EEG DGD space-time literature and code [June 12]

This text continues the results of May 30. The conclusion was: to build a DGD model with the linear decoder (and) as a state (process) space model.

The question to keep

The data analysis discussion on July 6 raises the central question: how can we merge participants into one joined dataset? To merge, the participants must **belong to the same general population.** For now, only a few participants can be merged. A participant from the rest requires a personal model, which results in the model ensemble. But due to the insufficient sample size, the robustness of such a model is below desired. Additional tests on the general population are needed.

The ressourses for DGD

1. Deep Discriminative Direct Decoders for High-dimensional Time-series Analysis // ArXiv 2022, https://arxiv.org/abs/2205.10947 code https://github.com/MrRezaeiUofT/Deep_Direct_Discriminative_Decoder-D4-
2. Bayesian Decoder Models with a Discriminative Observation Process // BioRxiv, 2020. https://www.biorxiv.org/content/10.1101/2020.07.11.198564v3
3. Analysis of Distributed Neural Synchrony through State-Space Coherence Analysis // BioRxiv, 2020. https://doi.org/10.1101/2020.07.13.199034
4. Inferring Cognitive State Underlying Conflict Choices in Verbal Stroop Task Using Heterogeneous Input Discriminative-Generative Decoder Model // BioRxiv, 2022 https://doi.org/10.1101/2022.11.28.518256
5. Closed loop enhancement and neural decoding of human cognitive control 2020. https://doi.org/10.1101/2020.04.24.059964 code https://github.com/TRANSFORM-DBS/Encoder-Decoder-Paper https://github.com/Eden-Kramer-Lab/COMPASS
6. Continuous Prediction of Cognitive State Using A Marked-Point Process Modeling Framework // Pubmed, 2019. https://doi.org/10.1109/EMBC.2019.8856681 code https://github.com/Eden-Kramer-Lab/StateSpaceMarkedPointProcess
7. Real-Time Point Process Filter for Multidimensional Decoding Problems Using Mixture Models // J. Neuroscience Methods https://doi.org/10.1016/j.jneumeth.2020.109006 code https://github.com/Eden-Kramer-Lab/GMM_PointProcess

For discussion

1. What is the basic DGD model in code?
2. What process defines the basic DGD model in code?

Discussed: the simplest models to test the DGD in the two new papers:

1. Decoding Hidden Cognitive States From Behavior and Physiology Using a Bayesian Approach // Neural Computation, 2019. doi:10.1162/neco_a_01196
   instead of 1. run 2.
2. Deep Direct Discriminative Decoders for High-dimensional Time-series Data Analysis (not published yet, see https://doi.org/10.48550/arXiv.2205.10947)

The state space model resources

The discussion shows the lack of information on the particular SSM model and toolboxes. But the spatial part of the time series remains undoubtful. So the extension of this list, devoted to the spatial time SSM models, is requested.

1. The Annotated S4. Efficiently Modeling Long Sequences with Structured State Spaces https://github.com/vadim-vic/annotated-s4
2. State Space Models with Generalized Orthogonal Basis Projections https://doi.org/10.48550/arXiv.2206.12037
3. Code https://github.com/HazyResearch/state-spaces Structured State Spaces for Sequence Modeling
4. See also LSSL, SaShiMi, DSS, HTTYH, S4D, and S4ND from Hazy Research. Models: https://github.com/HazyResearch/state-spaces/tree/main/models
5. Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review https://github.com/pyRiemann/pyRiemann
6. Spatial time SSM with Graph Laplacian: Longitudinal predictive modeling of tau progression along the structural connectome https://doi.org/10.1016/j.neuroimage.2021.118126
7. Spatial time SSM with GCANs based on symmetry group transformations using geometric alg https://arxiv.org/abs/2302.06594

Appendix

The cross-validation procedure (discussed)

The procedure supposes the train, test, and validation splitting. The new dataset will be fitted by a selected fixed model with parameter (fine) tuning:

1) preprocess each user's trial individually or align [standardization procedure over the whole session] with the other user data,
2) [NEW Keep several (random? which?) users completely out of training and model selection process]
3) keep [NEW random/compact chunks] a fixed proportion of trials for each user in the non-trained set (user balance for leave-K-each_user_trials-out procedure),
4) shuffle the users and their trials randomly, make the training procedure,
5) (OR) split the 3) for train and testing to select a model or features,
6) for the 4) case, make an average for test trials splitting the model selection,
7) estimate quality on leave-K-each_user_trials-out set,
8) repeat 3) and 6) for a fixed model CV Fold times.

The cross-validation settings

1. The proportion for train-test-validation: 70:20
2. The proportion (number of trials per user) for leave-K-user-out: 10
3. The number of CV Fold iterations for averaging: 10
4. Number of totally leaved-out users: 10/100, 10% (unfeasible for now, see the note in the beginning)
5. [We need to report only sensitivity and specificity] For general statistics, refer to the table

Table 6. The class balance, each user per task

	Task	Classes	1034	1037	1045	1158	1363	1368	1385	2038	6639	7974	7977	7980	1327
Class 1	Encoding	Large Small	41	50	60	62	43	45	59	38	50	48	61	44	56
	Lexical	Old New	48	51	55	49	57	46	51	48	51	46	56	52	50
	Recognition	Old New	55	50	45	53	54	41	37	44	52	45	55	57	29
	Recognition	Old Non	48	51	55	49	57	46	51	48	51	46	56	52	50
	Recognition	New Non	40	49	50	49	46	42	49	38	48	42	49	53	34
Class 2	Encoding	Large Small	39	47	45	47	45	38	51	38	47	49	55	40	52
	Lexical	Old New	40	49	50	49	46	42	49	38	48	42	49	53	34
	Recognition	Old New	53	46	44	51	55	41	37	46	48	45	54	49	29
	Recognition	Old Non	50	55	56	49	49	43	53	48	50	43	55	49	45
	Recognition	New Non	50	55	56	49	49	43	53	48	50	43	55	49	45

Extract from the slides on the Genetic feature selection algorithm

The number of features exceeds the size of the sample set. Thus, we expect the robustness of models to be low; see https://doi.org/10.1016/j.chemolab.2015.01.018. To select the most informative features over 128 electrodes and 4 peaks (512 in total), we use the genetic algorithm optimization discussed in https://doi.org/10.1016/j.eswa.2017.01.048 .

Since the number of electrodes (four peaks give 512 features for 128 electrodes) is comparable to the number of objects in the dataset (64,...,907) and the EEG ERP electrode signals are highly correlated, the logistic regression with elastic net regularization, used for the feature selection previously, delivers unstable results. As a result, we explore other feature selection techniques with a promise of being robust; these techniques are: 1) a discrete genetic feature selection algorithm, and 2) a quadratic programming feature selection algorithm. Here, we show the genetic algorithm results.

The concept of genetic optimization is to create a population of models and randomly exchange their features with the accuracy evaluation, selecting the best models after every new population offspring. According to Holland's schema theorem, the number of informative features will exponentially grow in the best models (https://dynamics.org/Altenberg/FILES/LeeSTPT.pdf)

The genetic algorithm selects a limited number of features, a subset from a large feature set (for this slide, it is 16 out of 512). First, a population of models exists: randomly selected subsets of features (for this slide, 14 models in the population). Second, for each member of the population, a mom and a random dad were selected. They exchange a random number of features (analog is the chromosome cross-over). A kid carries the same number of features. Some features of each kid in the offspring are randomly replaced with new ones (analog is the mutation). The quality of each kid in the offspring is evaluated (tune parameters using the test data according to the likelihood and evaluate the quality using the test data according to AUC; cross-validate for each model). Make the new population of the same size from the best members of the old population and the offspring (analog is fitness). Repeat the second step. The stop criterion is the given number of generated populations (10000 for this slide) or convergence in the feature occurrence. The properties of the genetic optimization algorithm in comparison to the other feature selection algorithms is comprehensively analyzed in https://doi.org/10.1016/j.eswa.2017.01.048

Minor supplementary

Another list of the basic features

1. Amplitude: The amplitude of an ERP component is the difference between the maximum and minimum values of the waveform. Amplitude features can provide information about the strength of the neural response to a specific event.
2. Latency: The latency of an ERP component is the time taken for the component to reach its peak from the onset of the stimulus. Latency features can provide information about the timing of the neural response to a specific event.
3. Frequency: The frequency of an ERP component refers to the oscillatory activity present in the EEG signal. Frequency features can provide information about the spectral characteristics of the neural response to a specific event.
4. Topography: The topography of an ERP component refers to the spatial distribution of the EEG signal over the scalp. Topography features can provide information about the brain regions involved in the neural response to a specific event.
5. Variability: The variability of an ERP component refers to the consistency of the neural response across different trials or participants. Variability features can provide information about the reliability of the neural response to a specific event.
6. Duration: The duration of an ERP component refers to the time period during which the component is present in the EEG signal. Duration features can provide information about the temporal characteristics of the neural response to a specific event.
7. Inter-trial coherence (ITC): The ITC of an ERP component refers to the phase consistency of the oscillatory activity across different trials or participants. ITC features can provide information about the phase-locking properties of the neural response to a specific event.
8. Peak-to-peak amplitude: The peak-to-peak amplitude of an ERP component refers to the difference between the maximum and minimum values of the waveform, but calculated from the preceding peak or trough to the current peak or trough. Peak-to-peak amplitude features can provide information about the relative strength of different components in the EEG signal.

Alternative possible data collections for augmentation

The HTNet results are based on large data collections. They reveal semisupervised manifolds in data and use these manifolds for robust classification.

1. The Human Connectome Project (HCP): The HCP is a large-scale project that aims to map the neural connections in the human brain using various neuroimaging modalities, including EEG. The HCP EEG dataset includes resting-state and task-related EEG recordings from over 500 participants, along with extensive demographic and behavioral data.
2. The EEG-fMRI Neurofeedback in Depression (ENLIGHT) dataset: The ENLIGHT dataset includes EEG and fMRI (functional Magnetic Resonance Imaging) recordings from patients with major depressive disorder who underwent neurofeedback training. The dataset includes over 500 hours of EEG recordings from 16 participants, along with clinical and behavioral data.
3. The EEG Database for Emotion Analysis using Physiological Signals (DEAP): The DEAP dataset includes EEG and physiological recordings from participants who watched a series of music videos designed to induce various emotional states. The dataset includes EEG recordings from 32 channels, along with self-reported emotional ratings and physiological measures.
4. The ERP Core Dataset: The ERP Core Dataset is a collection of ERP recordings from over 500 participants who completed a set of standard ERP paradigms, including the oddball and N-back tasks. The dataset includes standardized experimental procedures and analysis guidelines, making it a valuable resource for cross-study comparisons.
5. The EEG Database for Cognitive Neuroscience (BrainVision): The BrainVision database includes EEG recordings from over 600 participants, along with demographic and behavioral data. The dataset includes a variety of experimental paradigms, including visual and auditory oddball tasks, memory tasks, and language tasks.
6. The Autism Brain Imaging Data Exchange (ABIDE): The ABIDE dataset includes EEG and fMRI recordings from individuals with autism spectrum disorder (ASD) and typically developing controls. The dataset includes data from over 1,000 individuals across 17 sites, making it a valuable resource for studying the neural mechanisms underlying ASD.
7. The Cognitive Neuroscience Test Reliability and Clinical Applications for Schizophrenia (CNTRACS) dataset: The CNTRACS dataset includes EEG recordings from individuals with schizophrenia and healthy controls who completed a set of standard cognitive tasks. The dataset includes data from over 300 participants across 7 sites, along with extensive clinical and behavioral data.
8. The Montreal Archive of Sleep Studies (MASS): The MASS dataset includes EEG recordings from individuals during various stages of sleep and wakefulness. The dataset includes data from over 800 participants, along with demographic and sleep-related data.
9. The EEG Database of Normal and Abnormal Human Brain Development (NEED): The NEED database includes EEG recordings from individuals across the lifespan, from infancy to late adulthood. The dataset includes data from over 1,000 participants, making it a valuable resource for studying the neural mechanisms underlying brain development.