Contents

ICANN/PASCAL2 Challenge: MEG Mind-Reading — Overview and Results

Arto Klami[1], **Pavan Ramkumar**[2], **Seppo Virtanen**[1], **Lauri Parkkonen**[2], **Riitta Hari**[2], **Samuel Kaski**[1,3]

[1,2]Aalto University School of Science
[1]Department of Information and Computer Science
[1,3]Helsinki Institute for Information Technology HIIT
[2]Brain Research Unit, Low Temperature Laboratory
[3]University of Helsinki

Abstract

This report summarizes the modeling challenge held in conjunction with the International Conference on Artificial Neural Networks (ICANN) 2011 and sponsored by the PASCAL2 Challenge Programme. The challenge aimed at promoting awareness of the task "mind reading" or "brain decoding" based on magnetoencephalography (MEG) data. For neuroscientists, the task provides a practical tool for understanding brain process underlying perception, since any mechanism that can be used for inferring the stimulus on the basis of brain activity must be related to processing of the stimulus. For machine learners and other modelers, the challenge provides an interesting real-world application playground for solving active machine learning problems such as multi-view learning and covariate shift.

The task was to infer from one-second time windows the type of visual stimulus shown to the subject. The best brain decoders, out of the 10 submissions, reached almost 70% *accuracy in the task with mere* 23% *chance-level, proving that even a short MEG measurement can be sufficient for brain decoding tasks with a reasonable number of stimulus categories.*

1 Introduction

A grand challenge in neuroscience is to understand the neural basis of sensory and cognitive processing, even to the extent to predict brain correlates of novel stimuli. This challenge can be formulated as a decoding problem: given the brain signals, read out some information about the stimuli that generated (or modulated) them [1]. The information read out can be category specific, identity specific, or the entire stimulus itself—corresponding to the machine learning tasks of classification, identification, or regression/reconstruction. Such decoding tasks are often called brain/mind decoding, or multivariate/multivoxel pattern analysis (MVPA).

The majority of the reported brain decoding results derive from functional magnetic resonance imaging (fMRI), from attempts to decode relatively simple properties or to choose the correct alternative amongst a few choices. For example, Kamitani et al. [2] inferred the orientation of edges out of 8 possible alternatives and Formisano et al. [3] identified what (out of three vowels) and whom (out of three alternative speakers) the subject was listening to. Recent studies have shown significant progress in decoding more and more complex perceptual phenomena, resulting in successful identification of natural images [4] and the meaning of nouns [5] in setups where the set of possible alternatives is larger, in the order of tens. All of these works fall into the category of classification or identification. Miyawaki et al. [6] have studied the task of reconstruction of small binary images from local image patches decoded from brain signals, and Naselaris et al. [7] extended reconstruction tasks to natural images.

While fMRI has very high spatial resolution throughout the brain, it has poor temporal resolution and the blood oxygenation level dependent (BOLD) signal is an indirect measure of neuronal activity. Riger et al. [8] have shown that it is possible to apply decoding similarly to magnetoencephalography (MEG); they predicted on the basis of single-trial MEG signals whether the subject recognized and memorized a natural image. With MEG it will be possible to focus on shorter timescales. Of particular interest is the feasibility of brain decoding for continuous processes using e.g. speech or video stimuli. Besides attempting to decode external stimuli, MEG has also been used for decoding the direction of hand-movement [9] or reconstructing hand-movement trajectories [10]. Nevertheless, the task of brain decoding from MEG is still in its infancy.

From another point of view, the brain decoding task can be seen purely as a challenging machine learning problem. The recorded brain signals are very high-dimensional and noisy, and consequently advanced classification or regression methods are needed for solving the prediction task. This is also demonstrated in practical work, with focus on advanced Bayesian solutions [10, 11] and completely novel types of machine learning strategies, such as the zero-shot learning concept [12]. Furthermore, many of the current trends in machine learning are highly relevant for solving the brain decoding challenges: (1) the models need to handle covariate shifts (changes in the input distribution between training and test data) [13] with approaches like domain adaptation [14], (2) sparse solutions such as lasso regression [15] are likely to be effective for the high-dimensional data sources, (3) the prediction tasks should ideally combine information from multiple sources through multi-view learning, and (4) especially analysis of multiple subjects would benefit from multi-task learning methods [16].

We organized the challenge for brain decoding based on MEG signals for four primary reasons. (1) To increase the awareness of the problem amongst both machine learning researchers and neuroscientists, (2) to study the feasibility of decoding continuous visual stimulus from short periods of MEG recordings, (3) to bring up some of the relevant methodological challenges for MEG brain decoding, and (4) to provide a simple benchmark data set. The challenge was organized in co-operation with the ICANN conference since it attracts machine learning researchers with interest in modeling neural processes. The motivations are largely shared by other recent attempts of promoting visibility of brain decoding in general, such as the 1st ICPR workshop on brain decoding, organized in conjunction with the 20th International Conference on Pattern Recognition.

2 Data

2.1 Stimuli

The brain decoding task in the challenge was to recognize the type of video stimulus shown to the subject. All videos were presented without audio, and five different types of stimuli were used:

1. Artificial: Screen savers showing animated shapes or text

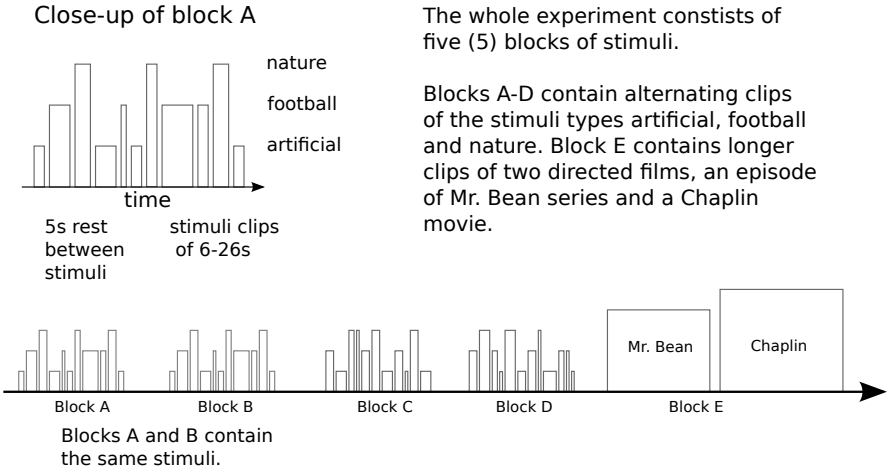

Figure 1. Illustration of the stimulus design. The subject viewed the same set of 5 blocks during two consecutive days. The first four blocks, labeled A-D, contained alternating short clips of artificial objects (animated shapes or text), football or nature documentaries, whereas the last block contained longer clips taken from a television series and a feature film. Within blocks A-D the different clips were separated by 5-s rest period showing a crosshair, and the clips lasted for 6-26 s. The two longer clips in block E, extracted from video content with a storyline, lasted for roughly 10 minutes.

2. Nature: Clips from nature documentaries, showing natural sceneries like mountains or oceans

3. Football: Clips taken from (European) football matches of Spanish La Liga

4. Mr. Bean: Clip from the episode “Mind the baby, Mr. Bean” of the Mr. Bean television series

5. Chaplin: Clip from the “Modern times” feature film

The stimuli were shown in five blocks (Figure 1). The first four blocks (A–D) contained alternating short clips of the first three stimulus types, so that each block contained a roughly equal number of clips for each stimulus type in random order. The clips lasted 6-26 s, and the different clips were separated by 5-s rest periods showing a crosshair in the center of the visual field. The first two blocks were identical, whereas blocks C and D contained different video clips.

After the four blocks described above, the subject viewed two continuous video clips containing a clear plot and storyline (clips from an episode of a television series and a feature film), each lasting roughly 10 min. These two clips were shown during the same experiment block.

2.2 Recording and preprocessing

We recorded MEG signals from one healthy 25-yrs old male who gave his written permission for releasing the data for the challenge.

MEG was acquired with a 306-channel Elekta Neuromag MEG system (Elekta Oy, Helsinki, Finland) with a basspand from DC to 330Hz and digitized at 1000 Hz. During the MEG recording, four small coils, whose locations had been digitized with respect to anatomical landmarks, were briefly energized to determine the subject's head position with respect to the MEG sensors. The continuous raw MEG data were further low-pass filtered at 50 Hz, and downsampled to 200 Hz. External interference was removed and head movements compensated for by using the signal-space-separation (SSS) method [17]. Finally, we applied piecewise mean and trend removal for each channel to compensate for very slowly varying signals that are likely to be artefacts.

Since identifying the videos would be relatively easy based on long sequences of MEG recordings, we chose to hand out only short 1-s signal epochs in random order. However, handing out only the raw measurement data would have resulted in a challenge that requires considerable expertise on MEG. In addition, it would have prevented reliable estimation of low-frequency waveforms because sharp filters could not be applied for signals as short as 1 s (200 samples). Consequently, we chose to precompute a number of features at different frequency bands. We applied a bank of 5 band-pass filters peaked at 2, 5, 10, 20, and 35Hz, and computed the envelopes of the signals at these frequencies by taking the absolute value of the Hilbert-transformed signal. The details of the filter bank are provided in Table 1.

For each sample (1-s epoch of the recording) the participants received six different data matrices, each containing 200 time points for 204 gradiometer channels of the MEG device. Those data matrices corresponded to the raw signals after the SSS preprocessing, and the envelopes at the five frequencies mentioned above.

3 Modeling problem

The modeling problem was to infer the stimulus from brain signals. Given the limited set of possible stimuli, this was a classification task: For each

Table 1. Details of the filter bank. The first column indicates the name of the filter, identified with a frequency within the band-pass area determined by the next two columns. The filters were Kaiser window FIR filters with stop bands increasing from 0.5Hz to 2Hz with increasing frequency. The order of the filters is shown in the last column.

Peak freq. (Hz)	Min freq. (Hz)	Max freq. (Hz)	Order
2	1	4	2009
5	4	7	2009
10	7	13	503
20	17	23	503
35	27	43	503

input signal the task was to infer the type of the stimulus. Consequently, the challenge was formulated as a classification problem. Given a set of labeled training examples, the task was to infer the labels for left-out test data.

For brain decoding, the generalization to new stimuli is critical. While the set of possible stimulus types needs to be limited to make inference possible, the actual stimulus content should be different for training and test samples. After all, the goal is not to recognize when the subject is watching a particular clip of a football match, but to identify the process of watching football in general. Besides generalizing to new stimuli, a brain decoding system will need to generalize over different recording sessions.

3.1 Data split

For studying the above properties, the data were split into training and test sets so that the following properties were satisfied:

- Some of the training and test instances were recorded using the same stimuli, whereas some test instances were taken from recordings of different stimuli of the same type. In total, 33% of the test samples consisted of recordings during stimuli not seen in the training phase.

- The training and test data were taken from different recording sessions. In particular, the training and test data were recorded during different days.

- A small portion of the test samples were labeled, to simulate brief train-

ing period during the test session and to enable studying possible differences between the data distributions.

- The samples were not continuous in time, to prevent attempts of ordering the samples given in random order.

The detailed split into training and test samples is described in Figure 2. In brief, both the training and test samples were of 1-s length and were separated from each other by 1 s. Out of the four blocks of short clips the blocks A, B and D recorded during day 1 were used for training and blocks A, B and C recorded during day 2 for testing. This resulted in 66% of the test samples having stimuli that exists also in the training data. The clips in block E were split to training and test data so that time (roughly) between 1:40 and 6:10 was used for training and time between 3:10 and 7:40 for testing, resulting in 68% of overlap between training and test data. Finally, a random class-balanced subset of 50 test samples were released with labels.

The training and test samples had, however, 1-s offset in timings. Hence, even the set of samples using the same stimuli are not exactly from the same time but instead are consecutive time points. If the time window between seconds 3 and 4 was used for training, then the window between seconds 4 and 5 was used for testing.

Overall, the setup resulted in 677 training samples with roughly class-balanced distribution (the number of samples for the five classes were: 140, 171, 96, 135, and 135), 50 labeled test samples, and 653 unlabeled test samples that the competitors needed to classify. The data are available at `http://www.cis.hut.fi/icann2011/mindreading.php`, and can be used for research purposes and scientific publications.

3.2 Machine learning concepts

Even though the main problem is that of regular classification, the particular setup of learning to decode MEG measurements leads to a number of more detailed machine learning challenges. Here we briefly overview the kind of aspects initially thought to be relevant for the task. The research on machine learning solutions for MEG mind decoding tasks would likely benefit from tackling these modeling issues, besides just working towards improved MEG signal analysis in general.

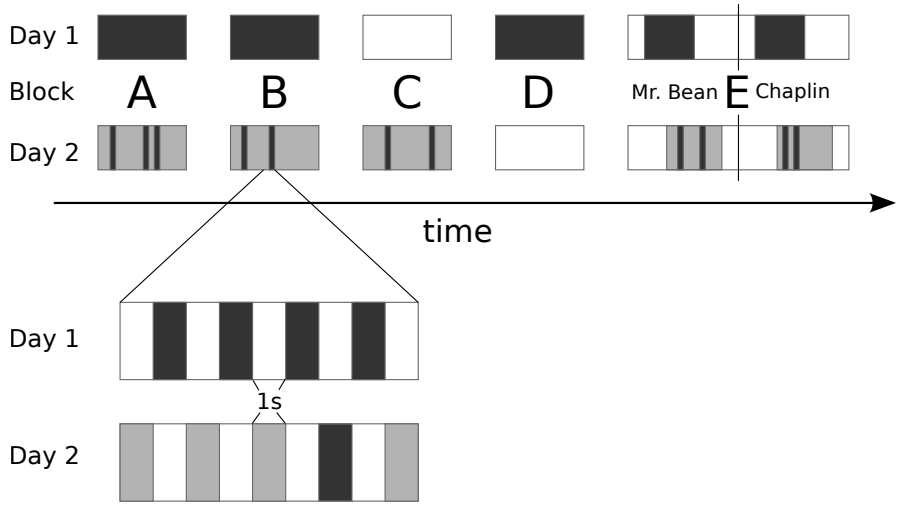

Figure 2. Illustration of the data split. The dark grey boxes correspond to the selection of data points for training data, the light gray boxes correspond to the choice of test samples, whereas the unshaded areas were not used in the challenge at all. The dark areas on the second day indicate the random choice of labeled test samples. Note that blocks A and B contained the same stimuli. The closeup shows how the 1-s samples were chosen with 1-s gaps between each other, and how the training and test samples taken from the same block were misaligned by 1 s.

Covariate shift/domain adaptation For real-use cases brain decoding systems need to work for new recording sessions, besides being able to predict merely new time points of existing recordings. Since (1) MEG instrumentation is subject to stochastic noise, and (2) since the state of the subject varies strongly from day to day, the data recorded during a different session generally do not follow the same distribution as the training data. Hence, computational models taking into account a change in the data distribution are needed. This problem is generally tackled under the term of domain adaptation [14], which is an active line of research in the machine learning community.

Multi-view learning MEG recording produces measurements for 204 gradiometer channels and 102 magnetometer channels, and for each signal we can extract multiple frequency bands or other types of features. Information encoded in different channels, frequency bands, and across different time scales is largely complementary. This suggests that multi-view learning methods could be useful for MEG decoding tasks. While it is possible to attempt decoding the stimuli from individual channels or based on simple predictors operating on all channels, there is reason to believe that clever integration of the different channels and frequency bands through

multi-view learning models could result in improved accuracy, as well as improved understanding of the underlying brain processes.

Multi-view learning methods have also been used for solving decoding tasks outside classification. For identification, multi-view learning methods based on canonical correlation analysis (CCA), such as the Bayesian CCA [18], can be used for extracting correlating projections of the brain activity and stimulus description, enabling direct comparison of brain measurements of test samples to the set of possible stimuli. Multi-view learning methods have also been used for extracting image bases for visual image reconstruction [19], as well as for inferring properties of natural music based on fMRI [20].

Generalization and overfitting Another consequence of the high-dimensional nature of the MEG recording is that it is very easy to overfit to the available training data. Therefore a successful decoding solution will have to be very carefully regularized to control the degree of generalization to new data. Many of the decoding works apply Bayesian modeling techniques [10, 11], which provide a way of tackling the overlearning issue in a justified way, or apply sparse solutions such as lasso regression [15].

Multi-task learning The variability across subjects is large for all brain imaging techniques. Typical analysis methods will either assume that all subjects are identical, which is a simplifying but incorrect assumption, or will resort to subject-specific modeling resulting in no information being transferred from one subject to another. Multi-task learning [16] studies computational models that combine the strengths of both approaches, by learning separate predictive models for the subjects simultaneously, so that the similarities between the subjects are utilized for improved accuracy while still allowing subject-specific variation. In this challenge, we provided data only from a single subject, and hence such models could not be applied, but in general multi-task learning of decoding models is likely to be crucial. Recently, Alamgir et al. [21] demonstrated how multi-task learning improved accuracy for EEG-based brain computer interfaces.

Table 2. The list of participating teams in alphabetical order of the first author. The team Tu & Sun provided two different solutions.

Name	Authors / Institute
Van Gerven & Farquhar	M.A.J. Van Gerven, J. Farqugar Donders Institute for Brain, Cognition and Behaviour, Radboud University Nijmegen, the Netherlands
Grozea	C. Grozea Fraunhofer Institute FIRST, Germany
Huttunen et al.	H. Huttunen, J-P. Kauppi, J. Tohka Department of Signal Processing, Tampere University of Technology, Finland
Jylänki et al.	P. Jylänki, J. Riihimäki, A. Vehtari Dept. of Biomedical Engineering and Computational Science, Aalto University, Finland
Lievonen & Hyötyniemi	P. Lievonen, H. Hyötyniemi Helsinki Institute for Information Technology HIIT, Finland
Nicolaou	N. Nicolaou Dept. Of Electrical and Computer Engineering, University of Cyprus, Cyprus
Olivetti & Melchiori	E. Olivetti, F. Melchiori NeuroInformatics Laboratory (NILab), Bruno Kessler Foundation and University of Trento, Italy
Santana et al.	R. Santana, C. Bielza, P. Larrañaga Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Spain
Tu & Sun	W. Tu, S. Sun Department of Computer Science and Technology, East China Normal University, China

4 Results

Overall, the challenge received 10 submissions from 9 different teams listed in Table 2. Multiple submissions per team were allowed if the solutions utilized significantly different modeling approaches.

4.1 Challenge results

The main criterion for evaluating the submissions was the classification accuracy on the test data. The baseline accuracy of predicting every sample to belong to the largest class in the training set would be 23%. The results of the participants are summarized in Table 3, showing that all

Table 3. The prediction accuracies (percent, bigger is better) of the competitors, sorted in the order of the overall accuracy that was the criterion for evaluating the submissions. The last three columns show the accuracy in separating the content with plot from the short clips (PvsC), the accuracy in predicting the short clip classes correctly (C), and the accuracy in identifying the longer clips with plot correctly (P). For all tasks the best accuracy has been boldfaced. A notable observation is that the best solution outperforms all others in making the correct predictions within both stimulus categories, but is only 7th best in making the split between the two categories. The last line shows the accuracy of majority voting based on the top nine submissions.

Team	Accuracy	PvsC	C	P
Huttunen et al.	**68.0**	89.7	**67.5**	**89.2**
Santana et al.	63.2	93.0	64.1	74.0
Jylänki et al.	62.8	93.0	56.8	85.8
Tu & Sun	62.2	**97.1**	50.1	87.0
Lievonen & Hyötyniemi	56.5	91.0	55.7	72.4
Tu & Sun (2)	54.2	96.6	44.3	75.8
Olivetti & Melchiori	53.9	94.6	41.4	85.4
Van Gerven & Farquhar	47.2	82.4	53.3	66.3
Grozea	44.3	88.5	39.1	67.7
Nicolaou	24.2	61.7	34.8	49.6
Pooled (top 9)	69.2	96.8	63.1	85.8

but one of the participants clearly surpass the baseline level, demonstrating successful brain decoding. The outlier submission falls at the chance level, suggesting either very heavy overlearning or mistakes in implementation. The range of accuracies, excluding the outlier, falls between 44% and 68%, demonstrating that there is a notable difference between the alternative decoding solutions. The solution of Huttunen et al. outperforms others by a margin of almost five percent, ending up as the clear winner, followed by three other solutions above 60% accuracy.

For many classification tasks combining several classifiers results in improved performance. While various advanced solutions, such as boosting, can be used for obtaining maximal benefit from multiple classifiers, already a simple majority voting of the results provides often a reasonably good model. Here, the combination of all 10 solutions results in accuracy of 68.9% and the combination of the 9 solutions exceeding the chance level gives 69.8%. Both figures are better than the best solution, but the margin is smaller than the difference between the individual solutions.

As the stimuli to be decoded consisted of two distinct categories, directed

Huttunen et al.

	1	2	3	4	5
1	94	29	16	10	1
2	22	100	10	18	1
3	25	16	51	10	0
4	3	4	12	85	21
5	2	2	4	3	114

Santana et al.

	1	2	3	4	5
1	67	54	14	15	0
2	25	110	5	11	0
3	19	14	57	12	0
4	1	1	5	59	59
5	1	0	0	4	120

Jylänki et al.

	1	2	3	4	5
1	67	32	43	8	0
2	36	89	18	8	0
3	30	6	61	4	1
4	6	6	11	78	24
5	1	0	1	8	115

Tu & Sun

	1	2	3	4	5
1	56	55	36	3	0
2	30	96	21	4	0
3	33	22	46	1	0
4	4	3	3	95	20
5	1	0	0	11	113

Figure 3. Confusion matrices of the top four submissions. The rows correspond to the true classes, whereas the columns are the predicted classes. The labels are 1:artificial, 2:football, 3:nature documentary, 4:Mr.Bean, 5:Chaplin.

films with clear storyline and short video clips, we can also look at the success rate in separating these two categories as well as the accuracy in classifying the samples within either category (Table 3). The accuracy in separating the two categories is computed as the binary classification accuracy, whereas the accuracy within each category is measured with the ratio of correct assignment amongst all samples for which both the true and predicted class are within that category. Interestingly, the best submission is not amongst the top ones in the easier task of separating the clips with plot from the rest, but has the best accuracy within both stimulus categories. One possible reason is that the other solutions have overfitted to solving the easier task of binary separation between the two categories. This is illustrated by the confusion matrices of the best four solutions in Figure 3.

The best solutions are described in more detail in the separate articles following this overview. Overall, the solutions focused quite strongly in feature selection, either by careful validation of possible alternative features or by building classifiers with automatic feature selection, such as L1-regularized lasso models. One team, Santana et al., tried an ensemble of more than one classifier. Three of the competitors, Olivetti & Melchiori

and both submissions by Tu & Sun, focused on solving the domain adaptation problem with advanced machine learning techniques, each having reasonable performance but not reaching the top positions, while many of the other teams addressed the shift in input distributions by placing more weight on the labeled test examples when validating the learned classifier.

4.2 Alternative prediction tasks

Even though the challenge was defined as decoding the stimulus based on 1-s MEG epochs, we can estimate how well the solutions would have fared with longer observations by pooling the predictions given for consecutive samples. For this purpose, we looked at the predictions obtained by majority vote for each short clip (classes 1-3), averaging as 8 observation per clip, and for each collection of 8 consecutive samples for the longer clips (classes 4-5). The best submission then gives 80% accuracy in predicting the class correctly for each clip or 8s period (Table 4), supporting the intuitive belief that solving the decoding task is easier based on longer observations. These accuracies provide a lower bound for the accuracy the competitors could have obtained if they had access to such 8s observations and had explicitly developed predictors for solving this alternative task.

As described in Section 3.1, the data set was split so that some of the test samples were picked from the same clips as the training samples (though with 1-s offset) while some were not. Even though the competitors were not aware which samples matched the training samples, we can inspect whether the accuracy of decoding differs from the two sets. Table 4 shows how almost all participants were more accurate in predicting the samples taken from the same clips that were available in training, providing a quantification of the increase in difficulty in brain decoding due to completely new stimulus content. On average, the accuracy was 6.3 percentage points higher for the samples included in the training data.

5 Discussion

The primary task in the challenge was to decode the type of the video stimulus from MEG data. Nine out of ten submissions succeeded in this task significantly above the chance level, showing that it is possible to decode

Table 4. Results of alternative decoding tasks (not part of the competition), sorted in order of the performance in the challenge results. For each task the best accuracy is boldfaced. The first column shows the accuracy for predicting correctly the whole clips by majority voting based on the samples within each clip (on average 8 samples per clip). For all but one participant the accuracy is better than when decoding the label for 1s samples, as expected. The second column gives the accuracy in the challenge decoding task for test samples taken from the clips used also in the training set, whereas the last column gives the accuracy for the test samples from clips not seen in the training set. For all contestants except one, the accuracy is better for the first group, showing clearly how generalizing to the new stimulus content makes the decoding task harder. Still, the accuracies for the new content are well above chance level.

Team	Full clips	Within train	Not in train
Huttunen et al.	**79.7**	**69.9**	**64.2**
Santana et al.	68.5	65.1	59.6
Jylänki et al.	76.0	66.2	56.0
Tu & Sun	70.7	64.4	57.8
Lievonen & Hyötyniemi	62.2	59.8	50.0
Tu & Sun (2)	57.0	59.5	43.6
Olivetti & Melchiori	61.8	55.6	50.5
Van Gerven & Farquhar	55.9	49.0	43.6
Grozea	41.3	44.4	44.0
Nicolaou	25.0	23.7	25.2

the various kinds of stimuli already from short 1-s windows of MEG data. The difference in accuracies between the approaches was considerable, with the best solution reaching near 70% accuracy while the majority of the solutions had around 50% success, showing that carefully developed machine learning solutions will achieve improved accuracy in brain decoding. Still a clear gap exists between the best solution and perfect accuracy, demonstrating that the task is far from trivial and especially that perfect decoding results are unlikely to be obtained with such brief signals, probably because of the low signal-to-noise ratio of the single-trial MEG. By pooling the competitors results for longer (8 s) periods of observations, the accuracy of the best solutions increases close to 80%. In future research it could be advisable to directly study the accuracy on multiple timescales, to better estimate the amount of data needed for inferring different types of stimuli.

The majority of the competitors focused on good feature selection and cross-validation of the learned models, demonstrating once again the importance of carefully controlling overlearning. In this challenge this aspect was particularly important due to the relatively big change in input data distribution between training and test data. For example, the top team explicitly mentioned in their submission that some of the more advanced features were neglected for that reason. Many of the teams also addressed the domain adaptation problem seriously. Some of the competitors handled the adaptation by giving more weight for the labeled test samples in cross-validation, whereas some teams applied more advanced techniques for correcting for the shift in the distribution, using methods of EasyAdapt, transfer-priority cross validation and transferable discriminant analysis.

In future, it would be interesting to see challenges with brain signals from more subjects. This would enable studying more advanced modeling concepts such as multi-task learning, while also providing information on to which extent the perceptual processes that are best for decoding the stimuli are shared by individuals. However, prior to releasing such data sets it could be beneficial to create a more finely processed feature set, since otherwise the amount of data becomes infeasible. Now the data of just one subject took a total of roughly 6 gigabytes in compressed format, and started to become a technical difficulty for some competitors.

For this challenge we used decoding accuracy as the primary criterion and evaluated the submissions additionally based on methodological nov-

elty of the approach. The challenge was also primarily advertised for modelers. Consequently, the submissions focused on these aspects and no neuroscientific interpretations were made. In future challenges it could be a good idea to value also neuroscientific findings when determining the winners, to encourage tighter interaction between modelers and neuroscientists as well as to provide insights into the perceptual processes revealed by succesful decoding.

Acknowledgments

We gratefully acknowledge the PASCAL2 European Network of Excellence, in particular their Challenge Programme, for sponsoring the challenge. We also thank the ICANN conference for hosting the challenge workshop.

The organizers have additionally received support from Academy of Finland (project numbers #133818 and #218072, and National Centers of Excellence Program 2006-2011) and from the aivoAALTO research project of the Aalto University.

Finally, we thank Kranti Kumar Nallamothu for preparing the stimuli for the experiment.

Bibliography

[1] J-D. Haynes and G. Rees. Decoding mental states from brain activity in humans. *Nature Reviews Neuroscience*, 7:523–534, 2006.

[2] Y. Kamitani and F. Tong. Decoding the visual and subjective contents of the human brain. *Nature Neuroscience*, 8(5):679–685, 2005.

[3] E. Formisano, F. De Martino, M. Bonte, and R. Goebel. "Who" is saying "What"? Brain-based decoding of human voice and speech. *Science*, 322(5903):970–973, 2008.

[4] K.N. Kay, T. Naselaris, R.J. Prenger, and J.L. Gallant. Identifying natural images from human brain activity. *Nature letters*, 452(20):352–356, 2008.

[5] T.M. Mitchell, S.V. Shinkareva, A. Carlson, K.M. Chang, V.L. Malave, R.A. Mason, and M.A. Just. Predicting human brain activity associated with the meanings of nounds. *Science*, 320(1191), 2008.

[6] Y. Miyawaki, H. Uchida, O. Yamashita, M. Sato, Y. Morito, H.C. Tanabe, N. Sadato, and Y. Kamitani. Visual image reconstruction from human brain activity using a combination of multiscale local image decoders. *Neuron*, 60:915–929, 2008.

[7] T.Naselaris, R.J. Prenger, K.N. Kay, M. Oliver, and J.L. Gallant. Bayesian reconstruction of natural images from human brain activity. *Neuron*, 63:902–915, 2009.

[8] J.W. Rieger, C. Reichert, K.R. Gegenfurtner, T. Noesselt, C. Braun, H-J. Heinze, R. Kruse, and H. Hinrichs. Predicting the recognition of natural scenes from single trial MEG recordings of brain activity. *Neuroimage*, 42:1056–1068, 2008.

[9] S. Waldert, H. Preissl, E. Demandt, C. Braun, N. Birbaumer, A. Aertsen, and C. Mehring. Hand movement direction decoded from MEG and EEG. *Journal of Neuroscience*, 28(4):1000–1008, 2008.

[10] A. Toda, H. Imamizu, M. Kawato, and M.A. Sato. Reconstruction of two-dimensional movement trajectories from selected magnetoencephalography cortical currents by combined sparse Bayesian methods. *NeuroImage*, 54(2):892–905, 2011.

[11] K. Friston, C. Chu, J. Mourao-Miranda, O. Hulme, G. Rees, W. Penny, and John Ashburner. Bayesian decoding of brain images. *NeuroImage*, 39(1):181–205, 2008.

[12] M. Palatucci, D. Pomerleau, G. Hinton, and T. Mitchell. Zero-shot learning with semantic output codes. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1410–1418, 2009.

[13] H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

[14] H. Daume III and D. Marcu. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research*, 26:101–126, 2006.

[15] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of Royal Statistical Society B*, 58(1):267–288, 1996.

[16] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[17] S. Taulu and J. Simola. Spatiotemporal signal space separation method for rejecting nearby interference in MEG measurements. *Physics in Medicine and Biology*, 51(7):1759–1768, 2005.

[18] A. Klami and S. Kaski. Local dependent components. In *Proceedings of ICML 2007, the 24th International Conference on Machine Learning*, pages 425–432, 2007.

[19] Y. Fujiwara, Y. Miyawaki, and Y. Kamitani. Estimating image bases for visual image reconstruction from human brain activity. In *Advances in Neural Information Processing Systems 22*, pages 576–584, 2009.

[20] S. Virtanen, A. Klami, and S. Kaski. Bayesian CCA via group sparsity. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 2011.

[21] M. Alamgir, M. Grosse-Wentrup, and Y. Altun. Multitask learning for brain-computer interfaces. In *Proceedings of 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 17–24, 2010.

Regularized Logistic Regression for Mind Reading with Parallel Validation

Heikki Huttunen, Jukka-Pekka Kauppi, Jussi Tohka
Tampere University of Technology
Department of Signal Processing
Tampere, Finland
{jukka-pekka.kauppi,heikki.huttunen,jussi.tohka}@tut.fi

Abstract

In our submission to the mind reading competition of ICANN 2011 conference, we concentrate on feature selection and result validation. In our solution, the feature selection is embedded into the regression by adding a ℓ_1 penalty to the classifier cost function. This can be efficiently done for regression problems using the LASSO, which generalizes also to classification problems in the logistic regression classification framework. A special attention is paid to the evaluation of the performance of the classification by cross-validation in a parallel computing environment.

1 Introduction

Together with the ICANN 2011 conference, a competition for classification of brain MEG data was organized. The challenge was to train a classifier for predicting the movie being shown to the test subject. There were five classes and the data consisted of 204 channels. Each measurement was one second in length and the sampling rate was 200 Hz. From each one-second measurement, we had to derive discriminative features for classification. Since there were only a few hundred measurements, the number of features will easily exceed the number of measurements, and

thus the key problem is to select the most suitable features efficiently. We tested various iterative feature selection methods including the *Floating Stepwise Selection* [4] and *Simulated Annealing Feature Selection* [1], but obtained the best results using Logistic Regression with ℓ_1 penalty also known as the *LASSO* [5].

2 Logistic Regression with the LASSO

Our classifier is based on the logistic regression model for class probability densities. The logistic regression models the PDF for the class $k = 1, 2, \ldots, K$ as

$$p_k(\mathbf{x}) = \frac{\exp(\boldsymbol{\beta}_k^T \mathbf{x})}{1 + \sum_{j=1}^{K} \exp(\boldsymbol{\beta}_j^T \mathbf{x})}, \text{ for } k \neq K, \text{ and} \quad (1)$$

$$p_K(\mathbf{x}) = \frac{1}{1 + \sum_{j=1}^{K} \exp(\boldsymbol{\beta}_j^T \mathbf{x})}, \quad (2)$$

where $\mathbf{x} = (1, x_1, x_2, \ldots, x_p)^T$ denotes the data and $\boldsymbol{\beta}_k = (\beta_{k0}, \beta_{k1}, \beta_{k2}, \ldots, \beta_{kp})^T$ are the coefficients of the model.

The training consists of estimating the unknown parameters $\boldsymbol{\beta}_k$ of the regression model, which can then be used to predict the class probabilities of independent test data. The simplest approach for estimation is to use an iterative procedure such as iteratively reweighted least squares (IRLS).

In the mind reading competition the number of variables is large compared to the number of measurements. If additional features are derived from the measurement data, the number of parameters p to be estimated easily exceeds the number of measurements N. Therefore, we have to select a subset of features that is the most useful for the model. Our first attempt was to find the optimal subset iteratively using simulated annealing, but soon we decided to use a method, with feature selection embedded into the cost function used in the classifier design. LASSO *(Least Absolute Shrinkage and Selection Operator)* regression method enforces sparsity via ℓ_1 -penalty, and in a least squares model the constrained LS criterion is given by [5, 3]:

$$\min_{\beta} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|_1 , \quad (3)$$

where $\lambda \geq 0$ is a tuning parameter. When λ is small, this is identical to the OLS solution. With large values of λ, the solution becomes shrunken

and sparse version of the OLS solution where only a few of the coefficients β_j are non-zero.

The LASSO has been recently extended for logistic regression [2], and a Matlab implementation is available at `http://www-stat.stanford.edu/~tibs/glmnet-matlab/`.

We experimented with numerous features fed to the classifier, and attempted to design discriminative features using various techniques. Our understanding is that those ended up being too specific for the the first day data and eventually a simplistic solution turned out to be the best, resulting in the following features:

- The detrended mean, i.e., the parameter $\hat{b}$ of the linear model $y = ax + b$ fitted to the time series.

- The standard deviation of the residual of the fit, i.e., stddev$(\hat{y} - y)$.

Both channels were calculated from the raw data; we were unable to gain any improvement from the filtered channels. Since there were initially 204 measurements, this makes a total of 408 features from which to select.

3 Results and Performance Assessment

An important aspect for the classifier design is the error assessment. This was challenging in the mind reading competition, because only a small amount (50 samples) of the test dataset was released with the ground truth. Additionally, we obviously wanted to exploit it also for training the model. In order to simulate the true competition, we randomly divided the 50 test day samples into two parts of 25 samples. The first set of 25 samples was used for training, and the other for performance assessment. Since the division can be done in $\binom{50}{25} > 10^{14}$ ways, we have more than enough test cases for estimating the error distribution.

The remaining problem in estimating the error distribution is the computational load. One run of training the classifier with cross-validation of the parameters takes typically 10-30 minutes. If, for example we want to test with 100 test set splits, we would be finished after a day or two. For method development and for testing different features this is certainly too slow.

Tampere University of Technology uses a grid computing environment developed by Techila Oy (`http://www.techila.fi/`). The system allows

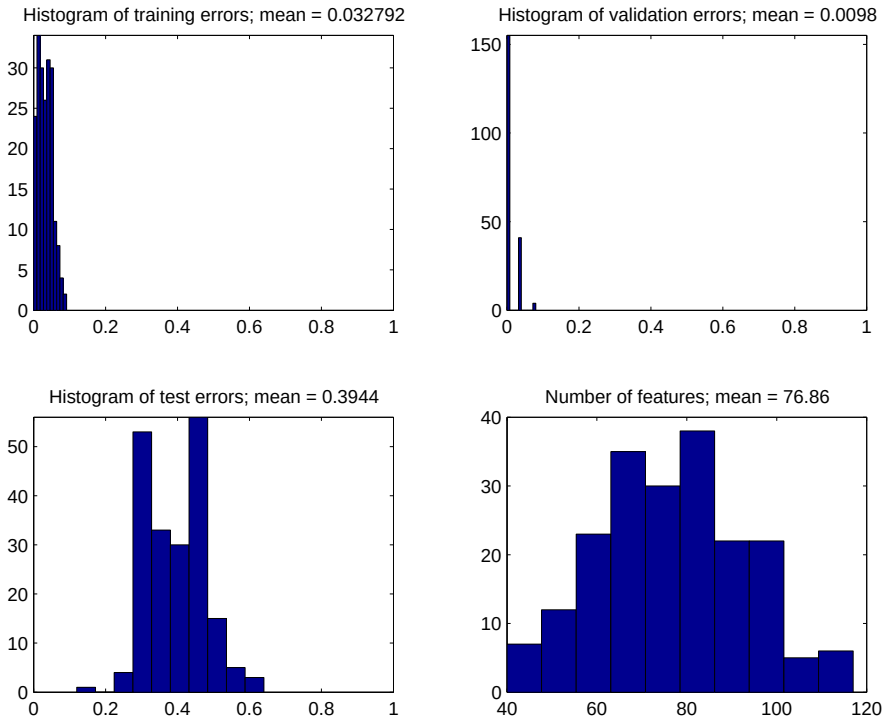

Figure 1. An example result of the error estimation. Figures (a-c) show the error distribution for the first day training data, for the 25 samples of second day data used for training and for the rest of the second day data, respectively. Figure (d) shows the number of features used on the average for the model. In this experiment we run the training for 200 test cases.

distributing the computation tasks to all computers around campus. Since our problem is parallel in nature, incorporating the grid into the performance assessment was easy: A few hundred splits of the test set were done, and one processor in the grid was allocated for each case.

Figure 1 illustrate an example test run. The performance can be assessed from the error distribution for the test data shown on Figure on the bottom left, which in this case is 0.394, or 60.6 % correct classification. We believe that the final result will be slightly better, because the final classifier is trained using all 50 samples of the second day data.

The error for the validation data (top right figure) is very small. This is because the samples were weighted such that second day data has a higher weight in the cost function.

After preparing the final submission, we studied the distribution of the predicted classes for the test data. The test samples should be roughly class-balanced, so a very uneven distribution could indicate a problem (such as a bug) in the classification. We also considered the possibility of fine tuning the regularization parameter based on the balancedness of the corresponding classification result. As the balancedness index we used the ratio of the cardinalities of the largest and smallest classes in

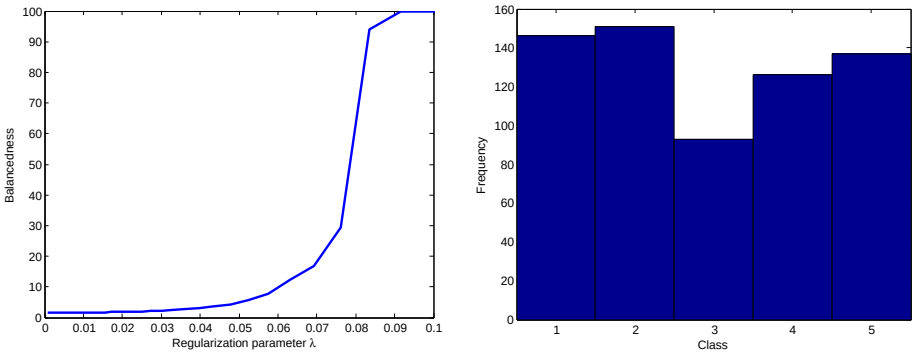

Figure 2. Left: The balancedness index for different values of regularization parameter. Right: The predicted class histogram for the test data.

the predicted result. The balancedness indicator is plotted as a function of the regularization parameter λ in Figure 2 (left).

The result clearly emphasizes the old rule: regularization increases the bias but decreases the variance. This can be seen from the curve in that less regularization (small λ) improves the class-balance (indicator close to unity). However, since it seems that the regularization parameter $\lambda = 0.0056$ selected using cross validation is at the edge of the well balanced region, we decided not to adjust the CV selection. The final predicted class distribution is shown in Figure 2 (right).

Bibliography

[1] Debuse, J.C., Rayward-Smith, V.J.: Feature subset selection within a simulated annealing data mining algorithm. Journal of Intelligent Information Systems 9, 57–81 (1997), 10.1023/A:1008641220268

[2] Friedman, J.H., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. Journal of Statistical Software 33(1), 1–22 (2010)

[3] Hastie, T., Tibshirani, R., Friedman, J.: The elements of statistical learning: Data mining, inference, and prediction. Springer Series in Statistics, Springer (2009)

[4] Pudil, P., Novovičová, J., Kittler, J.: Floating search methods in feature selection. Pattern Recogn. Lett. 15(11), 1119–1125 (Nov 1994)

[5] Tibshirani, R.: Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society, Series B 58, 267–288 (1994)

An ensemble of classifiers approach with multiple sources of information

Roberto Santana, Concha Bielza, Pedro Larrañaga
Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid
roberto.santana@upm.es, mcbielza@fi.upm.es,
pedro.larranaga@fi.upm.es

Abstract

This paper describes the main characteristics of our approach to the ICANN-2011 Mind reading from MEG - PASCAL Challenge. The distinguished features of our method are: 1) The use of different sources of information as input to the classifiers. We simultaneously use information coming from raw data, channels correlations, mutual information between channels, and channel interactions graphs as features for the classifiers. 2) The use of ensemble of classifiers based on regularized multi-logistic regression, regression trees, and an affinity propagation based classifier.

1 Type of information used for classification

The first building block of our approach is the combination of different sources of information extracted from the MEG signals. We hypothesize that different transformations to the brain signals could reveal diverse types of brain signatures useful for the classification purpose. Therefore, we have tried different information processing variants to unveil this information. In all cases, the starting point was the time series output from the $N = 727$ training cases, for the $k = 204$ channels. For the training set, there are a total 727 cases and 204 time series for each case. The MEG

output data corresponds to 200-component numerical vector.

The first type of brain signal representation is constructed by splitting the time series in segments of 5 contiguous time points, and adding the raw signals in each segment. We obtain, for each channel, a vector of 50 features. Therefore, for a fixed frequency, each of the 727 cases will be represented by $204 \times 50 = 10200$ features. We call to this relatively simple transformation of the initial information *raw data*.

For each of the cases, we use its corresponding raw data to compute the correlations between each pair of channels for this case. For example, to compute the correlations between channels i and j, their corresponding vectors of 50 raw values are used. As a result, a symmetric matrix $\mathbf{W}_{204\times204}$ is obtained from each case. The final set of features of each case will comprise a vector of $n = \frac{204 \cdot 203}{2} = 20706$ values corresponding to the upper triangular part of the correlation matrix (without the main diagonal). This type of information is called *channels correlations*. This approach intends to compute the interaction between different brain regions during the solution of the recognition task.

In a similar way we compute, for each case, the matrix of mutual information between the channels. First, the continuous data corresponding to two variables, are discretized and from the discretized values the mutual information is obtained. The bin size for discretizing all the data was fixed to equal value of 11. Similarly to the computation of the correlation, the final set of features will comprise vector of $n = \frac{204 \cdot 203}{2} = 20706$ values which are called the *mutual information between channels*. This approach also tries to unveil interaction between different brain regions that could be specific to each mental task.

In the fourth signal processing procedure, the correlation matrix is used to construct interaction graphs between the different channels. The idea is that a further analysis of the graph using topological measures from network theory can serve to reveal local and global information that is not directly recognizable from the correlation values.

The interaction graph $G = (V, A)$ is such that $V = \{v_1, \ldots, v_{204}\}$ is the set of vertices and arc $a_{i,j}$ between vertices v_i and v_j is defined as follows:

$$a_{i,j} = \begin{cases} 1 & if \quad i < j \text{ and } cr_{i,j} > 0.5 \\ -1 & if \quad i < j \text{ and } cr_{i,j} < -0.5 \\ 0 & \text{otherwise} \end{cases}$$

where $cr_{i,j}$ is the correlation coefficient between channels i and j, and

$Information - Freq$	$Full$	$2H$	$5H$	$10H$	$20H$	$35H$
Raw	236	0	0	0	0	0
$Correlation$	547	64	122	501	806	3566
$MutualInf.$	31	5	14	49	98	356
$Interactiongraph$	16	0	0	39	61	349

Table 1. Number of selected features of each type of information and frequencies.

values 1, -1 and 0 for $a_{i,j}$ respectively mean that there is an arc from v_i to v_j, there is an arc from v_j to v_i, or there is no arc between v_i and v_j.

The interaction graph is an arbitrary way to represent strong correlations (below -0.5 or above 0.5) between pairs of channels. We expect that if there are higher order interaction patterns between the channels, at least some of them could be unveiled by a topological analysis of these graphs.

Once correlation graphs have been constructed, a number of (local) topological measures are computed for each node (e.g. clustering coefficient, path length, betweenness centrality, etc.). In addition, a number of global topological measures are computed for the complete graph (e.g. graph density, graph diameter, etc.). The number of local features was $n_{local} = 204 \cdot 13 = 2652$ and the number of global features was $n_{global} = 7$. The total number of topological features extracted for each graph was $n = 2659$. We call to this type of information *channel interactions graphs*.

1.1 Feature selection

In order to identify a reduced set of significant features, we applied, for each feature, a statistical test to determine whether there exists significant different between the 5 different classes for the given feature. The statistical test was applied to each pair of classes. The idea was to identify whether a given feature is effective at identifying differences between any of the 10 possibles pairs of classes. A more stringent requirement would be the identification of features that are significantly different between the 5 classes altogether. However, in our approach we keep features that detect "local" differences between classes.

The statistical test of choice was the Wilconxon rank sum test of equal medians and the parameter $\alpha = 10^{-5}$ was fixed for all the statistical tests. Table 1 shows the number of significant features found for each frequency

	Raw data					Correlation				
Class	1	2	3	4	5	1	2	3	4	5
1	–	**63**	96	10	8	–	61	67	2326	2922
2		–	221	46	0		–	281	2374	3437
3			–	12	55			–	1852	2456
4				–	**0**				–	3102

	Mutual information					Interaction graph				
Class	1	2	3	4	5	1	2	3	4	5
1	–	10	3	249	221	–	**0**	5	92	151
2		–	25	270	254		–	4	281	147
3			–	211	174			–	244	133
4				–	330				–	**278**

Table 2. Number of significant features for all pairs of classes and types of information.

and each type of information. Table 2 shows the number of significant features found for each pair of classes and using all sources of information. Notice that a feature may be significant in the comparison of two or more pairs of variables. Emphasized in bold are the marked differences between the raw data and the interaction graph types of information in terms of the number of relevant features they respectively find for class pairs $(1, 2)$ and $(4, 5)$. These differences confirm our hypothesis that different types of information may reveal different types of brain signatures.

For the classification purpose we use the combined set of all the 6860 relevant features included in Table 1.

2 Classification approaches

Three different classification approaches were used: Elastic net regularized multi-logistic regression [3], regression trees [1] and affinity propagation [2]. The first two methods are supervised classification methods and were initially evaluated in the training set using a 5-fold cross-validation scheme. The second method is an unsupervised classification method that we directly used as a way to classify the test cases similarly as described in [4].

Using 5-fold cross-validation on the training set with the complete set

of 6860 variables we observed that elastic net multi-logistic regression was able to reach a 0.83 classification rate for different values of $\beta \in \{0.01, \ldots, 0.9\}$. We then trained the model using the complete set of 727 solutions and used it to classify the test set. 21 different classifications corresponding to different pairs of (α, β), those that achieved and accuracy over 0.98 in the complete training set, were obtained. We called this set of solutions MLRSet.

To evaluate the regression trees, the set of 6860 variables was split into 26 different sets of (overlapping) variables. Each set excluded a subset of features relevant in the identification of 2, 3 or 4 classes, i.e. we used the grouping of variables shown in Table 2 to partition the set of variables. For each subset of features, we used cross-validation on the training set, to learn a regression tree for each subset of features. Of the initial set of 26, three regression trees were removed due to achieve a classification accuracy under 0.48. The remaining 23 were used to create an ensemble of regression trees with the majority vote strategy. Its application, using 5-fold cross-validation on the training set gave an accuracy of 0.6066. The application of each individual tree to the test set produced a set of 23 solutions. We called this set of solutions TreeSet.

Affinity propagation was applied to the combined set of training and test cases. However, by penalizing the preference values of the test cases we enforce that only train cases are allowed to be an exemplar. A test case is classified in the same class its corresponding exemplar belongs to. To evaluate the quality of the classification, we computed the number of non-exemplar training cases that were correctly classified. We have previously observed [4] that this may be an indirect measure of the classification quality for the test cases. 9 different similarity measures were applied to the 26 sets of variables in which the initial set of features was partitioned. As a result, we obtained a set of 234 clusterings. From these clusterings, we selected those for which the number of correctly classified non-exemplar training cases was above 0.60. There were 11 such clusterings. Each cluster determines a assignment to the test cases. We called this set of solutions APSet.

To obtain the final solution, we compute, for each of the three sets produced by the classifiers, the class probability for each test case. The class probability is simply the frequency of each class in the corresponding set for the given test case. The final probability of a case is found as a weighted sum of the probabilities for each of the three sets, i.e.

$p_F = 0.4p_{MLRSet} + 0.3p_{TreeSet} + 0.3p_{APSet}$. The weights were determined according to the accuracies obtained by the two supervised classification algorithms in the training set and we assumed that affinity propagation achieved a classification rate similar to regression trees. The final assignment of a given test case will correspond to the class with the highest class probability in p_F.

Bibliography

[1] L. Breiman. *Classification and regression trees*. Chapman & Hall/CRC, 1984.

[2] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007.

[3] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

[4] R. Santana, C. Bielza, and P. Larrañaga. Affinity propagation enhanced by estimation of distribution algorithms. In *Proceedings of the 2011 Genetic and Evolutionary Computation Conference GECCO-2011*, Dublin, Ireland, 2011. Accepted for publication.

Multi-class Gaussian process classification of single-trial MEG based on frequency specific latent features extracted with linear binary classifiers

Pasi Jylänki, Jaakko Riihimäki, Aki Vehtari
Deptartment of Biomedical Engineering and Computational Science, Aalto University, Finland
{pasi.jylanki, jaakko.riihimaki, aki.vehtari}@aalto.fi
http://www.becs.tkk.fi

The approach is based on calculating power features from the filtered MEG signals and doing a supervised linear dimensionality reduction for the gradiometer channel space. The dimensionality reduction is done with binary classifiers separately for each class and frequency band. The resulting lower dimensional features are classified using a multi-class Gaussian process classifier [2].

The Power features were extracted by calculating the mean squared amplitude from all the 204 planar gradiometer channels for each of the five prefiltered frequency bands. Logarithms of these power features were normalized to zero mean and unit variance separately for the both measurement days to give a 204-dimensional feature vector $\mathbf{x}_{i,k}$ for all the labeled observations $i = 1, ..., n$ and frequency bands $k = 1, .., K$, where $K = 5$.

Dimensionality reduction

Linear one-versus-rest logistic classifiers were used to reduce the 204-dimensional feature space into a one dimensional latent space for each of the five classes and five frequency bands separately. For a frequency band k and an input vector $\mathbf{x}_{i,k}$, the probability of class c is modeled as

$$p(y_{i,c} = 1 | \, \mathbf{w}_{k,c}, v_{k,c}, x_{i,k}) = (1 + \exp(-z_{i,k,c}))^{-1}, \tag{1}$$

where $\mathbf{w}_{k,c}$ are the coefficients of the linear predictor and $v_{k,c}$ a bias term, $z_{i,k,c} = \mathbf{x}_{i,k}^{\mathrm{T}}\,\mathbf{w}_{k,c} + v_{k,c}$ the latent value we are trying to estimate, and $y_{i,c} \in \{-1, 1\}$ a class label which is 1 for all the observations in the class c and -1 otherwise (see, e.g., [1]). To model the possible linear shifts in the power features between the different measurement days, a dummy variable $x_{i,0} \in \{-1, 1\}$ indicating the recording day, was included in $\mathbf{x}_{i,k}$ as an additional predictor. A Gaussian prior $p(\mathbf{w}_{k,c}) = \mathcal{N}(\mathbf{0}, \sigma_w^2\mathbf{I})$ with a variance parameter σ_w^2 was assumed for the linear coefficients, and also a Gaussian prior $v_{k,c} \sim \mathcal{N}(0, \sigma_v^2)$ was set for the bias term.

Combining the likelihood of all the labeled observations $\mathbf{y}_c = \{y_{1,c}, ..., y_{n,c}\}$ from the both measurement days with the priors results in a conditional posterior distribution

$$p(\mathbf{w}_{k,c}, v_{k,c} |\, \mathcal{D}_{k,c}, \sigma_w^2, \sigma_v^2) \propto \left(\prod_{i=1}^{n} (1 + \exp(-y_i z_{i,k}))^{-1} \right) p(\mathbf{w}_{k,c}) p(v_{k,c}), \quad (2)$$

where $\mathcal{D}_{k,c} = \{\mathbf{y}_c, \mathbf{X}_k\}$, $\mathbf{X}_k = [\mathbf{x}_{1,k}, ..., \mathbf{x}_{n,k}]^{\mathrm{T}}$. Since the posterior distribution (2) is analytically intractable an approximative inference method is required. The Laplace approximation was chosen because it is computationally convenient for the logistic model (see, e.g, [1, 2]). In the Laplace approximation a multivariate Gaussian approximation

$$q(\mathbf{w}_{k,c}, v_{k,c}) = \mathcal{N}(\boldsymbol{\mu}_{k,c}, \boldsymbol{\Sigma}_{k,c})$$

is formed by doing a second order Taylor expansion for

$$\log p(\mathbf{w}_{k,c}, v_{k,c} |\, \mathcal{D}_{k,c}, \sigma_w^2, \sigma_v^2)$$

around the posterior mode. Point estimates for the parameters σ_w^2 and σ_v^2 were determined by optimizing the approximative log marginal posterior distribution $\log q(\sigma_w^2, \sigma_v^2 |\, \mathcal{D}_{k,c})$ obtained by approximating the log marginal likelihood, $\log p(\mathbf{y}_c \,|\, \mathbf{X}_k, \sigma_w^2, \sigma_v^2)$, with the Laplace's method as described in [2]. Relatively flat half-Student-t priors with scale 10 and degrees of freedom $\nu = 10$ were assigned for the variance parameters to prevent them from becoming very large.

From the posterior approximation $q(\mathbf{w}_{k,c}, v_{k,c})$, a Gaussian approximation is obtained for the latent values related to both the labeled and unlabeled input vectors for class c:

$$q(z_{i,k,c}) = \mathcal{N}(m_{i,k,c}, V_{i,k,c}), \quad (3)$$

where $m_{i,k,c} = \mathbf{x}_{i,k}^{\mathrm{T}}\,\boldsymbol{\mu}_{k,c}$, $V_{i,k} = \mathbf{x}_{i,k}^{\mathrm{T}}\,\boldsymbol{\Sigma}_{k,c}\,\mathbf{x}_{i,k}$, and one is appended to the feature vector $\mathbf{x}_{i,k}$ to account for the bias $v_{k,c}$. The expected values $m_{i,k,c}$

from all the C classes and K frequency bands as well as the dummy variable $x_{i,0}$ indicating the recording day were combined to form new 26-dimensional input vectors $\mathbf{m}_i = [m_{i,1,1}, m_{i,2,1}, ..., m_{i,K,C}, x_{i,0}]$ for a multi-class classifier.

Multi-class classification

Using the latent vectors $\mathbf{m}_i$ as new inputs, the type of the video stimulus was predicted using a nonlinear Gaussian process (GP) multi-class classifier with a squared exponential covariance function [2]. The softmax function was used to model the class probabilities according to

$$p(\mathbf{y}_i \,|\, \mathbf{f}_i) = \exp(f_{i,c}) \left(\sum_{j=1}^{C} \exp(\mathbf{y}_i^{\mathrm{T}} \mathbf{f}_i) \right)^{-1}, \tag{4}$$

where $\mathbf{f}_i = [f_{i,1}, ..., f_{i,C}]^{\mathrm{T}}$ is a vector of the latent function values related to data point i and $\mathbf{y}_i = [y_{i,1}, ..., y_{i,C}]^{\mathrm{T}}$ is the corresponding target vector which has entry one for the correct class for the observation i and zero entries otherwise. Following [2], independent zero-mean GP priors were placed for each class, that is, $p(\mathbf{f}_c \,|l_{\mathrm{se}}, \sigma_{\mathrm{se}}^2) = \mathcal{N}(\mathbf{0}, \mathbf{K})$, where $\mathbf{f}_c$ collect all the latent function values related to class c. The covariance matrix $\mathbf{K}$ is defined by the squared exponential covariance function

$$[\mathbf{K}]_{i,j} = k_{\mathrm{se}}(\mathbf{m}_i, \mathbf{m}_j \,|\theta) = \sigma_{\mathrm{se}}^2 \exp\left(-\frac{1}{l_{\mathrm{se}}^2} \sum_{l=1}^{d} (\mathbf{m}_{i,l} - \mathbf{m}_{j,l})^2 \right), \tag{5}$$

where $d = 26$, σ_{se}^2 is a magnitude parameter which scales the overall variation of the unknown function, and l_{se} is a length-scale parameter which governs how fast the correlation decreases as the distance increases in the input space.

Combining the likelihood of the observations $\mathbf{y} = \{\mathbf{y}_1, ..., \mathbf{y}_n\}$ with the priors $p(\mathbf{f}_c \,|l_{\mathrm{se}})$ results in an analytically intractable posterior distribution for the latent function values $\mathbf{f} = \{\mathbf{f}_1, ..., \mathbf{f}_n\}$, and again the Laplace approximation is used for approximate inference as described in [2]. The Laplace approximation results in a Gaussian posterior approximation for $\mathbf{f}$, and to approximate the predictive distribution it can be analytically combined with the conditional GP prior $p(\mathbf{f}_* \,|\, \mathbf{f}, \mathbf{m}, \mathbf{m}_*)$, where $\mathbf{m}$ collects the training inputs and $\mathbf{f}_*$ is a $C \times 1$ vector of latent values related to an unlabeled test input $\mathbf{m}_*$. Using the Laplace approximation also a marginal likelihood approximation $q(\mathbf{y} \,|\, \mathbf{m}, l_{\mathrm{se}}, \sigma_{\mathrm{se}}^2)$ can be obtained to determine point estimates of the parameters l_{se} and σ_{se}^2. However, optimiz-

ing the marginal likelihood resulted in a very small length scale and instead more conservative estimates $l_{\text{se}} = 2$ and $\sigma^2_{\text{se}} = 1$ were selected based on cross-validated predictive tests with the data from the second day. In practise, both the dimensionality reduction as well as the multi-class classification were implemented with the freely available GPstuff software package (`http://www.lce.hut.fi/research/mm/gpstuff/`).

Bibliography

[1] Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, (2006)

[2] Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press, (2006)