples, 2 trial types.

3. Samples are then concatenated over the 2 trial types to obtain a data set of shape 149 subjects, 63 channels, 139 samples, 278 samples.

4. Finally, the data set is reshaped to 149 subjects, 17514 features.

This data set is then used to predict the drug (or placebo) given to every subject. In particular, we employed four machine learning models:

- K-Nearest Neighbors (KNN)
- Random Forest (RF)
- Support Vector Machine (SVM)
- AdaBoost (ADB)

The four models hyper-parameters have been optimized and trained as explained in the following section.

i. Training & Hyper-parameters Tuning

Each model's selected hyper-parameters have been optimized [1] and their performances evaluated in 10-Fold Stratified Cross-validation. In particular:

- KNN has been fitted to the training data via `fitcknn()`. The following hyper-parameters have been optimized: `Number of Neighbors` and `Distance`. The optimization surface is shown in Figure 1.

- RF has been fitted to the training data via `TreeBagger()`. The following hyper-parameters have been optimized: `Number of Trees`, `NumPTS`, `MinLeafSize` and `MaxNumSplits` via `Quantile Error & Bayesian Optimization` The optimization surface is shown in Figure 2.

- SVM has been fitted to the training data via `fitcecoc()`. The following hyper-parameters have been optimized: `BoxConstraint`, `KernelFunction` and `KernelScale`.

[1] The optimal hyper-parameters set for each model can be found at Gitlab Repository

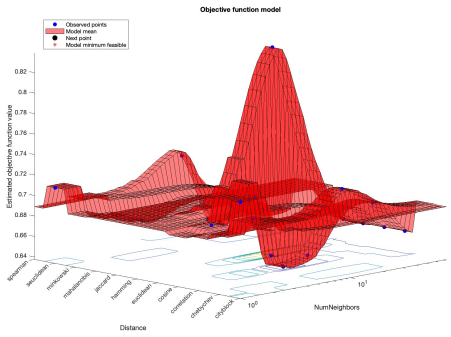

Figure 1: *3D plot illustrating the estimated objective function value with respect to the parameters NumNeighbors and Distance, obtained through the fitcknn (KNN) function in MATLAB. Blue points represent observed points, while the estimated mean performance between observed points is depicted in red.*

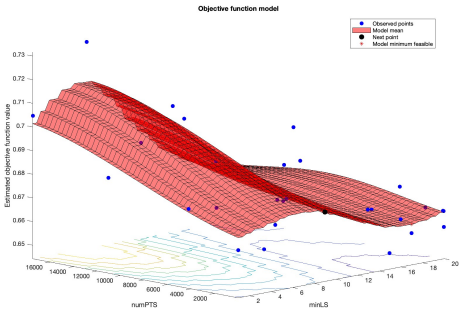

Figure 2: *3D plot depicting the estimated objective function value in relation to the parameters NumPTS and MinLS when optimizing the TreeBagger (RF) model using Quantile Error and Bayesian Optimization in MATLAB. Blue points represent observed points, while the estimated mean performance between observed points is depicted in red.*

- ADB has been fitted to the training data via `fitcensemble()` with `AdaBoostM1` method in case of binary classification and `AdaBoostM2` method for multi-class. The following hyper-parameters have been optimized: `NumLearningCycles`, `LearnRate`, `MinLeafSize` and `MaxNumSplits`.

ii. Results

Following (Schöbi et al., 2021), we ensured the validity of our models performances by comparison with the performance of 30 models trained under label permutations in a 10-fold cross-validation setting. For each classification, the best performing model has been chosen, a Gaussian curve has been fitted to the histograms of accuracies and the p-value estimated. In particular, we performed the following binary and multi-class classifications:

- Galantamine vs Placebo
- Amisulpride vs Placebo
- Levodopa vs Placebo
- Biperiden vs Placebo
- Biperiden vs Galantamine
- Amisulpride vs Levodopa
- All Drugs + Placebo

The results are reported in table 1 and visualized in Figures 3 and 4 The results show that this naive modeling

Classes	Best Model	Accuracy	P-Value
Galantamine/Placebo	RF	0.636	0.859
Amisulpride/Placebo	RF	0.680	0.304
Levodopa/Placebo	RF	0.675	0.099
Biperiden/Placebo	KNN	0.767	**0.002**
Biperiden/Galantamine	RF	0.729	**0.004**
Amisulpride/Levodopa	KNN	0.540	0.378
All Drugs	RF	0.362	**0.001**

Table 1: *Results Classification without Embedding*

technique achieves a good performance in differentiating *Biperiden vs Placebo* and *Biperiden vs Galantamine*. In particular, the models obtain an accuracy above 70% and a p-value < 0.05 in both the cases. However, it is a modeling techniques that lacks both of interpretability and robustness.

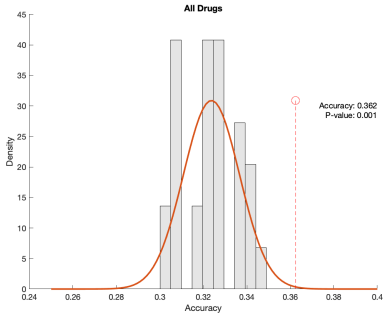

Figure 3: *No Embedding All Drugs Classification Results. Accuracies of 30 label-permuted trained models are shown as grey bars and the accuracy of the chosen model is depicted with a red-dotted line. A gaussian fit is overlaid to the histogram in red and the relative p-value can be found in the legend.*

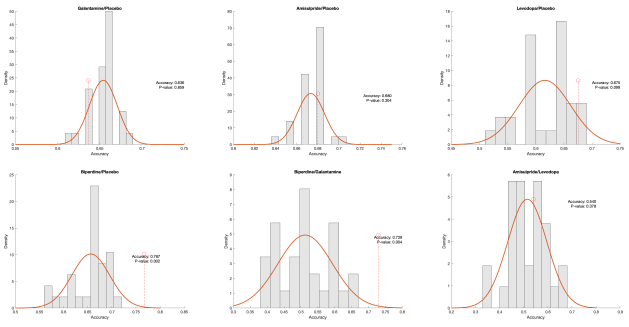

Figure 4: *No Embedding Binary Classifications Results. Accuracies of 30 label-permuted trained models are shown as grey bars and the accuracy of the chosen model is depicted with a red-dotted line. A gaussian fit is overlaid to the histogram in red and the relative p-value can be found in the legend.*

VI. Laplacian Eigenmaps Embedding

Our second modeling approach is inspired by (Gramfort & Clerc, 2007) and (Belkin & Niyogi, 2003), which exploit Laplacian Eigenmaps as a dimensionality reduction technique. Laplacian Eigenmaps is a non-linear dimensionality reduction technique that aims at preserving the local structure of high-dimensional data in a lower-dimensional latent space. In the case of EEG data, we aim to preserve the complex spatial and temporal dependencies embedded in the data. Furthermore, this technique has been proven to be robust to noise. For what concerns the pre-processing of the data, we followed the same pipeline used for the classification without embedding, i.e.:

1. Standard and Deviants are defined according to the first approach described in the Pre-Processing section to obtain 106 standard trials and 119 deviant trials.
2. Standard and deviant trials are averaged to obtain a data set of shape 149 subjects, 63 channels, 139 samples, 2 trial types.
3. Samples are then concatenated over the 2 trial types to obtain a data set of shape 149 subjects, 63 channels, 139 samples, 278 samples.
4. The data set is reshaped to 149 subjects, 17514 features.
5. Finally, we apply the Laplacian Dimensionality Reduction technique to obtain a data set of shape 149 subject, 2 features.

This data set is then used to predict the drug (or placebo) given to every subject.

i. Hyper-parameters Tuning

The four models previously described have been employed, the selected hyper-parameters have been optimized with the same techniques explained above [2] for the laplacian eigenmaps embedded data set and their performances evaluated in 10-Fold Stratified Cross-validation.

ii. Results

The evaluation scheme is consistent with the previously described one. The model accuracies and p-values against permuted labels classifications are reported in table 2 and visualized in Figures 5 and 6.

Classes	Best Model	Accuracy	P-Value
Galantamine/Placebo	Laplacian + ADB	0.662	0.059
Amisulpride/Placebo	Laplacian + ADB	0.680	0.090
Levodopa/Placebo	Laplacian + ADB	0.662	**0.047**
Biperiden/Placebo	Laplacian + ADB	0.699	**0.026**
Biperiden/Galantamine	Laplacian + RF	0.604	0.203
Amisulpride/Levodopa	Laplacian + ADB	0.520	0.310
All Drugs	Laplacian + ADB	0.342	**0.013**

Table 2: *Results Classification Laplacian Embedding*

The results show that a laplacian-eigenmaps-based technique achieves an overall level of accuracy that is inferior to the one examined in the previous section. However, it demonstrates an increased robustness as the models achieves significant p-values in the same number of tasks as the previous technique, and it manages to obtain lower p-values in most of the other tasks. Although not being a fully-interpretable modeling approach, laplacian eigenmaps ensure an higher-interpretability level compared to the previous technique. Indeed, the low-dimensional latent space obtained with this method could reveal clusters or groups of similar EEG patterns.

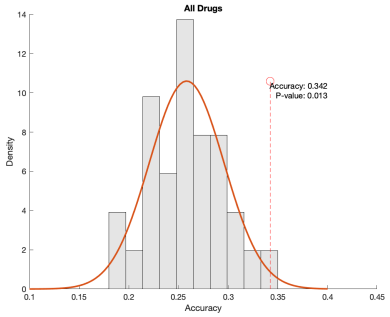

Figure 5: *Laplacian Embedding All Drugs Classification Results. Accuracies of 30 label-permuted trained models are shown as grey bars and the accuracy of the chosen model is depicted with a red-dotted line. A gaussian fit is overlaid to the histogram in red and the relative p-value can be found in the legend.*

[2] The optimal hyper-parameters set for each model can be found at Gitlab Repository

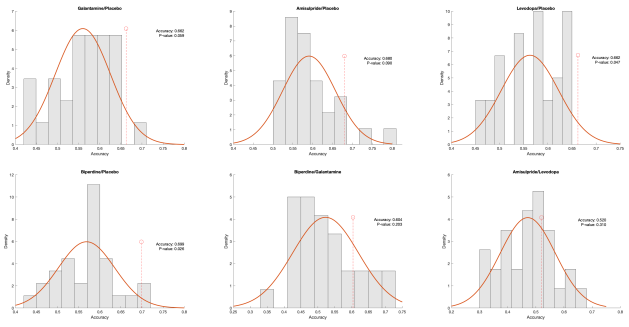

Figure 6: *Laplacian Embedding Binary Classifications Results. Accuracies of 30 label-permuted trained models are shown as grey bars and the accuracy of the chosen model is depicted with a red-dotted line. A gaussian fit is overlaid to the histogram in red and the relative p-value can be found in the legend.*

VII. Dynamic Causal Modeling (DCM)

We attempted to model the EEG data with a DCM and use the estimated parameter posterior means as features of a classifier that predicts drug labels.

The main motivation for this approach was that previous work successfully classified muscarinic agonist and antagonist drugs from epidural EEG data in a MMN paradigm in rats using this approach (Schöbi et al., 2021).

Furthermore, it is hypothesized that acetylcholine and Dopamine modulate NMDA receptor function (Stephan, Baldeweg, & Friston, 2006) and that NMDA signaling is mainly used in hierarchically descending cortical connections (Self, Kooijmans, Supèr, Lamme, & Roelfsema, 2012). Hence, one can hypothesize that acetylcholine and Dopamine agonistic and antagonistic drugs modulate the connection strength of descending cortical connections.

With a DCM it is possible to infer the connection strength of descending cortical connections from EEG recordings (Pereira et al., 2021). Hence, it might be possible that descending connection strength parameters estimated with a DCM are predictive of acetylcholine and Dopamine agonistic and antagonistic drugs.

To test this hypothesis, we designed multiple DCMs in SPM12 and fitted the pre-processed EEG of all subjects with each DCM. We then trained simple classifiers on the pruned estimated posterior mean parameters of the DCM to predict the drug labels with 10fold cross-validation.

i. Architectures

We explored the prediction performance (cross-validated classification accuracy) of three different mean trial definitions (as defined and motivate in section III), three cortical-column models (ERP: 3-population convolution-based neural mass model, CMC: 4-population convolution-based neural mass model, NMDA: 3-population conductance-based neural mass model that includes the NMDA receptor), three connection models (see Figure 7) and two simple classifiers (RF and SVM).

The reason we choose exactly those cortical-column models was as follows. The ERP model is the simplest model, CMC is the most widely used in the literature and NMDA models the NMDA receptor which is modulated by the drugs that was administered to the subjects (Stephan et al., 2006). The connection models were chosen because they have been used before to model EEG in an MMN paradigm (Garrido et al., 2008; Ranlund et al., 2016; David et al., 2006). In the case of the David2006 model only for simulated data. We added this model because it includes the orbitofrontal cortex (OF) and it was shown that MMN responded are significantly different on electrodes near the OF between subject that received either amisulpride or Biperiden (Weber et al., 2022).

The remaining DCM parameters were set as follows: normal sized template head model with 'EEG BEM' inversion, Tdcm=[0,452], Nmodes=8, h=1, onset=60 (for all other prior parameters mean and all prior variances the default values were used). The parameters of the classifiers were set to equal values for all different DCM configurations.

We pruned the extracted DCM posterior parameter means with an ANOVA test. We only selected parameters for which the null-hypothesis of an ANOVA of the parameters group according to group

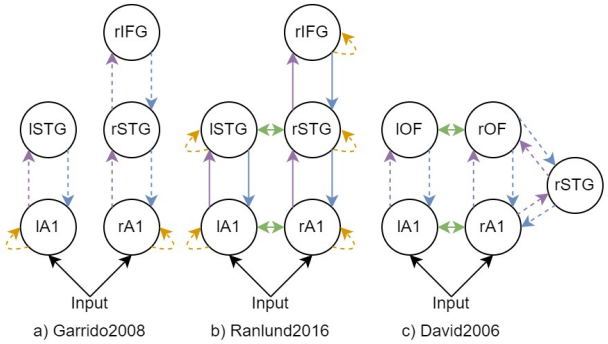

Figure 7: *The three DCM connection models. A1: primary auditory cortex; STG: superior temporal gyrus; IFG: inferior frontal gyrus; OF: orbitofrontal cortex; l: left hemisphere; r: right hemisphere; purple arrows: forward connections (c.), blue arrows, backward c., green arrows: lateral c., yellow arrows: intrinsic c.; dashed arrows indicate c. that are modulated between conditions. a) model from (Garrido et al., 2008) b) model from (Ranlund et al., 2016) c) model from (David et al., 2006).*

ii. Results

We evaluated the cross-validated drug label prediction accuracy of most of the $3 * 3 * 3 * 2 = 54$ combinations of mean trial definitions (StdVsDev/StdVsDevStb/ExtStdVsDev), cortical column models (ERP/CMC/NMDA), connection models (Garrido2008/Ranlund2016/David2006) and classifiers

(RF/SVM). We don't report results for the 8 models StdVsDevStb/ExtStdVsDev + CMC/NMDA + David2006 + RF/SVM because the model inversion was not finished on time for the report deadline. We report results for the DCM model with the highest accuracy for each of the classification tasks in Table 1 in Table 3. In addition to the accuracy, we report mean and standard deviation of a fitted Gaussian; and p-value of a permutation test as in Section ii to show if the reported accuracies are significantly better than chance.

We reached classification accuracies that are significantly better than chance (permutation test p-value $p_{PT} < 0.05$ for all binary classification task but not for the 5 class task (cf. Table 3). There was no model that performed well at all classification task. The model with the highest cumulative accuracy (summed accuracy of all 7 classification task) was StdVsDevStb+CMC+Garrido2008+RF. It achieved significant accuracies for all three binary classification tasks that involve Biperiden, Placebo and Galantamine (accuracies: Galantamine/Placebo = 0.714, Biperiden/Placebo = 0.712, Biperiden/Galantamine = 0.729). We depict the mean and standard error of mean (SEM) over treatment groups of posterior mean estimates of this model in Figure 8) for parameters for which the null-hypotheses of an ANOVA test is rejected. We find that all parameters that pass the ANOVA test are extrinsic connection parameters. Six out of seven parameters are extrinsic forward connections. This may hint be a hint that the acetylcholine modulating drugs Biperiden and Galantamine mainly impact forward connections, which is contradictory to our initial hypothesis.

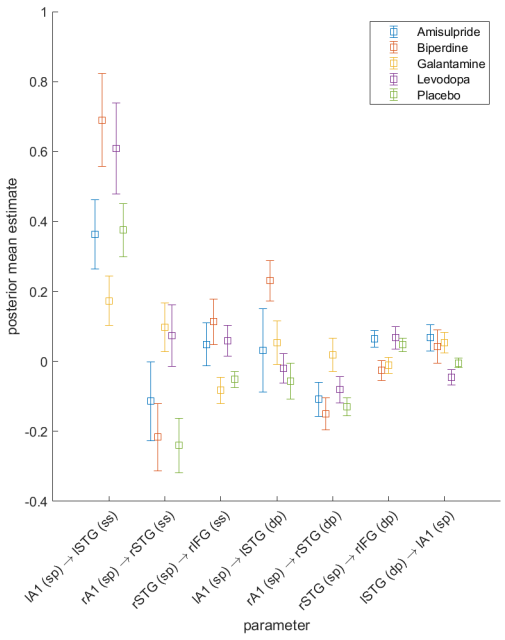

Figure 8: *Posterior estimates of parameter means with ANOVA p-value smaller than 0.05 for all subjects for the DCM model StdVsDevStb+CMC+Garrido2008. Squares depict means over subjects per treatment group. Error bars depict SEM. All depicted parameter are extrinsic connection parameters. ss: spiny stellate neurons, sp: superficial pyramidal neurons, dp: deep pyramidal neurons.*

VIII. NeuralHGF: Novel method for inversion and simulation of Hierarchical Gaussian Filter for EEG Data

In this section we propose a novel approach for performing inversion of Gaussian Hierarchical Models using Neural Networks. We first review the theory behind variational inference and Hierarchical Gaussian Filter (Mathys et al., 2014). We then look at Variational Autoencoders (Kingma & Welling, 2022), a popular method for generative modelling using Deep Neural Networks. Then we propose our method, building on the theory for Variational Autoencoders and Hierarchical Gaussian Filter. Then we move on to the specific problem of EEG data generative modelling and analyse training performance, network parameters. Finally we use the generative embedding from our model to perform classification on the dataset.

i. Variational Inference and Hierarchical Gaussian filter

Hierarchical Gaussian Filter (Mathys et al., 2014) is a popular technique used in generative modelling and inversion of behavioural data. It assumes that the behavioural data is generated by Gaussian random walks in latent space followed by a response model. It is possible to build a hierarchy of latent spaces, where a variable in every level (apart from the lowest level) performs a Gaussian random walk and is influenced by the distribution of the random variable in the next (higher) level. In this project, we will only be looking at 3-level Hierarchical Gaussian models. Assuming we have some observations regarding behavioural data $X^{(i)}, i \in \{1, 2...T\}$, where T is number of timesteps. We assume 2 levels above $X^{(i)}$ as $Z_1^{(i)}, Z_2^{(i)}$. The equations governing the dynamics for each of the above variables is given by:

$$Z_2^{(i)} \sim \mathcal{N}(Z_2^{(i-1)}, \nu)$$
$$Z_1^{(i)} \sim \mathcal{N}(Z_1^{(i-1)}, f_1(Z_2^{(i-1)})$$
$$X^{(i)} = g(Z_1^{(i)})$$

Here g is the response function. Common formulations for g are the step response, sigmoid response in case of behavioural data. Such a formulation allow us to compute the perceptual uncertainty and volatility and can be helpful in understanding cognitive mechanisms. Note that $X^{(i)}$ are the measured responses, which can be eye movements, behavioural data or EEG data. However, in case of multi-sensor EEG data, the optimal choice of the response model may not be straightforward, and it may need modifications in the above HGF model to allow applicability in our project.

Classification Task	Best Model	Accuracy	μ_{PT}	σ_{PT}	p_{PT}
Galantamine/Placebo	StdVsDevStb+CMC+Garrido2008+RF	0.714	0.658	0.008	**<0.001**
Amisulpride/Placebo	StdVsDev+ERP+David2006+RF	0.707	0.679	0.007	**<0.001**
Levodopa/Placebo	StdVsDev+ERP+Garrido2008+RF	0.701	0.659	0.011	**<0.001**
Biperiden/Placebo	StdVsDev+NMDA+David2006+SVM	0.753	0.696	0.006	**<0.001**
Biperiden/Galantamine	StdVsDevStb+CMC+Garrido2008+RF	0.729	0.499	0.066	**<0.001**
Amisulpride/Levodopa	StdVsDev+NMDA+David2006+RF	0.680	0.522	0.080	**0.024**
All Drugs	ExtStdVsDev+ERP+Ranlund2016+RF	0.362	0.342	0.015	0.083

Table 3: *Classification accuracies for DCM model with the best accuracy for each classification task. μ_{PT} and σ_{PT} are the mean and standard deviation of the fitted Gaussian in the permutation test. p_{PT} is the p-value of the permutation test (bold if $p_{PT} < 0.05$)*

ii. Variational Autoencoder

We now briefly discuss the theory behind Variational Autoencoders, a popular method for generative modelling using Deep Neural Networks (DNNs). This discussion will lead us towards the motivation and theory for Neural-HGF (discussed in next section). Assuming a dataset of input data $\boldsymbol{X} \in \mathcal{X}$. We wish to learn the distribution of the data $\mathcal{X}$. The dataset can be of images, sentences or other high dimensional data, hence hand-designing the distribution is not feasible. To learn the distribution over the data, we assume that the data is generated from an underlying latent variable $\boldsymbol{Z}$ in an abstract low dimensional space. We assume the prior distribution for $\boldsymbol{Z}$ to be standard normal distribution. The distribution of $\boldsymbol{X}$ given the value of the latent variable $\boldsymbol{Z}$ is parametrized by θ (learnable) and given as $p_\theta(\boldsymbol{X}|\boldsymbol{Z})$. Then, using sum rule and product rule, the likelihood for a datapoint $\boldsymbol{X} \in \mathcal{X}$ is given as:

$$p_\theta(\boldsymbol{X}) = \int p_\theta(\boldsymbol{X}|\boldsymbol{Z})p(\boldsymbol{Z})d\boldsymbol{z}$$

The posterior distribution for latent variable (also called the embedding) given a datapoint $\boldsymbol{Z}$ represents the belief about the value of the underlying latent variable being the cause for generating $\boldsymbol{X}$ and is given as $p_\phi(\boldsymbol{Z}|\boldsymbol{X})$, parametrized by ϕ. To learn the latent space embedding we need to find the optimal value of ϕ and to produce the distribution for the dataset we sample from underlying latent space and use the optimal value of θ to sample from $p_\theta(\boldsymbol{X}|\boldsymbol{Z})$. To learn the posterior distribution over latent space variables, we use a neural network parametrized by ϕ which is called the encoder. To learn the likelihood of data given a latent space observation, we use a neural network architecture parametrized by θ called the decoder. The entire setup is called Variational Autoencoder and the networks are learnt using the ELBO loss(Kingma & Welling, 2022). Importantly, Variational Autoencoders are a class of unsupervised learning models which do not require labels for input data, since the task is to only learn the underlying distribution of input data.

iii. Proposal for Hierarchical Gaussian filtering using Neural Networks

Immediately we can see similarities between Variational Autoencoders and Hierarchical Gaussian Filters. The reason for similarity is obvious: both are techniques of variational inference using a latent space of Gaussian random Variables. However, the setup of Variational autoencoders allows to approach the problem of finding the optimal response function in HGFs in an automatic way. The encoder in the VAE is performing model inversion and estimating posterior distribution over latent space variables given evoked responses, while the decoder is performing response simulation given the latent space variables. But, since encoder and decoder are neural network architectures whose parameters can be learnt, this means that the unknown optimal response model in the HGF can also be learnt. This is just a paraphrasing of the statement that Neural Networks are function approximators which can be used to learn the response models for our multisensor EEG data. However, there are two important steps to solve before applying the Variational Autoencoder setup to perform HGF inversion.

- The vanilla implementation of VAEs only considers a single level of latent variables. This can be extended to have hierarchical VAEs(Sønderby, Raiko, Maaløe, Sønderby, & Winther, 2016), which allows modeling volatilities like in HGF.
- Typically we would use fully connected layers in the higher levels of encoder and decoder. As a result, even if latent space variables are Gaussian random variables, they lie in an abstract space and the connection to observed time series response data is not straightforward. Such a formulation removes the interpretation of the arrow of time and it is unclear whether a particular latent variable corresponds to an observation at a particular timepoint or whether it represents some general feature about the entire time series. This is unlike the HGF where we know the estimates for latent variable parameters at each point in time, which can be useful when connecting the Hierarchical model estimates to Mismatch Negativity. But if we design the encoder and decoder neural networks to be either 1-D convolution layers or Recurrent units, we can restrict the latent space variables to also be a time series and the decoder can be made to only perform the response function approximation without warping the latent space. As a result of this, we can have latent space variables that are themselves time series and connected to the observed time series responses in a restricted and interpretable manner.

Hence, by carefully designing the neural network architecture for encoder/decoder and using hierarchical latent space variables, we can use the function approximation properties of neural networks to learn the optimal response function for the HGF in an automatic way. To the best of our knowledge, this method has not been used in the context of EEG sensor data modeling, thus representing a novel approach, which we propose to call NeuralHGF. This approach has a direct biological analogue: the brain as a learned neural network performing variational inference via HGF and the synaptic connections computing the response function.

iv. Design of network and training algorithm

The overview of the architecture is presented in Fig 9. The training procedure is summarised as follows:

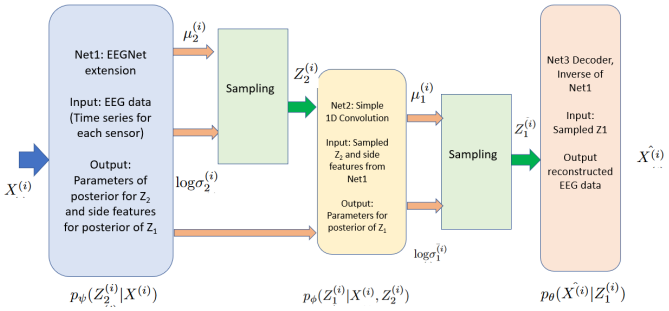

Figure 9: *Network overview for NeuralHGF*

- Given: A minibatch of multivariate time series data (here multivariate means multiple EEG sensors). Each datapoint is the average of EEG over trials for one candidate in the drug trial. Standards and deviants are averaged separately and counted as separate datapoints.
- Pass the input data through Net1. Net1 computes $p_\psi(Z_2^{(i)}|X^{(i)})$, the posterior distribution for the latent variable time series $Z_2^{(i)}$. Note that $Z_2^{(i)}$ is a time series of shape $1 \times T$, where T=139 is timepoints in data. Net1 is similar to EEGNet and by using time domain convolutions we preserve interpretability of data.
- Sample $Z_2^{(i)}$ and pass it to Net2. Net2 computes $p_\phi(Z_1^{(i)}|X^{(i)}, Z_2^{(i)})$, posterior distribution for latent variable time series $Z_1^{(i)}$. Note that $Z_1^{(i)}$ is a time series of shape $1 \times T$, where T=139 is timepoints in data. Net2 is a single 1-D convolution layer.
- Sample $Z_1^{(i)}$. Pass it through Net3 which reconstructs EEG data (decoder).
- Update parameters using ELBO loss $\mathcal{F} = E_{p_\psi(Z_2|X)p_\phi(Z_1|X,Z_2)}(p(X|Z_1)) - E_{p_\psi(Z_2|X)}(KL(p_\phi(Z_1|X,Z_2)||p(Z_1|Z_2))) - KL(p_\psi(Z_2|X)||p(Z_2))$.
- For reconstruction loss we use L1 loss. The prior for Z_2 is $p(Z_2) \sim \mathcal{N}(0, I)$, prior for Z_1 is $p(Z_1|Z_2) \sim \mathcal{N}(0, e^{Z_2})$.
- We use two latent variable time series of dimension 1x139. Generative embedding is of dimension 4x139 (represents means and variances for both the latent variables) and represents a reduction to 6 percent of data. When doing classification, we pass standards EEG data, extract generative embedding, then we pass deviants EEG data and extract generative embedding. These two are combined and fed to the classifier as features.

v. Training and validation results on data generation

In Fig 10 we provide the plots for evolution of reconstruction loss and KL-Divergence loss as mentioned in the ELBO function. In Fig 11 we provide some examples of reconstructed data by the NeuralHGF model. One can see that the variational autoencoder is capable of compressing and reconstructing the data fairly well.

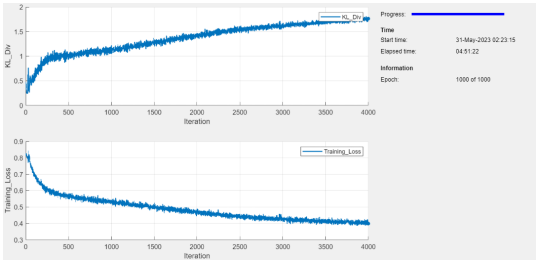

Figure 10: *Training plots for Neural HGF*

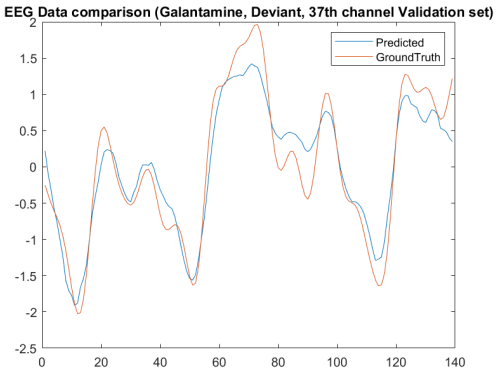

Figure 11: *Comparison of Ground truth and predictions for selected EEG data from Validation set*

vi. Results for Classification with Embedding

Now we move on to performing classification on the data. We use the NeuralHGF model to generate embeddings. We then train another Machine Learning model to perform classification based on these generated embeddings. We perform 4-Fold Stratified Cross

Classification Task	Best Model	Accuracy
Galantamine/Placebo	NHGF+RF	0.667
Amisulpride/Placebo	NHGF+RF	0.680
Levodopa/Placebo	NHGF+SVM	0.662
Biperiden/Placebo	NHGF+RF	0.699
Biperiden/Galantamine	NHGF+KNN	0.618
Levodopa/Amisulpride	NHGF+RF	0.583
All drugs	NHGF+SVM	0.3422

Table 4: *Classification accuracies using NeuralHGF generative embeddings*

validation and report the accuracies for different classification task in Table 4. We use the same four classification models as given in Section 5.

vii. Analysis and Discussion

Weaknesses:

- Classification accuracy is better than random, but compared to other generative embeddings is worse.
- We had mentioned that we are using Neural Networks to learn the optimal response function of HGF and connection between hierarchical levels. In this project, even though we have trained the neural network to perform HGF inversion and simulation, a study of the resulting response function from latent space to EEG sensor data is lacking: shape, characteristics and biological interpretation.
- In this project, we used time domain convolutions in the encoder and decoder neural network to preserve the interpretation of latent space. A better approach would have been to use LSTMs(Hochreiter & Schmidhuber, 1997) with multivariate setting, where the multivariate setting corresponds to different sensor channels, which would give greater interpretability and model capacity.

Strengths:

- Novel implementation to perform HGF inversion and response simulation. This formulation can be used to remove the response function engineering and selection during HGF and is especially useful in complex data. This technique was used here for EEG sensor data, but can be extended to other problems as well: merging the interpretation capabilities of HGF with function approximation power of neural networks. To the best of our knowledge, this is the first work in this regard.
- Classification scores are still better than random, and there is scope for improvement by changing the neural network architecture to have LSTMs.

IX. Convolutional Recurrent Attention Model (CRAM) (Extra)

[3] This modeling technique is inspired by (Tao et al., 2020), (Zhang, Yao, Chen, & Monaghan, 2019), (Kwak, Song, & Kim, 2021). All of the three papers develop a deep neural network (DNN) featuring both convolutional and recurrent layers, as well as attention and self-attention layers to uncover the spatio-temporal dependencies contained in the EEG data. Starting from these examples, we developed a custom DNN architecture to perform the multi-class predictions in the *All Drugs* task.

i. Architecture

The pipeline followed by our model and its architecture can be visualized in Figure 12. In particular, standard and deviant trials are processed via the same layers, but separately throughout the architecture to be merged via an attention layer right before performing predictions.

- Standard and Deviants are defined according to the first approach described in the Pre-Processing section to obtain 106 standard trials and 119 deviant trials.
- Standard and deviant trials are averaged to obtain two data sets of shape 149 subjects, 63 channels, 139 samples. From here on, we will discuss only one branch of the architecture as it is equivalent to the other one.
- The data set is passed through a Channel-Wise Attention structure made of the following layers: Mean-Pooling, two Fully-Connected (FC) with Tanh activation and Softmax.
- The Attention-weighted data are fed to a Convolutional layer with Exponential Linear Unit (ELU) activation then Max-Pooled, Flattened and Dropout. This layer allows to extract the spatial dependencies in the data.
- The data set is then fed to a bi-layered LSTM followed by attention. This layer allows to extract the temporal dependencies in the data.
- The two branches of the architecture are merged again via concatenation.
- Then, attention on the concatenated data is performed.
- Finally, the predictions on the five classes are obtained via a FC layer with tanh activation and softmax layer.

ii. Implementation & Training

The model neurons per layer and hyper-parameters have been optimized in a stratified 10-fold cross-validation setting. In particular:

- Optimizer: Adam

[3] This modeling technique is defined "Extra" because it has been developed in Python and hence cannot be considered towards the grading according to the course project rules. We developed it out of curiosity and, although lacking of a generative embedding and proper interpretability, the results are promising.

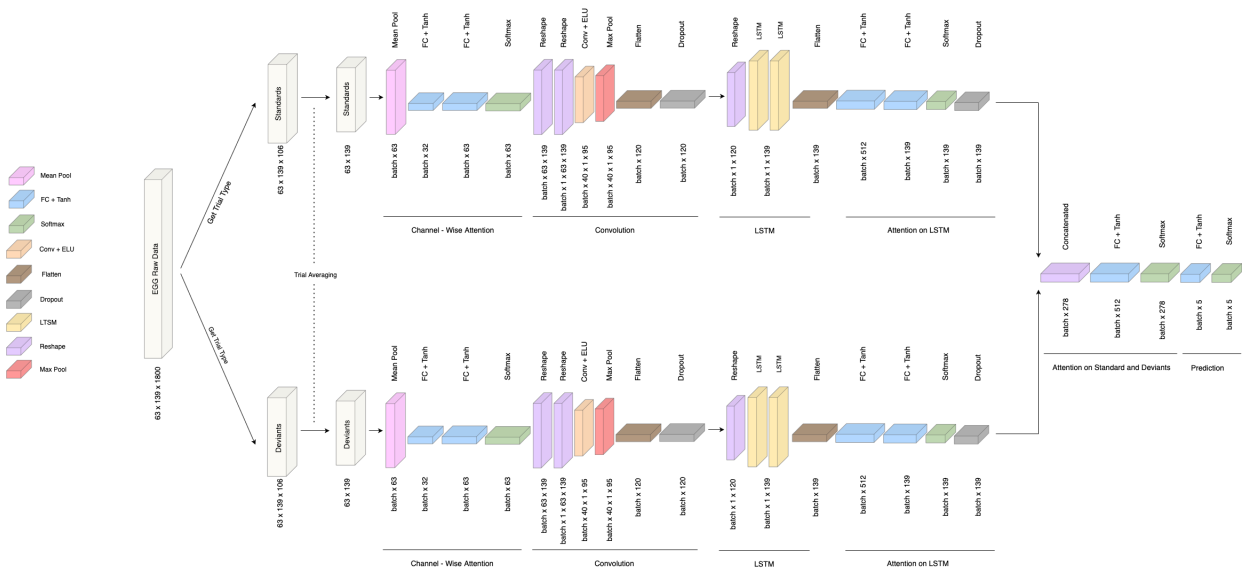

Figure 12: *CRAM Architecture Diagram*

- Batch size: 11
- Learning rate: 0.0001

The training has been performed for 200 Epochs on NVIDIA T4 Tensor core GPU for a training time of approximately 4-5 hours. We achieved a 10-fold cross-validated accuracy of 0.49 on the *All Drugs* classification task. Further analysis on the binary classification tasks and permuted-labels model training comparisons have not been performed.

X. Results & Conclusion

We explored a large variety of different modeling techniques. We achieved better classification accuracy than chance for all 6 binary classification tasks (Table 5). However, we were not able to beat the baseline model for each binary classification task. This may indicate a limitation of the explored models for the modeling of the provided data. One reason might be the small number of samples and the possibly large inter-subject variability. We reckon that the model would have performed better if all drugs would have been administered to all subjects as in the experimental paradigm in (Schöbi et al., 2021). For the 5 class classification we only reach an accuracy that is significantly better than chance with our extra method CRAM. However, this method was not fully evaluated and is is a data-based approach that has not interpretability. Among generative modelling approaches, we achieved the best results with DCM and the best DCM model shows good discriminability for the acetylcholine modulating drugs Biperiden and Galantamine, but not for the Dopamine modulating drugs. We also proposed and implemented a novel approach called NeuralHGF for combining interpretability of HGF with approximation power of neural networks. While this approach may not have given the best results, there is a lot of scope for improvement of architecture and extension to other problem modalities.

Classification Task	No Embedding	Laplacian	DCM	NeuralHGF	CRAM
Galantamine/Placebo	0.636	0.662	**0.714**	0.667	-
Amisulpride/Placebo	0.680	0.680	**0.707**	0.680	-
Levodopa/Placebo	0.675	0.662	**0.701**	0.662	-
Biperiden/Placebo	**0.767**	0.699	0.753	0.699	-
Biperiden/Galantamine	**0.729**	0.604	**0.729**	0.618	-
Amisulpride/Levodopa	0.540	0.520	**0.680**	0.583	-
All Drugs	0.363	0.342	0.362	0.342	**0.49**

Table 5: *Final Results Table. Accuracies for all classification tasks and all classification models. The best model for each task is highlighted in bold. This table summuraizes the results from Table 1 - 4.*

References

Belkin, M., & Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, *15*(6), 1373–1396.

David, O., Kiebel, S. J., Harrison, L. M., Mattout, J., Kilner, J. M., & Friston, K. J. (2006). Dynamic causal modeling of evoked responses in eeg and meg. *NeuroImage*, *30*(4), 1255–1272.

Garrido, M. I., Friston, K. J., Kiebel, S. J., Stephan, K. E., Baldeweg, T., & Kilner, J. M. (2008). The functional anatomy of the mmn: a dcm study of the roving paradigm. *Neuroimage*, *42*(2), 936–944.

Gramfort, A., & Clerc, M. (2007). Low dimensional representations of meg/eeg data using laplacian eigenmaps. In *2007 joint meeting of the 6th international symposium on noninvasive functional source imaging of the brain and heart and the international conference on functional biomedical imaging* (pp. 169–172).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Kingma, D. P., & Welling, M. (2022). *Auto-encoding variational bayes.*

Kwak, Y., Song, W.-J., & Kim, S.-E. (2021). Deep feature normalization using rest state eeg signals for brain-computer interface. In *2021 international conference on electronics, information, and communication (iceic)* (pp. 1–3).

Mathys, C. D., Lomakina, E. I., Daunizeau, J., Iglesias, S., Brodersen, K. H., Friston, K. J., & Stephan, K. E. (2014). Uncertainty in perception and the hierarchical gaussian filter. *Frontiers in Human*

Neuroscience, *8*. Retrieved from https://www.frontiersin.org/articles/10.3389/fnhum.2014.00825 doi: 10.3389/fnhum.2014.00825

Pereira, I., Frässle, S., Heinzle, J., Schöbi, D., Do, C. T., Gruber, M., & Stephan, K. E. (2021). Conductance-based dynamic causal modeling: A mathematical review of its application to cross-power spectral densities. *NeuroImage*, *245*, 118662.

Ranlund, S., Adams, R. A., Díez, Á., Constante, M., Dutt, A., Hall, M.-H., ... others (2016). Impaired prefrontal synaptic gain in people with psychosis and their relatives during the mismatch negativity. *Human brain mapping*, *37*(1), 351–365.

Schöbi, D., Homberg, F., Frässle, S., Endepols, H., Moran, R. J., Friston, K. J., ... Stephan, K. E. (2021). Model-based prediction of muscarinic receptor function from auditory mismatch negativity responses. *NeuroImage*, *237*, 118096.

Self, M. W., Kooijmans, R. N., Supèr, H., Lamme, V. A., & Roelfsema, P. R. (2012). Different glutamate receptors convey feedforward and recurrent processing in macaque v1. *Proceedings of the National Academy of Sciences*, *109*(27), 11031–11036.

Stephan, K. E., Baldeweg, T., & Friston, K. J. (2006). Synaptic plasticity and dysconnection in schizophrenia. *Biological psychiatry*, *59*(10), 929–939.

Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., & Winther, O. (2016). *Ladder variational autoencoders.*

Tao, W., Li, C., Song, R., Cheng, J., Liu, Y., Wan, F., & Chen, X. (2020). Eeg-based emotion recognition via channel-wise attention and self attention. *IEEE Transactions on Affective Computing*.

Weber, L. A., Tomiello, S., Schöbi, D., Wellstein, K. V., Mueller, D., Iglesias, S., & Stephan, K. E. (2022). Auditory mismatch responses are differentially sensitive to changes in muscarinic acetylcholine versus dopamine receptor function. *Elife*, *11*, e74835.

Zhang, D., Yao, L., Chen, K., & Monaghan, J. (2019). A convolutional recurrent attention model for subject-independent eeg signal analysis. *IEEE Signal Processing Letters*, *26*(5), 715–719.