How much does movement and location encoding impact prefrontal cortex activity? An algorithmic decoding approach in freely moving rats

by

Adrian Lindsay

BCS, University of British Columbia, 2013
BSc, University of British Columbia, 2008

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in
The Faculty of Graduate and Postdoctoral Studies
(Neuroscience)

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)

April, 2018

© Adrian Lindsay, 2018
Abstract

Specialized brain structures encode spatial locations and movements, yet there is growing evidence that this information is also represented in the rodent medial prefrontal cortex (mPFC). Disambiguating such information from the encoding of other types of task-relevant information has proven challenging. In order to determine the extent to which movement and location information is relevant to mPFC neurons, tetrodes were used to record neuronal activity while limb positions, poses (i.e. recurring constellations of limb positions), velocity and spatial locations were simultaneously recorded with two cameras every 200ms as rats freely roamed in an experimental enclosure. Regression analyses using Generalized Linear Models revealed that over half of the individual mPFC neurons were significantly responsive to at least one of the factors and many were responsive to more than one. On the other hand, each factor accounted for only a very small portion of the total spike count variance of any given neuron (<20% and typically <1%). Artificial Neural Networks and Random Forests were used to analyze ensemble activity and revealed that ensembles were usually superior to the sum of the best neurons in encoding movements and spatial locations. These results indicate that movement and location information have a small impact on any individual mPFC neuron, yet because these weak signals are so widely distributed across the population, this information is strongly represented at the ensemble level.
Lay Summary

It is often difficult to determine exactly what is driving changes in the activity of single neurons in the rodent medial prefrontal cortex (mPFC). Basic things such as movements or spatial locations can activate these neurons and interfere with the ability to extract task-related information. Here we applied a series of powerful techniques to carefully quantify the relationship between a rat’s movements and the activity of these neurons. Overall, the firing related to limb movements, poses and spatial locations had little impact on individual neurons, yet this widely distributed information became prominent at the ensemble, or group, level. A framework where all neurons participate to varying degrees in the encoding of all events has certain advantages that could prove useful for higher-order cognitive processing and in guiding artificial intelligence approaches.
Preface

Training and surgery for experimental subjects was carried out by Barak Caracheo and Jamie Grewal. Electrophysiology recording sessions were carried out by Barak Caracheo and Jamie Grewal as part of a Pavlovian conditioning task (Caracheo et al., 2015), although the included data was recorded during the pre-task period. The video recording apparatus was designed and built by me.

I designed and wrote the annotation and analysis software, with use of software packages referenced in the materials and methods section, with refactoring and UI improvements made by Daniel Leibovitz. I annotated the recorded video with assistance from Daniel Leibovitz.

Analysis was designed and carried out by me. A version of the main section of this paper, and the associated figures, has been submitted for publication. All other work unless specified is original to the author and this document. All procedures were conducted in accordance with the Canadian Council of Animal Care and approved by the Animal Care Committee of the University of British Columbia (Certificate A14-0084).
# Table of Contents

Abstract ................................................................................................................................. ii

Lay Summary ......................................................................................................................... iii

Preface ...................................................................................................................................... iv

List of Figures ........................................................................................................................ vi

Acknowledgments .................................................................................................................. vii

Introduction ............................................................................................................................. 1

Materials and Methods .......................................................................................................... 2
  Experimental Subjects and Operant Chamber ....................................................................... 2
  Surgery and Electrophysiology Data Acquisition .................................................................. 2
  Software and Acquisition of Video Data ................................................................................ 4
  Data Analyses ........................................................................................................................ 5

Results ....................................................................................................................................... 8
  Single Unit Analysis ............................................................................................................... 9
  Ensemble encoding of individual limb positions .................................................................... 11
  Ensemble encoding of poses ................................................................................................. 12
  Ensemble encoding of spatial position .................................................................................. 13

Discussion .............................................................................................................................. 16

References ............................................................................................................................... 19
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Multi-camera video annotation / Electrophysiology</td>
<td>3</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Implants / Histology</td>
<td>4</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Model Architectures</td>
<td>7</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Relationship between firing and the rats’ body movements and location</td>
<td>8</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Single neuron location mappings</td>
<td>9</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Proportions of neurons significant on given movement/position factors</td>
<td>10</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Ensemble vs single neuron encoding of body positions</td>
<td>11</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Predictions of body positions</td>
<td>13</td>
</tr>
<tr>
<td>Figure 9</td>
<td>Predictions of spatial locations</td>
<td>15</td>
</tr>
</tbody>
</table>
Acknowledgments

I would like to acknowledge the support of my supervisor, Dr. Jeremy Seamans, and my committee in the pursuit of this work. I would also like to acknowledge the work of Barak Caracheo and Jamie Grewal designing and running experiments and data collection, as well as Daniel Leibowitz for his work in refactoring software and annotating data.

This work was supported by funding from the Canadian Institutes of Health Research (MOP-93784 and MOP-84319).
Introduction

Prefrontal cortex (PFC) neurons encode a wide variety of stimuli, actions and outcomes and many neurons multiplex information across domains (Rigotti et al., 2013; Duncan, 2001; Lapish et al. 2008; Jung et al., 1998). The fact that these neurons can encode such diverse types of information greatly complicates the ability to make attributions about what a neuron might be responding to at any given time. A particularly notable example is the encoding of information during working memory tasks; while it is clear that some rat medial PFC (mPFC) neurons fire throughout the delay period of such tasks (Baeg et al., 2003; Laubach et al., 2000; Hyman et al., 2010; Euston et al., 2012; Cowen et al. 2012; Wang et al. 2011; Durstewitz et al. 2000), questions remain about the information represented by this activity. Delay-period activity has been linked to the prospective tracking of likely trial outcomes (Hyman et al., 2012; Myroshnychenko et al., 2017), the encoding of reward-related feedback from preceding trials (Laubach et al., 2015) or the active tracking of spatial locations, which could be highly relevant since most rat working memory tasks are spatial in nature (Jung et al., 1998; Laubach et al., 2015; Euston and McNaughton, 2006). Based on a series of careful analyses, Euston & McNaughton (2006) and Cowen & McNaughton (2007) argued that delay-period activity can also be attributed to the encoding of the paths traversed or the movements rats engage in during delay periods. These latter studies raised the broader question of how much of the activity recorded on any task is actually movement or location related.

To gain a deeper perspective on this issue we performed a detailed video analysis of body and limb movements in rats implanted with tetrode arrays aimed at the mPFC. The goal was not to explore the absolute limits of movement or location encoding of mPFC neurons but rather to get a realistic picture of how much movement or location encoding affects firing in a typical task situation. In order to approximate a task situation while avoiding overt task correlates that could interfere with the ability to cleanly extract movement or location signals, rats were trained on a simple Pavlovian conditioning task, but no tones or outcomes were delivered when the data were collected. The first analysis involved a regression through a Generalized Linear Model (GLM) that assessed the contribution of various movement factors (the inputs) to the observed firing of individual neurons (the outputs). Because movement information may not be completely contained in the firing of individual mPFC neurons taken one at a time, we also performed several different types of ensemble analyses. For these analyses, the firing rates of all neurons were used to predict the rat’s spatial location, the position of individual limbs or constellations of limb positions (i.e. postures or “poses”). Since it was unlikely that all neurons contributed equally to all of these factors, the Random Forest (RF) algorithm was employed; it was able to iteratively consider random subsets of neurons in order to select the best candidates based on a measure of mutual information between firing and a particular factor. Several Neural Network (NN) models were also used that considered relationships between small groupings of neurons across a range of time scales (temporal lags and leads). Results indicated that spatial location and movement information was distributed across the population but accounted for at best <20% of firing rate variance in those neurons maximally sensitive to a particular factor.
Materials and Methods

Code Repository Links: Video Annotation - https://github.com/Loken85/Lab_Video_Annotator
Analysis - https://github.com/Loken85/ephys_ML_repo

A note on terminology: The fields of neuroscience, machine learning, and statistical methods make regular use of a number of technical terms, many of which are overloaded between the fields. To avoid confusion, we have defined these terms when necessary in the text, and here highlight two particular cases. The term “ensemble” is regularly used in neuroscience to refer to groups of neurons, while in statistics “ensemble methods” are a class of algorithms which aggregate results across many instances of learners (such as the Random Forest algorithm utilized in this paper). We use the term in the neuroscience sense throughout. Similarly, we use “neurons” to refer to individual cells in our electrophysiology recordings, but the field of machine learning, in particular, neural networks and deep learning, use the term “neuron” to refer to a single component or unit in a neural network. In this paper “neurons” refer to recorded cells from the rat brain, and the term “unit” is used to refer to the building blocks of our neural networks.

Experimental Subjects and Operant Chamber

Four male Long-Evans rats (Charles River Laboratories, Montreal) weighing between 400 and 470g were used. They were housed in an inverted 12-h day/12-h night cycle and were food restricted to 90% of their free-feeding weight but given unlimited access to water for the duration of the experiment. All procedures were conducted in accordance with the Canadian Council of Animal Care and approved by the Animal Care Committee of the University of British Columbia. Recording sessions took place inside a custom made behavioural chamber (30cm x 25cm x 60cm) built for a Pavlovian conditioning task (Caracheo et al., 2015) though the presented data were recorded while the rats moved freely in the chamber in the pre-task period in the absence of any tones or outcomes.

Surgery and Electrophysiology Data Acquisition

Rats were surgically implanted with a custom built 16-tetrode hyperdrive array (Hyman et al 2012; Ma et al 2016). They were anesthetized under iso-flurane gas, their skull was surgically exposed and a 4mm by 3mm hole was drilled. The dura was removed to expose the brain around coordinates +3.0mm from Bregma and ±0.5mm from the midline. The tetrode microdrive implant was positioned over the area and fixed to the skull with 11 skull screws and dental acrylic. Two additional screws were used as ground wires and were placed in the posterior skull. Tetrodes were lowered ~1000μm on the day of surgery and then the rats were given 1–2 weeks of recovery. Tetrodes were advanced up to 1000μm more to their target prior to the first recording session. In between sessions, tetrode drives were turned between 20 – 50μm to maximize the units recorded and obtain different populations. Based on tetrode advancement records, the positions were estimated to have been mostly in the medial wall, within the anterior cingulate cortex (ACC) up to the border of the prelimbic (PL) cortex. Tetrodes were attached to EIB-36TT boards, plugged into two HS-36 headstages and connected via tether cables to a Digital Lynx 64-channel system and then to a PC workstation. Electrophysiological data and behavioral events were captured using Cheetah 5.0. Files were exported into Offline Sorter (Plexon, Inc.) and were manually sorted based on three-dimensional projections of wave form peaks, valleys and principal components (Figure 2b). Once cells had been sorted, they were exported to Neuroexplorer 4 (Nex Technologies) and then to Matlab (Mathworks) for further analysis. When the experiments ended,
rats were perfused and their brains collected and sliced on a cryostat. Slices were mounted on slides and viewed under a microscope to confirm the anatomical locations of tetrode tracts (Figure 2c).

Figure 1: A) Multi-camera Video Annotation. A frame from the annotation set showing a rat moving freely in the operant chamber. The coloured dots on both views serve as indicators for marked limbs and body parts. The annotations from both camera views are combined to produce an accurate 3D location of each tracked body part for each frame. The user can select the camera that provides the most accurate view of a limb and the xyz coordinates of the limb are simultaneously captured in both data streams. This software does not automatically detect these points, but they are set by the user in each frame. B) Spike sorting based on waveforms recorded from a single tetrode. Collections of waveforms (top and left) from 9 single units isolated from one tetrode of a 16-tetrode array from a single recording session. Each trace contains the signal from the 4 wires of the tetrode. Clouds of firing events (bottom right) from each of the single units plotted in 3-D space. The dimensions of the space are relative peak to valley distances across 3 of the 4 tetrode wires.
**Software and Acquisition of Video Data**

Cameras were placed horizontally to record a side-on view and below to record a bottom-up view. The two video streams were synchronized and co-registered and imported into a custom, purpose-built python package for video annotation. This software allowed us to annotate body and limb positions on a frame-by-frame basis for both camera views (Figure 1a). The frame rate was set to match the binning used for the neural spike count matrix (200ms). The camera that provided the best view of a limb was used to score the limb position in each frame. The central point of the skeleton was set to the centre point of the rat’s body. The x-y-z location of this point gave the spatial location of the animal at each time bin. Changes in the x-y-z coordinates across time bins were used to calculate the animal’s velocity. The positions of 9 body parts (head, mid-shoulder, front left limb, front right limb, pelvis, rear left limb, rear right limb, base of the tail, mid-point of tail) were calculated relative to the centre point of the body. Since the two camera views were co-registered, the body positions could be collapsed to 9 values/time bin. Therefore, each time bin was associated with a spatial location (the xyz coordinate of the centre of mass), 1 instantaneous velocity value and 9 body positions (relative to centre of mass) that were temporally aligned with an N item spike count vector (N= number of neurons). This alignment was
set by paired pulses to the video and electrophysiology recording rigs, matching the timing to within 1ms.

Data Analyses

The annotated positions were broken down into three data sets for each session: (1) Individual body positions as defined by the xyz coordinates of the body relative to the centre of mass of the animal. (2) The velocity of the animal at each point in time. (3) The location in 3D space of the animal within the enclosure.

A note on algorithm design and selection: The complicated, non-linear relationships between the firing of mPFC neurons and observable behavioural factors present an opportunity for the application of a number of complex analysis techniques, which can provide a more complete description of encoding over traditional statistical analysis. This power in part, stems from these algorithms ability to represent a larger range of possible relationships between inputs and outputs, as well as from aggregate learning and exploiting randomisation. For example, training a neural network (fully-connected network) searches the space of differentiable functions to find the relationship between these inputs and outputs. Recurrent networks take this a step further, defining a series of functions through time, making learning a search over possible program space. This can be expanded again, by stacking network layers on top of each other (deep learning), granting the network a vast space of functions and scales with which to represent the relationship. However, this representational power comes at a price. These complicated networks are more likely to suffer from overfitting (even relatively small networks are capable of memorizing large data sets) and these networks have a huge number of parameters to search over; This makes learning extremely time-consuming and data intensive (additional parameters expand the dimensionality of the search space). These downsides can be mitigated by appropriate algorithm selection and by making informed assumptions about the nature of the data and the structural relationships between and within groups of inputs in the space. The models presented in this paper follow this approach (this is covered in more detail in the discussion below). It is important to note, that while these models are not necessarily the “optimum” or best possible performers on these problems, they are the product of careful selection and considerable architecture and parameter optimisations and illustrate various means to identify the relationships between neurons and output factors. For more detail and examples, refer to the readme and documentation provided in the linked analysis code repository.

Generalized Linear Model (GLM): The GLM is a generalized form of linear regression which allows for response variables with non-normal error distributions. This allows us to model firing response as a Poisson sequence with a logarithmic link-function to our factors; Essentially this is a basis-function restricted estimator with the assumption that firing rates can be modeled as a Poisson sequence (Poisson distributions are described by a single parameter, lambda = emission or occurrence rate for a given time period, and are a reasonable approximation of the time-binned firing behaviour of neurons in this part of the brain). The GLM is setup as follows: predictors \( \eta \) are linear combinations of parameters \( \eta = X\beta \), linked to the mean \( \mu \) of the output by the link function \( \mu = \ln(\mu) \). As formulated, each GLM makes a maximum-likelihood estimation of the relationship between a single input factor and the firing rate of a particular neuron. Regressions which provided a significant relationship between a factor and firing rate \( p < = 0.01 \) denoted the neuron as responsive to that factor. The GLM was run on each factor independently for each neuron. Possible correlation confounds between individual neurons that are responsive to a given factor were accounted for by calculating their
partial correlation (The partial correlation between two variables, \(X\) and \(Y\), on some controlling variable \(Z\), written as \(p_{X,Y,Z}\), is defined as the correlation between the residuals \(e_x\) and \(e_y\) resulting from the regression of \(X\) with \(Z\) and \(Y\) with \(Z\).

**Random Forest (RF):** RF analysis was conducted using the python sci-kit learn package (Pedregosa et al., 2011). The RFs were tuned for forest size and split size using “out-of-bag” (OOB) error across all sessions. OOB error is the error rate on OOB samples and is calculated as follows: Each individual tree in the RF is trained on a subset of the total training set. Samples from outside this subset, or bag, which are unseen during training by that tree, are used to validate each tree. The average across all trees is referred to as OOB error. The RFs were trained on a per session basis using a normalised, balanced (by class), random subset of time bins (by cross-validation) and then tested by prediction on the remainder of the time bins. The parameters with the best average performance for each factor across all sessions were reported. The RF architecture schematic is shown in Figure 3a.

**Neural Networks (NNs):** The networks were coded using the Keras package for python (Chollet et al., 2015), which utilized Theano (Al-Rfou et al., 2016) to perform tensor operations. Our networks were run with GPU parallelisation using nVidia’s CUDANN package (Nickolls et al., 2008). A multi-layer convolutional neural network (CNN) was used to evaluate the regression between the multi-dimensional input (neuron spike count data) and output (relative limb positions) space, as described in Figure 8b. The network was comprised of several convolution layers of rectified linear (relu) units, with a mean-squared error loss (between predicted (y-hat) and the actual (y), limb position as follows: \(SE = \frac{1}{2N} \sum_{i=1}^{N} ||\hat{y}_i| - y_i||^2\). The network was trained on a per session basis using a normalised, random subset (80%) of time bins with 10% held out as a validation set and then tested by prediction on the remaining 10% to generate the reported results. The Recurrent Neural Network (RNN)(Graves, 2016) was used for spatial position encoding. After structural testing and optimisation, the multi-layer RNN evaluated the relationship between time series of the multidimensional input (the binned firing rates) and output (the spatial location factor) space, as described in Figure 8c. It considered not only the current time bin but the 4-6 time bins that preceded and/or followed it to better inform its decisions about the spatial location of the rat. The network was comprised of layers of long-short-term memory units (LSTMs), a modification on the basic recurrent structure which minimized the impact of numerical instability during back-propagation (Hochreiter and Schmidhuber, 1997; Gers et al., 2000). The output layer was a softmax layer (Bishop, 2006; LeCun et al., 1998), trained using categorical cross-entropy as the loss function (this measured the cross-entropy between two distributions; an estimated distribution, \(q\), and the true distribution, \(p\), computed as follows: \(H(p, q) = -\sum_x p(x) \log(q(x))\)). As above, the network was trained on a per session basis using a normalised, class-balanced, random subset of time bins, with 10% held out as a validation set and was then tested by prediction on the remaining 10% to generate the reported results. One major concern with unconstrained predictors with many parameters like NNs is the tendency to overfit, particularly on small data sets like the one considered here (Hastie et al., 2009). We utilised an effective method to curb this overfitting by preventing complex co-adaptations, or paired relationships between units in the network from forming with a technique called drop-out whereby connections between individual units were dropped with some specified probability during training (Srivastava et al., 2014; Gal and Ghahramani, 2016). We formulated the problem as a categorical prediction to further combat overfitting, to compensate for uneven distributions of locations in the data and to disambiguate location from relative distance; each location was technically treated as equidistant from every other location, so that the algorithm learned the true location correspondence,
as opposed to some amalgamation of averaged distances. The enclosure was divided into two 4x4 grids stacked on top of one another. The position for each time bin was set as the cube containing the rat’s center of mass. The training set data was always balanced for cube occupancies. In all cases our NN were trained and evaluated using cross-fold validation; This randomisation procedure combats unintended effects on performance due to evolving distributions in the data. Reported results are averaged across folds for a given session.

Figure 3: Model architectures. A) RF Architecture. Diagrammatic representation of the RF utilised for ensemble regression/classifications. Input in the form of binned spike counts was introduced at the bottom of the trees and progressed through each tree to the top-layer output. The output layer consisted of the class votes of the all the individual trees in the forest. During training, the trees were constructed decision layer by decision layer. At each layer of each tree, a random subset of the inputs (neurons) were selected. A single input was then selected from this group based on maximal mutual information with the output and used to define a single decision. This process continued until the classes were uniquely separated in each tree. The validation error for training was done by out-of-bag testing on the constructed trees. B) CNN Architecture. Diagrammatic representation of the CNN utilised for ensemble regression on body part position. Input in the form of binned spike counts entered the network at the bottom and progressed through the computational layers to the output layer at the top. The network contained two convolutional layers (yellow boxes) of 64 filters, each of which was 4 to 7 x 1 units in size. The output of the first of these layers is condensed (orange boxes with purple edges) by maximum pooling before being passed to the second convolutional layer. This output layer is flattened (orange boxes with blue edges) before being passed to a final layer of fully connected output units which provide the predicted body/limb position signals. During training, the output of the convolutional layers was subject to connection dropout to reduce overfitting and improve performance. C) RNN Architecture. Diagrammatic representation of the RNN utilised for ensemble categorization of spatial locations. Input, in the form of a fixed length time series of binned spike counts (5 x 200 ms bins, or 1s in length), entered the network at the bottom and progressed through the computational layers to the output layer at the top. The network contained three long-short term memory (LSTM) recurrent layers, each with 32 units. The first two layers (blue boxes with purple outlines) are fully connected and passed the full 5 time-steps on to the next layer. The third layer (blue boxes with red outlines) only passed the last step in the time series on to the final output of the layer, which consisted of 32 fully connected units with Softmax activation to provide the predicted location. During training, the connections between the LSTM layers were subject to dropout to reduce overfitting and improve performance.
Results

Figure 1a provides a screen-shot of the custom written application used to track spatial locations and body positions throughout each of the sessions. The central point of the rat’s body (the xyz coordinate of the midpoint between shoulders and pelvis) was taken as the spatial location of the animal at each time bin. Changes in this point across time bins were used to calculate animal’s velocity. The positions of 9 body parts (head, mid-shoulder, front Left limb, front right limb, pelvis, rear left limb, rear right limb, base of the tail, mid-point of tail) were calculated relative to the centre point of the body. Consequently, each time bin was associated with one spatial location, one instantaneous velocity value and 9 body position values that were temporally aligned with an N item spike count vector (N= number of neurons).

Figure 4: Relationship between firing and the rats’ movement and spatial locations. The coefficient of determination (R2) between the firing rate of individual neurons and each of the A) body position factors, B) the general movement/velocity factor and C) the spatial location factor. Only the R2 values of the top 50 (A) or top 120 (B-C) neurons significant (p<0.01) on a given factor are plotted. Total significant neurons of each factor were: Head=181, Shoulders=193, Left Front limb=170, Right Front limb=170, Pelvis=194, Left Rear limb=179, Right Rear limb=157, Tail Base=173, Tail Mid=124.
Single Unit Analysis

The database for the present study was derived from 8 recording sessions in 4 rats and contained a total of 492 neurons. The first analysis involved a Generalized Linear Model (GLM). For each neuron, the input, or predictors to this GLM were the animal’s current spatial location, instantaneous velocity or limb positions while the spike counts comprised the output. Across all factors, on average, 38% of the neurons were significant on any single factor. 61% of the neurons attained significance on at least one body position factor (Figure 4a). However, for those neurons which showed significant responses, the R² between the model factor and the firing rate vector was relatively low (<0.1; Fig 6a)
such that any of these factors accounted for at most, 10-20% of a neuron’s session-wide firing rate variance. Only 26% of the neurons were found to be significant on the velocity factor and in none of these cases were high $R^2$ values observed (Figure 4b) (this could be due, in part, to the small enclosure size). As a result, the velocity factor was not analyzed further. In total, 134 neurons were significant on the spatial location factor, but individually none of these neurons were particularly good at accurately encoding spatial location and $R^2$ values over 0.1 were found in only a few cases (Figure 4c). Figure 5 shows the firing maps of two neurons with relatively high $R^2$ values (0.15 and 0.08) on the spatial location factor. In spite of attaining statistical significance, their firing was diffuse but variable across the enclosure.

Most neurons tended to be multi-responsive as 42% of limb responsive neurons were responsive to spatial location and 41% to body movement while 13% were responsive to all three factors (Figure 4b). This high degree of overlap is somewhat misleading as the factors were not independent in that they provided different perspectives on the same moving object. This multi-collinearity could not be avoided and was the reason the model was run independently for each factor. As a consequence, it was difficult to parse the relative contributions of the factors to the overall firing rates of the neurons. Average correlations between individual neurons within sessions were less than 1% ($p <= 0.01$), and average partial correlations between neurons regressed on individual body position factors, the velocity factor, and the location factor were all less than 1% ($p <= 0.01$).

![Figure 6: Proportions of neurons significant on a given movement/position factor. A) The percentages of neurons (out of a total of 492) which had a significant response ($p<0.01$) to movement of a given body part (left), overall velocity (middle) or spatial location (right). B) Venn diagram of neurons responsive to the three categories of factors and their overlap. Blue = number of neurons.](image-url)
neurons responsive to at least one of the body position factors, Green = number of neurons responsive to velocity, Magenta = numbers of neurons responsive to spatial location.

**Ensemble encoding of individual limb positions**

The ensemble analysis of body movements was formulated as a regression problem using the Random Forest (RF) algorithm with the binned spike counts of all neurons as inputs and the body/limb position factors as outputs. To facilitate comparisons with the single neuron data, encoding performance was evaluated by calculating an ensemble $R^2$ relative to the model factors. The RF did a reasonable job of predicting the animal’s relative head (Figure 7a) or limb positions and $R^2$ values for the ensembles were relatively high on most factors (Figure 7b) with an overall average of 0.36. Direct comparisons to single units within the same session (as shown in Figure 7b) highlighted the superiority of the ensemble, as $R^2$ values derived from the RF were usually higher than the sum of the $R^2$ values derived from the GLMs of all the significant individual neurons. While the ensembles were better overall, part of this improvement stemmed from the fact that the RF selectively considered only the informative portion of a given neuron’s activity with regards to a particular factor.

![Figure 7: Ensemble versus single neuron encoding of body positions. A) An example of the actual head position across 180 frames/time bins (36s) (dotted red line) versus the predicted head position (blue line) derived from the RF. B) The overall average (and s.d.) $R^2$ values between the RF-predicted and the actual body/limb positions are plotted alongside the ordered single neuron $R^2$ (red bars) values from GLMs run on a single session.](image)
Ensemble encoding of pose

The single neuron and ensemble analyses considered each body/limb position factor in isolation, yet it seemed unlikely that mPFC neurons would be precisely tuned in this way to single limbs. Therefore, we considered whether mPFC ensembles encoded ‘poses’ or constellations of body/limb positions. Rearing would for instance be a well-known pose created by a recurring constellation of relative limb position values, although we refrained from giving such labels to the poses (limb positions were continuous variables, yielding infinite possible arrangements). The RF was again used for this analysis but in a slightly different way; the forests first learned the relationships between ensemble activity and possible poses and were then used to predict poses (i.e. provide a probability that the limbs were in certain positions) in unseen data based only on ensemble activity. The negative mean-squared error (NMSE) was used to evaluate the difference between the body/limb positions that were predicted from ensemble activity versus those that were observed in the test data. In addition to the RF, a convolutional neural network (CNN; Figure 8b) was also used that attempted to find the relationships between distinct poses and unique activity patterns in sets of neurons taken 4-7 at a time. The CNN in essence treated each neuron as a partial feature detector and learned the activity patterns in local groups of these feature detectors that were predictive of constellations of body/limb positions.

We found that both the RF and CNN were able to provide reasonable predictions of learned poses in the unseen test data. Despite some variance across the sessions, the average NMSE was significantly below what was obtained when the time bins were shuffled (Figures 8a,b) or when the actual spike counts for each neuron were substituted with spike counts generated by random poisson-processes with identical means (not shown). To provide context for these NMSE values, Figures 8c (right and left) show examples of the degree to which the poses predicted by the RF differed from the actual poses. In these examples the size of the spheres indicates the MSE of the predicted body/limb positions depicted.
Figure 8: Predictions of body positions. A) The overall mean squared error (MSE) between the true and predicted positions of all 6 pose components used in the RF analyses. Blue bars are test MSEs, yellow bars are the MSEs when the firing rate matrix was shuffled independently of the output poses. B) Same as A, but for the CNN. (C) Two images of the animal taken from single video frames in session 07J were overlaid by coloured circles representing the MSE in the positions of the body/limbs positions as predicted from the RF analyses. Each circle is centered at the actual position of the body part as originally scored using the software shown in Figure 1, while the radii of the circles denote the MSE in the RF-predicted positions (dot colors: head = white, front right paw = red, front left paw = blue, rear right paw = dark red, rear left paw = dark blue, tail = pink). The overall average MSE across all body parts depicted in the images were 69.7 (left) and 122.4 (right). Only the MSE spheres for those body parts that can be clearly seen in the example images are depicted here.

Ensemble encoding of spatial position

We next attempted to predict the location of the animal within the enclosure using a similar approach. It would be unrealistic to expect that the ensembles could predict exact locations on the scale of individual pixels, so the operant chamber was divided into 32 cubes (2 stacked 4 x 4 grids). As in the analyses above, we first tried using the RF (Figure 9a) and found that it could provide reasonable predictions about the spatial location of the rat based on the ensemble activity. While the quality of spatial location encoding was variable both within (i.e. depending on which subsets of time bins were used) and between sessions, accuracy was always higher compared to chance performance assessed by
shuffling of the spatial location assignments in each time bin of the original data (Figure 9b). The ensembles exhibited a nearly 50% accuracy rate for determining the location of the rat even though the chance rate was on average 1/32 or 3%. In fact, these accuracy measurements were somewhat conservative as they were calculated from the strict maximum of the probabilistic output and instances where the prediction probabilities were split across two or more cubes were counted as errors. The CNN performed statistically no better than the RF and therefore the results from the CNN were not shown.

Previously, Euston and McNaughton (2006) found that some mPFC neurons were sensitive to the trajectory the rat took through a task enclosure. For this reason, we also considered whether the location predictions could be improved by incorporating trajectory information as the rat entered or exited a specific location. Based on the observation that the ensembles carry information about the rat’s spatial location (Figure 9a), we reasoned that trajectory information should be contained in the spike count time series. A multi-layer Recurrent Neural Network (RNN)(Graves, 2016; Figure 2c) was chosen for this purpose that evaluated the relationship between time series of multidimensional inputs (the binned spike counts) and outputs (the spatial locations) across 1s epochs (i.e. 5 x 200ms time steps). During training, spike counts from the present time bin as well as the 4 time-bins that flanked it were used to update the weights between the units in the RNN. An example of location output prediction for the RNN is shown in Figure 9c while accuracy measures for each session are shown in Figure 9d. Similar to the RF, the predictions made by the RNN were probabilistic across possible locations, so results were conservative measures of maximum probabilities. The predictions were significantly better for the RNN than the RF in 5 of the sessions (accuracy ranges measured by cross-validation are non-contiguous, marked by * in Figure 9d). This improvement was realized because the RNN learned which cubes would likely be visited based on the evolution of ensemble, thereby constraining the probable output space.
Figure 9: Predictions of spatial locations. A) Spatial location predictions generated by the RF for unseen test data across a 10s period (50 video frames / time bins). The true location is indicated by a black dot, the circles indicate correct location predictions and Xs indicate incorrect location predictions. This figure contains only the subset of 4 locations (i.e. cubes, ordered on the y-axis) the animal happened to visit during the chosen time period. B) The location accuracy per session for the RF classifiers. Blue bars = accuracy of predictions for the test data, Yellow bars = accuracy of predictions when position assignments were shuffled independently of the firing rate matrix. C-D) Same as A-B, but for the RNN.
Discussion

In the present study, we precisely tracked 9 points on a freely moving rat’s body every 200ms in the absence of an overt task and then quantified the relationship between this body position data and the firing of multiple single neurons recorded simultaneously in the mPFC. While a significant relationship was found between firing and the rat’s movements or spatial locations, the relationship was uniformly quite weak for individual neurons. The ensemble analyses exploited the signals of the best neurons and as a result returned better results. The superior performance of the RNN over the other ensemble algorithms in predicting spatial location indicated that additional information was contained in the time series of firing activity that evolved as the rat moved through the enclosure.

Jung et al (1998) performed the first detailed study of mPFC correlates and described more than 20 different movement correlates as well as the broad spatial tuning properties of these neurons. Euston and McNaughton (2006) and Cowen and McNaughton (2007) later highlighted that movement/location signals and putative cognitive signals can be difficult to disambiguate. Expanding on this work, we sought to provide a detailed assessment of the precise degree to which movements or locations affect the firing of mPFC neurons in a typical task setting. We found that the majority of neurons were responsive to at least one of the 11 factors (9 body position factors, a general velocity factor and a location factor) while many exhibited significant responses to more than one factor. This might be expected given that PFC neurons tend to be multi-responsive. However, the more important consideration is that the factors all described various aspects of the same moving body (e.g. changes in body position were partially informative of changes in spatial location and vice versa). In spite of the large proportion of neurons that showed significant responses to any single body part, their individual contributions were small and even the activity of the best single neurons was insufficient to accurately decode the position of a single limb. Based on our analyses, we would conclude that most mPFC neurons are weakly responsive to a wide range proprioceptive, posture and position information.

It should be noted that the apparatus and environment were not designed specifically to optimize movement or location encoding; a more repetitive task, performed over many days would likely more strongly entrain neurons. The enclosure was also quite small, and it is possible that exploration of a larger, or novel enclosure would have evoked stronger neural responses. Furthermore, the recordings were performed outside the context of an overt task (but in a typical task environment) so as to preclude non-movement correlates from interfering with the analysis. While there are obvious advantages to this approach, movement signalling in mPFC is highly plastic and action correlates can be altered if the action leads to a motivationally relevant outcome (Kennerley et al 2009; Ma et al 2014). It is also important to bear in mind that while movement signaling was weak in any individual neuron, this information was highly distributed across the population. As a result, it is not possible to assign strict movement correlates to mPFC neurons by analyzing the responses of single neurons in a single, isolated situation. The present results do not give the absolute strength of movement or spatial encoding in mPFC neurons but rather provide a picture of how such factors could interfere with putative correlates recorded in operant tasks.

Ensembles generated more complete descriptions of movement and location encoding than single neurons, although the level of improvement depends to some extent on how one sets up the analyses. One consideration is whether to treat all neurons equally or to weigh certain neurons (or
some portion of the firing range of certain neurons) more than others. In the present study, this selection process was automated in an informed manner by the RF algorithm. Generally speaking, an RF is a collection of decision trees which decides on an output using a process of aggregate voting and selection of maximum probability (Breiman, 2001). RFs have an advantage in unconstrained problem spaces in that they are resistant to overfitting because the estimate quickly approaches the expectation of the distribution as the number of estimators (trees) grows (Hastie et al., 2009). Each branching point of a tree in the forest considers only a randomly selected subset of the input dimensions (in this case neurons) and selects the candidates from that subset based on a measure of mutual information with the output (in this case segments of a limb position or spatial location vector). Therefore, each decision tree could be viewed as a weighting of neurons on a given factor (or portion of a given factor) with the forest providing an aggregate of single neuron mappings to all the model factors. In the GLM analyses, the firing rate of each neuron was correlated (via a link function) to a limb position vector across all time bins. By contrast, the RF created trees with neurons responsive to portions of a limb position vector and these trees biased the forest consensus which in turned helped to make the RF predictions more accurate.

Another advantage of the ensemble analysis was that it allowed us to search for the encoding of constellations of co-occurring limb positions that we referred to as poses. The CNN (Dumoulin and Visin, 2016) searched for functional relationships within small groups of neurons, taken 4-7 at a time. Each neuron was treated as a partial feature detector, and patterns in local groups of feature detectors encoded specific portions of a given pose. The complete pose was encoded by the combined information across many such subgroups of neurons. Even though the relationship between neural activity and the poses were statistically significant, both of our predictive algorithms work agnostically of pre-defined constellations of body positions and as such the output or “pose” space does not necessarily yield separation into meaningful or behaviourally relevant poses. While it is theoretically possible, at present we cannot comment on how well they reflected well-known behaviors, such as grooming, rearing etc.

Ensembles were also superior to single neurons for spatial decoding. While the RF performed well, adding complexity to the model, with a CNN, did not result in significant improvements. However, adding the additional temporal dimension utilized in the RNN was superior to single neurons, the RF, and the CNN. This was likely because the animals were moving from one location to another and by utilizing the time series information, the RNN limited the range of possible future or past locations on which to base its predictions. Interestingly, improvements from temporal signals were not observed in the decoding of pose (the RNN did not perform better than the RF or CNN), as one might expect. This type of time series-based encoding would complement the real-time tracking of movement trajectories by individual mPFC neurons as identified by Euston & McNaughton (2006). As a result, the mPFC could in theory use knowledge about where the rat has been, where it is going and the path taken between these two points to inform ongoing decisions.

The finding that ensembles performed better than the best single neurons is not surprising and is consistent with past studies conducted on neurons from various brain areas (Foldiak and Young, 1995; Shadlen and Newsome, 1998; Carmena et al., 2005; Riehle et al., 1997; Laubach et al., 2000). One reason the individual neurons performed poorly in the present study was their inherent unreliability in that they did not respond the same way every time a limb was in a particular position. While this is disadvantageous when considering each neuron in isolation, it may be beneficial for ensemble encoding.
We have observed that a given mPFC neuron can respond to a certain task element on one trial, only to respond to another task element or not at all on the next trial (Ma et al., 2016). But the non-responsive trials tend not to be shared across the population so that when one neuron drops out another fills in, thereby maintaining a constant level of overall responsivity in the network (Ma et al., 2014; 2016). In the field of computer science, the phenomenon of ‘drop-out’ has proven to be an advantageous feature for neural-network based machine learning algorithms. In this case ‘drop out’ occurs when individual connections are eliminated with some probability during the training of networks, as was the case for our CNNs and RNNs. This drop-out helps to suppress the formation of strict dependencies between individual units and a particular output feature. This prevents overfitting and overspecialisation (Srivastava et al., 2014; Gal and Ghahramani, 2016) and makes the network better able to generalise across tasks. Although direct comparisons between biological brains and “neural” network models should be made with caution, it may be that frontal cortex networks use their inherent unreliability for similar purposes. Specifically, multi-responsivity coupled with high trial-to-trial variability ensures that the neurons do not become entrained to specific events. On the other hand, because the variability or ‘drop out’ is not synchronized across the population, the ensembles always maintain an accurate representation through time. As discussed above, most neurons were minimally responsive to most of the factors and this may have extended to all neuron if the recording periods had been extended or the ‘significance’ criteria had been slightly relaxed. Since past studies have reported an almost innumerable array of other frontal cortex neuron correlates, it may be that all frontal cortex neurons are at least minimally responsive to all relevant events. Like the superior performance of the RF algorithm, frontal cortex ensembles may produce coherent and consistent representations via an aggregate voting process across groups of highly variable and unreliable neurons. These properties may be what gives the frontal cortex the flexibility required to respond to both the similarities and differences inherent in complex, ever-changing environments. Endowing artificial neural networks with similar properties could conceivably expand their flexibility and functionality as well.
References


Repository Links

Lab Video Annotator Repository: https://github.com/Loken85/Lab_Video_Annotator

Electrophysiology Machine Learning Repository: https://github.com/Loken85/ephys_ML_repo