32nd Annual International Conference of the IEEE EMBS
Buenos Aires, Argentina, August 31 - September 4, 2010

In Search of More Robust Decoding Algorithms for Neural Prostheses, a Data Driven Approach

Erk Subasi, Benjamin Townsend, Hansjörg Scherberger

Abstract— **In the past decade the field of neural interface systems has enjoyed an increase in attention from the scientific community and the general public, in part due to the enormous potential that such systems have to increase the quality of life for paralyzed patients. While significant progress has been made, serious challenges remain to be addressed from both biological and engineering perspectives. A key issue is how to optimize the decoding of neural information, such that neural signals are correctly mapped to effectors that interact with the outside world - like robotic hands and limbs or the patient's own muscles. Here we present some recent progress on tackling this problem by applying the latest developments in machine learning. Neural data was collected from macaque monkeys performing a real-time hand grasp decoding task. Signals were recorded via chronically implanted electrodes in the anterior intraparietal cortex (AIP) and ventral premotor cortex (F5), brain areas that are known to be involved in the transformation of visual signals into hand grasping instructions. We present a comparative study of different classical machine learning methods with an application of decoding of hand postures, as well as a new approach for more robust decoding. Results suggests that combining data-driven algorithmic approaches with well-known parametric methods could lead to better performing and more robust learners, which may have direct implications for future clinical devices.**

I. Introduction

DRAWING on a wealth of knowledge about cortical movement processing, together with advances in signal processing and acquisition that have been made in recent years, the field of brain machine interfaces (BMIs) has the potential to become a viable assistive tool for patients with chronic spinal cord injury, stroke, and other motor debilitating diseases. It was demonstrated that 2D and 3D hand and arm location can be reconstructed from the activity of populations of M1 neurons in macaque monkeys [1]-[4] and monkeys can use these 3D control signals to operate a robotic arm in order to feed themselves [5]. Our group has been working on the development of a system for the specific decoding of hand grasping postures (Fig. 1). In contrast to other studies, our approach aims at decoding neural activity in the anterior intraparietal cortex (AIP) and ventral premotor cortex (F5), higher-order motor planning areas, where movements are represented more abstractly than in M1[6],[8],[10]. Signals from these areas were interpreted in real time by a decoding algorithm that predicted the intended hand movement. The predicted hand posture was then fed back visually to the animal in conjunction with a small juice reward that was given if the prediction matched the movement instruction (Fig. 1). In line with the abstract nature of the movement representation in premotor and parietal cortex, we make a discrete prediction of the final hand posture instead of predicting the full continuous trajectory of the hand and arm during reach-to-grasp.

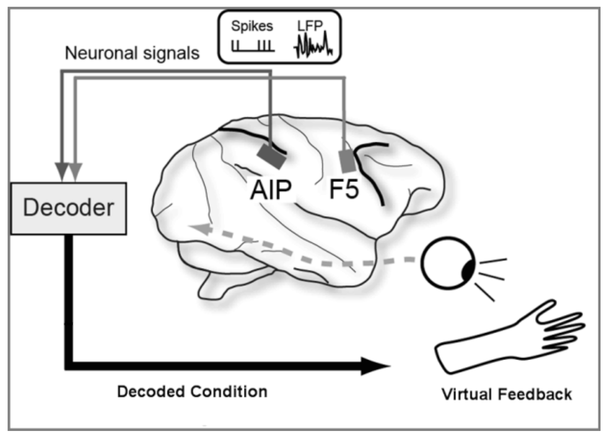

Fig. 1: Decoding of hand grasping signals in AIP and F5.

II. Methods

A. Experimental paradigm

Two macaque monkeys were trained in a delayed grasping task, where they first placed their hands at rest and fixated an LED before a handle was presented in one of 5 different orientations. The animal was instructed with the color of an additional LED to grasp the handle either with a power grip or a precision grip, and to withhold movement execution until the fixation LED dimmed. After a variable delay period (700-1100 ms), the animal was allowed to make a reach and grasp the handle. Correctly executed trials were rewarded with a small amount of juice.

Manuscript received April 1, 2010. This work was supported by the Neuroscience Center Zurich (ZNZ), the Swiss National Science Foundation (SNSF), and the NCCR Neural plasticity and Repair.

E. Subasi is with the Institute of Neuroinformatics, University Zurich / ETH Zurich, Winterthurerstrasse 190, 8057 Zurich, Switzerland (phone: +41-44-635-3067; fax: +41-44-635-3053; e-mail: erk@ini.phys.ethz.ch).

B. Townsend, was with the Institute of Neuroinformatics University Zurich / ETH Zurich. He is now with the Department of Microsystems Engineering (IMTEK), Laboratory for Biomedical Microtechnology, University of Freiburg, Georges-Köhler-Allee 102, 79110 Freiburg, Germany (e-mail: benjamin.townsend@imtek.de).

H. Scherberger was with the Institute of Neuroinformatics University Zurich / ETH Zurich and is now with the German Primate Center, Kellnerweg 4, 37077 Göttingen, Germany (e-mail: hscherberger@dpz.eu).

B. Setup

After successful training, floating micro-electrode arrays (MicroProbe Inc, Gaithersburg, MD, USA) with 16 monopolar electrodes were permanently implanted in AIP (2 x 16) and F5 (3 x 16) of a first animal (Z). In a second animal (S), we used 32-electrode arrays with similar specifications and implanted 2 arrays in each area. From these electrode arrays, we recorded action potentials (spikes) from many individual neurons while the monkey performed the grasping task. A typical recording session gave between 20-30 single- and multi-units in monkey Z and around 100 units in monkey S, which were significantly tuned to the parameters of the task (grip type or grip orientation, each tested by 1-way ANOVA, $p<0.05$). Most of these "units" comprised action potentials from several cells around the electrode that could not be assigned to individual neurons by online spike sorting, and were essentially treated as "multi-unit" signals. A minority of channels gave clearly isolated "single units" as evidenced by the inter-spike interval histogram and waveform homogeneity. We will examine the differences between encoding and decoding from single unit versus multi unit signals in a forthcoming report. Neural signals were sampled using a Cerebus Neural Signal Processor (Cyberkinetics Inc, Foxborough, MA) and streamed to separate decoding and recording computers. Spike sorting was conducted online by manually setting time-amplitude discrimination windows for animal Z, and using proprietary automated spike-sorting features of the Cerebus system for animal S.

C. Neural Coding Scheme

Our goal in this work was to compare different machine learning approaches on the same data set in an offline analysis. For the sake of simplicity, we used here only spike rates that were computed from online-sorted spike counts. However, one should note that other signal modalities like local field potential activity, as well as other encoding schemes like temporal coding, are likely to contain additional information that could be utilized in neural prosthetic applications.

III. MOTIVATION

A. Decoding Problem & Common Approaches

If we want to decode neural information optimally, a good point to start with is to ask: "How does the brain actually decode this information?" *Population coding* plays well with stochastic and distributed characteristics of neurons and matches some of the nice properties of motor cortical areas. First of all, population coding has some built-in noise compensation characteristics and is robust. It also gives rise to short-term memory in the system and can instantiate complex and non-linear functions [9]. An early-developed population coding mechanism is "population vector analysis" which is essentially a cosine function fit to the observed activity direction [7]. The population vector algorithm enjoyed substantial success until maximum likelihood estimation (MLE) approaches turned out to be superior in capturing the underlying probability distributions. Being a parametric method, MLE usually assumes Poisson statistics for motor neuron firing rates and for the parameter estimation to be tractable, scientists make the strong assumption that the individual neurons are firing independently. Thus, we reach Naïve Bayesian Classifiers, which are widely used in BMI community for discrete predictions.

B. Motivation

Naïve Bayesian Classifiers can be trained quite efficiently in a supervised setting and despite their non-realistic independence assumptions and parametric nature they perform very well in cortical signal decoding and are treated to date as state of the art in many settings. But frequently, analysis of multiple simultaneously recorded spike trains with these naive assumptions will raise the legitimate question whether the data is treated adequately. The absence of well-developed statistical methods for analyzing multiple point processes is the main concern for practitioners with classical statistics background. Here we address this issue from a data driven perspective, where we define the decoding goal as having the best prediction accuracy, without spending much effort on optimal modeling of the data source.

C. Two Cultures in Formal Predictions

There are two broad categories of approach towards generating predictions from neural data. The first, *"Data modeling approach"* utilizes an underlying model that is constructed a-priori to generate data. This approach relies heavily on parameter estimation techniques and model validation that is achieved usually by goodness-of-fit tests. However, one should be careful in the choice of estimators while keeping in mind that even significant results may be misleading if the assumptions regarding the initial model are not appropriate.

In contrast, the "*Algorithmic modeling approach"* requires no a-priori model for the underlying data generation. Instead, black-box learners work on all available data and validation is checked by predictive accuracy. One should be particularly careful about over-fitting when working with this set of algorithms. Being strong learners, this family usually provides better prediction performance as compared to model-based procedures, but they suffer from lack of explanatory power.

The main motivation of this work was to bring some of the well known methods from data modeling and algorithmic modeling together, first for the purposes of comparison and second to investigate a possible combined approach.

IV. RESULTS

To that extent, we created a testing-platform, using the open source JAVA package *RapidMiner [11]*, where different thoroughly tested algorithms can be plugged in and compared in the same setup in a standard way.

TABLE I

PERFORMANCE OF DECODING FOR, ANIMAL S, THE BEST 3 PERFORMANCES ARE HIGHLIGHTED FOR EACH COLUMN

Record IDs :	A0515	A0519	A0520	A0525	A0526	A0527	A0528	A0508	AVERAGE
Generic Algorithms									
NaiveBayes-Poisson	0.391	0.581	0.514	0.5	0.654	0.582	0.758	0.632	**0.577**
NaiveBayes-Gaussian	0.449	0.602	0.543	0.5	0.605	0.532	0.697	0.658	**0.573**
BayesianLogisticRegression	0.406	0.581	0.476	0.386	0.444	0.468	0.689	0.526	**0.497**
BayesNet-K2	0.449	0.398	0.39	0.341	0.37	0.481	0.409	0.5	**0.417**
DecisionTree	0.304	0.366	0.343	0.295	0.259	0.354	0.22	0.303	**0.306**
NaiveBayesTree	0.246	0.269	0.286	0.386	0.185	0.278	0.303	0.303	**0.282**
kNN	0.435	0.462	0.457	0.386	0.42	0.481	0.538	0.539	**0.465**
Perceptron	0.464	0.548	0.41	0.318	0.346	0.494	0.636	0.368	**0.448**
MultiLayerPerceptron	0.507	0.484	0.524	0.364	0.58	0.443	0.72	0.658	**0.535**
LinearSVM	0.464	0.548	0.667	0.341	0.63	0.506	0.705	0.632	**0.562**
RbfSVM	0.536	0.57	0.648	0.432	0.617	0.468	0.697	0.645	**0.577**
Ensemble Methods									
Adaboost-NaiveBayesPoisson	0.362	0.538	0.543	0.5	0.667	0.544	0.72	0.563	**0.555**
Adaboost-NaiveBayesGaussian	0.406	0.602	0.552	0.5	0.617	0.544	0.697	0.589	**0.563**
Adaboost-DecisionTree	0.304	0.344	0.343	0.273	0.235	0.367	0.205	0.321	**0.299**
Adaboost-kNN	0.391	0.462	0.39	0.318	0.395	0.38	0.53	0.457	**0.416**
Adaboost-Perceptron	0.333	0.505	0.333	0.318	0.37	0.38	0.606	0.458	**0.413**
Adaboost-LinearSVM	0.478	0.581	0.476	0.341	0.519	0.392	0.682	0.502	**0.496**
BayesianBoosting-NaiveBayesPoisson	0.391	0.581	0.514	0.5	0.654	0.582	0.75	0.602	**0.572**
BayesianBoosting-DecisionTree	0.449	0.409	0.362	0.295	0.284	0.342	0.402	0.398	**0.368**
BayesianBoosting-LinearSVM	0.478	0.581	0.476	0.341	0.519	0.392	0.682	0.533	**0.500**
MultiBoosting-NaiveBayesGaussian	0.493	0.602	0.543	0.455	0.58	0.532	0.697	0.578	**0.560**
MultiBoosting-DecisionTree	0.319	0.516	0.486	0.432	0.457	0.405	0.477	0.404	**0.437**
Proposed Methods									
DecisionTree - NB & SVM	0.507	0.634	0.623	0.477	0.605	0.582	0.742	0.645	0.602
Neural Network - NB & SVM	0.522	0.624	0.592	0.477	0.593	0.544	0.758	0.645	**0.594**

TABLE II

PERFORMANCE OF DECODING FOR, ANIMAL Z, THE BEST 3 PERFORMANCES ARE HIGHLIGHTED FOR EACH COLUMN

Record IDs :	B0206	B0208	B0214	B0523	B0626	B0729	B0731	B0828	AVERAGE
Generic Algorithms									
NaiveBayes-Poisson	0.317	0.336	0.360	0.244	0.298	0.331	0.261	0.335	**0.310**
NaiveBayes-Gaussian	0.331	0.353	0.377	0.225	0.316	0.344	0.261	0.341	**0.318**
BayesianLogisticRegression	0.245	0.387	0.333	0.206	0.237	0.318	0.228	0.293	**0.281**
BayesNet-K2	0.266	0.395	0.360	0.169	0.202	0.248	0.152	0.263	**0.257**
DecisionTree	0.129	0.387	0.228	0.225	0.246	0.248	0.272	0.281	**0.252**
NaiveBayesTree	0.216	0.193	0.158	0.163	0.167	0.242	0.250	0.192	**0.197**
kNN	0.317	0.294	0.289	0.225	0.228	0.293	0.272	0.311	**0.279**
Perceptron	0.295	0.429	0.289	0.200	0.289	0.261	0.207	0.269	**0.280**
MultiLayerPerceptron	0.338	0.328	0.368	0.256	0.246	0.338	0.304	0.317	**0.312**
LinearSVM	0.302	0.294	0.325	0.250	0.237	0.306	0.272	0.269	**0.282**
RbfSVM	0.345	0.353	0.368	0.269	0.246	0.331	0.283	0.359	**0.319**
Ensemble Methods									
Adaboost-NaiveBayesPoisson	0.295	0.420	0.325	0.244	0.298	0.293	0.293	0.335	**0.313**
Adaboost-NaiveBayesGaussian	0.259	0.403	0.298	0.225	0.316	0.293	0.250	0.341	**0.298**
Adaboost-DecisionTree	0.151	0.345	0.237	0.213	0.263	0.261	0.272	0.263	**0.251**
Adaboost-kNN	0.245	0.092	0.281	0.094	0.096	0.248	0.174	0.281	**0.189**
Adaboost-Perceptron	0.216	0.387	0.289	0.225	0.254	0.287	0.098	0.240	**0.249**
Adaboost-LinearSVM	0.252	0.403	0.289	0.244	0.263	0.287	0.293	0.311	**0.293**
BayesianBoosting-NaiveBayesPoisson	0.288	0.353	0.307	0.256	0.316	0.331	0.261	0.257	**0.296**
BayesianBoosting-DecisionTree	0.288	0.328	0.377	0.238	0.254	0.306	0.261	0.317	**0.296**
BayesianBoosting-LinearSVM	0.252	0.403	0.289	0.225	0.254	0.287	0.293	0.305	**0.289**
MultiBoosting-NaiveBayesGaussian	0.353	0.370	0.368	0.250	0.281	0.331	0.293	0.323	**0.321**
MultiBoosting-DecisionTree	0.259	0.387	0.333	0.294	0.316	0.306	0.304	0.287	**0.311**
Proposed Methods									
DecisionTree - NB & SVM	0.327	0.378	0.351	0.276	0.325	0.338	0.279	0.377	0.331
Neural Network - NB & SVM	0.295	0.328	0.368	0.238	0.289	0.350	0.283	0.353	**0.313**

TABLE III

VARIANCE OF DECODING PERFORMANCE AMONG RECORDING DAYS

	Animal Z	**Animal S**
NaiveBayes-Poisson	0.00160	0.01232
RbfSVM	0.00218	0.00861
DecisionTree - NB & SVM	0.00152	0.00685

In particular, we tested: Naïve Bayes classifiers with Poisson and Gaussian data assumptions, Perceptrons, Decision Trees, Logistic Regression Classifiers, k-Nearest Neighbor classifiers, Naïve Bayes trees, Multi Layer Perceptrons (classic back-propagation neural networks), and Linear / Radial Basis Function (RBF) Support Vector Machines (SVM). We also implemented some widely used ensemble method approaches, given the tremendous attention they have drawn in the last decade. Referring to our discussion on two cultures for statistical modeling, all our non-Bayesian algorithms were closer to the algorithmic modeling family. Moreover, Naïve Bayesian classifier with a Poisson distribution assumption is our benchmark because of its frequent use.

Due to space constraints, it is not feasible to provide implementation details of all learners here. However, as a rule of thumb, we did not extensively optimize parameters of individual learners, but always tried to stay in a reasonable parameter space and run simple optimization routines on out-of-sample data where applicable. The reported results are decoding accuracies out of ten conditions with 10-fold cross validations. We present decoding results with 24 different learning methods from 8 recording sessions from each animal (Table I, Table II). There is a first set of 11 algorithms, which is selected from a wide variety of widely used machine learning methods. Another set of 11 algorithms are constructed using ensemble methods which utilize some learners from the previous set as their base learners. Finally, we propose two additional algorithms in an attempt to combine data modeling and algorithmic modeling approaches for obtaining more robust classifications.

Looking at the last columns of Table I and Table II, to the average decoding accuracies, the first thing to notice is that there is a substantial difference for both animals (0.60 vs. 0.33 for best performing classifiers). This is not surprising looking at Fig.2. Recordings from Animal S are significantly richer in terms of the number of tuned units.

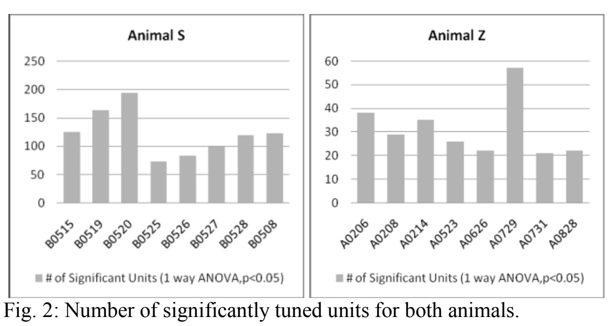

Fig. 2: Number of significantly tuned units for both animals.

Further analyzing the tables, we observe that Naïve Bayes share the top rank with RBF-SVMs among generic algorithms. From ensemble methods MultiBoosting using Naïve Bayes as base learner performs best especially for the second animal. Finally, the proposed combined approach (Decision Tree – NB & SVM) beats both the simple algorithms and ensemble methods in terms of average decoding accuracy.

V. DISCUSSION

It is important to note that Naïve Bayes classifiers are among the best performers. In addition, some strong classifiers do not perform as well, especially ensemble methods that utilize Naïve Bayes classifiers as base learners. This brings us to the conclusion that the Poisson firing rate model assumption is indeed close to reality and Naïve Bayesian classifiers are already doing well in capturing population coding characteristics of motor cortical areas. Furthermore, SVMs perform identical to Naïve Bayes in terms of average performance while they both show different characteristics in individual recordings. Thus, one can speculate that once population correlations increase or the firing characteristics of individual neurons deviate from Poisson, a strong data driven method like SVM might outperform Naïve Bayes classifiers. Motivated with this line of thinking, we proposed a two-level classifier that attempts to combine the best features of both approaches. In the base level both SVMs and Naïve Bayes classifiers learn the data independently. At the top level, there is a final decision maker, either a decision tree or neural network, that has access to the outputs of both classifiers and to the input data. This approach indeed turned out to be the best performing classifier in our analysis. Furthermore, it also had minimum variance among the best performing approaches (across daily sessions), i.e. it is a more robust learner (Table III). Thus, we believe that by providing an average higher performance more robustly, 2-level classifiers may have important implications for future clinical applications.

VI. REFERENCES

[1] E. E. Fetz, “Operant conditioning of cortical unit activity”, *Science* 163: 955-958, 1969

[2] Serruya, M. D., Hatsopoulos, N. G., Paninski, L., Fellows, M. R., and Donoghue, J. P., 2002, "Brain-machine interface: Instant neural control of a movement signal," *Nature*, 416(6877), pp. 141-142.

[3] Taylor, D. M., Tillery, S. I., and Schwartz, A. B., 2002, "Direct cortical control of 3D neuroprosthetic devices," *Science*, 296(5574), pp. 1829-1832.

[4] Hochberg, L. R., Serruya, M. D., Friehs, G. M., Mukand, J. A., Saleh, M., Caplan, A. H., Branner, A., Chen, D., Penn, R. D., and Donoghue, J. P., 2006, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature*, 442(7099), pp. 164-171.

[5] Meel Velliste, Sagi Perel, M. Chance Spalding, Andrew S. Whitford and Andrew B. Schwartz, “Cortical control of a prosthetic arm for self-feeding”, *Nature*,doi:10.1038/nature06996, 2008

[6] Musallam, S., Corneil, B. D., Greger, B., Scherberger, H., and Andersen, R. A., 2004, "Cognitive control signals for neural prosthetics."*Science*, 305(5681), pp. 258-262.

[7] Georgopoulos AP, Lurito JT, Petrides M, Schwartz AB, Massey JT (1989) Mental rotation of the neuronal population vector. *Science* 243: 234-236

[8] Markus A. Baumann, Marie-Christine Fluet, and Hansjörg Scherberger, “Context-Specific Grasp Movement Representation in the Macaque Anterior Intraparietal Area” *The Journal of Neuroscience*, 2009, 29(20):6436-644

[9] Alexandre Pouget, Peter Dayan, Richard Zemel, “Information processing with population codes”, Nature Reviews Neuroscience 1, 125-132 (November 2000)

[10] Townsend B, Subasi E, Scherberger H, “Real time decoding of hand grasping signals from macaque premotor and parietal cortex.”13th Annual Conference of IFES Society “From Movement to Mind” Freiburg, Germany: Biomedical Engineering, 53 supp. 1. (2008)

[11] RapidMiner, http://rapid-i.com/