Encoding Models In Neuroimaging

Fabian A. Soto[1] and F. Gregory Ashby[2]

[1]Florida International University, USA
[2]University of California, Santa Barbara, USA

Contents

1 Introduction **1**

2 Voxel-Based Encoding Models **3**
2.1 Encoding Model 4
2.2 Measurement Model 7
2.2.1 Aggregating Channel Responses 8
2.2.2 Linking Neural Activation to the BOLD Response 10
2.2.3 Dynamic Encoding Models 13
2.3 Population Receptive Fields 16
2.4 Feature Spaces and Model Interpretation 18

3 Model Inversion **21**
3.1 Population Response Reconstruction 23
3.2 Stimulus Decoding and Reconstruction 26

4 Representational Similarity Analysis **27**
4.1 Estimating an RDM 28

5 Testing Encoding Models Against Behavioral Data **29**
5.1 Encoding/Decoding Observer Models 29
5.2 Model-Based FMRI 32
5.3 Joint Neural and Behavioral Modeling 33

6 Conclusions **34**

7 Related Literature **35**

8 Acknowledgments **35**

1 Introduction

One of the greatest barriers to progress in mathematical psychology is model mimicry. In almost every domain of cognitive modeling, there are competing models that assume qualitatively different perceptual and cognitive processes, yet are able to mimic the behavioral predictions of each other. One reason for this is that although competing models may make very detailed predictions about

psychological processes, historically those processes have been unobservable and, as a result, the models are tested only against crude dependent measures, such as response accuracy and response time.

Within the past few decades, a wide variety of new neuroimaging technologies have been developed that allow levels of observability into human brain function that seemed unimaginable when many currently popular mathematical models in psychology were first proposed. Included in this list are functional magnetic resonance imaging (fMRI), positron emission tomagraphy (PET), magnetoencephalography (MEG), functional near-infrared spectroscopy (fNIRS), electrocorticography (ECoG), and high-resolution electroencephalography (EEG). Although these methods all have limitations, they nevertheless have the potential to allow unprecedented observability into the perceptual and cognitive processes predicted to underlie competing mathematical models of perception and cognition. As a result, testing models against neuroimaging data in addition to the more traditional response accuracies and response times offers an exciting possible solution to the model mimicry problems that plague mathematical psychology.

Despite their promise, neuroimaging data are infrequently used to test mathematical models of the type that are common in mathematical psychology. There are several reasons for this. First, neuroimaging is still a relatively new technology and neuroimaging data analysis is still in a period of rapid development. Second, all of these neuroimaging technologies were developed outside of mathematical psychology. Third, most models in mathematical psychology make few, if any neuroscience predictions. At first glance, the latter of these reasons seems the most limiting, but in fact, several data analysis methods that were developed to analyze fMRI data can be used to test models that make no neuroscience assumptions. Included in this list are *model-based fMRI* and *representational similarity analysis* (RSA).

All neuroimaging technologies work in a similar way. In all cases, recordings are collected at discrete times and locations in the brain while the subject is engaged in some perceptual or cognitive task. The recordings are directly (e.g., ECoG, EEG) or indirectly (e.g., fMRI, PET) related to neural activation. The spatial resolution varies. ECoG can sometimes measure action potentials in single neurons, whereas each EEG electrode is influenced by millions of neurons. Temporal resolution also varies, with ECoG, EEG, and MEG at one extreme (with resolutions near 1 ms) and PET at the other (with resolutions of 5 – 10 sec). State-of-the-art functional MRI scanners, with multi-band slice acquisition, have a temporal resolution of about 500 ms and a spatial resolution of 1 – 2 mm (i.e., which is limited by the point-spread function of the blood-oxygen-level dependent, or BOLD response; Fracasso et al., 2021).

In general, neuroimaging data analysis techniques can be classified as either *encoding* or *decoding* methods. Encoding methods use knowledge of the experimental design and stimuli to build a model that predicts the neural activation that should be generated at each recording site on every trial. Decoding methods refer to approaches that make inferences in the opposite direction – that is, they use the observed recordings to make predictions about stimuli and other events in the experiment (Haynes and Rees, 2006; Naselaris et al., 2011; Norman et al., 2006; Pereira et al., 2009). The idea is that if a brain region of interest (ROI) responds differently to two different stimulus attributes then that ROI might be processing those attributes differently. The most widely used decoding method is known as pattern classification or even more commonly as multi-voxel pattern analysis (MVPA).

Encoding models are similar to traditional models in mathematical psychology. To model behavior in a task, a mathematical psychologist will typically combine assumptions about the underlying perceptual and cognitive processes with knowledge of the task to write equations that predict the participant's accuracy and/or response time. To build an encoding model, assumptions about the underlying neural processes are combined with knowledge of the task and the type of neuroimaging technique being used to write equations that predict values of the dependent variable that is measured at each recording site. For example, an encoding model of fMRI data would predict

the observed BOLD response at each voxel in response to each stimulus presentation. Forward inferences of this type are used for two primary purposes. First, they can be used to identify brain regions that are sensitive to specific attributes of the stimulus events. For example, when natural scenes are described by the outputs of many phase-invariant Gabor filters, simple fMRI encoding models accurately predict the BOLD response in early visual areas, but not in high-level areas of visual cortex (Kay et al., 2008; Naselaris et al., 2009). In contrast, when the same scenes are described using semantic category labels, encoding models accurately predict activation in high-level visual areas but not in early visual cortex (Mitchell et al., 2008; Naselaris et al., 2009). Second, encoding models can be used to test theories of cognitive and neural processing. If a theory accurately describes the cognitive and neural processing that occurs during a specific task, then it should be possible to use that theory to construct an encoding model that accurately predicts the dependent variables recorded in a set of pre-specified ROIs.

Because these two goals are somewhat different, it is not surprising that a diverse set of encoding models have been proposed (e.g., Ashby, 2019). The most widely used fMRI encoding model is the general linear model (GLM), which is used most commonly to identify brain regions that are sensitive to the simplest possible attribute of a stimulus event – namely, its presence or absence. All other encoding models are more ambitious. Arguably the next most popular fMRI encoding approach is dynamic causal modeling (DCM), which identifies a candidate set of brain regions that mediate event processing, along with all of their functional interconnections (Ashby, 2019; Friston et al., 2003). DCM is also more complex than other encoding models, partly because it uses a nonlinear model relating the BOLD response to neural activation and partly because it uses a variational Bayesian approach for model selection.

The vast majority of encoding models were developed to be tested against fMRI data. Even so, for the most part, the models can all be applied to any neuroimaging technology. The only significant difference from one technology to another is in the interface that converts predicted activation in a neural population to values of the dependent variable that the technology measures. For example, in the case of fMRI data, one needs to model the transformation from neural activation to the BOLD response recorded in fMRI experiments. With EEG data, one needs to include a head model that accounts for electromagnetic properties of the head and of the sensor array. But in all cases, the model of each neural population and of how the population activations are combined is roughly the same. However, because the models were developed for application to fMRI data, we will assume an fMRI application in the rest of this chapter. For most of the chapter, this just means that we will refer to a recording site as a voxel, and the time between recordings as the TR (repetition time; the amount of time it takes the scanner to measure BOLD responses from all voxels in the brain). Except for this nomenclature, the only part of the chapter unique to fMRI is discussed in the subsection entitled "Linking neural activation to the BOLD response," which considers the interface between the neural activations predicted by the models and the dependent variable most commonly measured in fMRI experiments.

2 Voxel-Based Encoding Models

Encoding models fall into two general classes: those that were constructed specifically to analyze fMRI data, and models that were originally designed for other purposes. The former class are often called *voxel-based encoding models*. The latter class can take many forms – from purely cognitive models of the type that are common in mathematical psychology to models with considerable biological detail (a branch of modeling called computational cognitive neuroscience; e.g., Ashby 2018). FMRI data are used along with a variety of other data types to test and refine these models. The process of testing such models against fMRI data is known as model-based fMRI. We consider model-based fMRI later in the chapter. This section describes voxel-based encoding models.

Voxel-based encoding models encompass a variety of different models, but they all share enough

features to be characterized within a single framework. As we will see in this section, all current voxel-based encoding models include an encoding model that predicts how every hypothesized neural population responds to each stimulus, and a measurement model that first transforms neural population responses into aggregate neural activity and then into values of the dependent variable being measured (e.g., the fMRI BOLD response). While most encoding models include a highly nonlinear transformation from stimulus to neural response, the measurement model is usually linear, and such models are often referred to as linearized encoding models. This means that most voxel-based encoding models can be seen as instances of linear regression with basis functions (Hastie et al., 2009).

2.1 Encoding Model

Encoding models begin with a mathematical description of the relation between a set of stimuli S_i, with $i = 1, 2, ...N_s$, and the response of a neural channel r_c, with $c = 1, 2, ...N_c$. Neural channels can represent either a single neuron or a population of neurons with similar properties, with the latter option being more common in the computational neuroimaging literature. Most encoding models assume that the channel response depends on the identity of the stimulus S_i, certain channel tuning parameters, various state variables, and properties of the neural noise. The tuning parameters, which are collected in the vector $\boldsymbol{\theta}$, include for example, constants that determine the channel's maximum possible response, and its preferred stimulus. The state variables, collected in the vector $\boldsymbol{x}$, include other variables that could affect the channel response, including for example, the responses of other channels in the population. Given these definitions, the standard approach is to first define the mean channel response

$$\mathrm{E}\left[r_c|\mathrm{S}_i\right] = f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right), \tag{1}$$

where E denotes expected value, and f_c is the channel tuning function, which is specified as part of the model. Tuning functions are discussed in more detail below, but it is important to note that in many applications, the alternative encoding models that are tested against data are identical, except for their tuning functions.

Most encoding models assume that channels operate in the presence of noise, but they differ in how that noise is modeled. One approach is to assume that the response of channel c to presentation of stimulus S_i equals

$$r_c(\mathrm{S}_i) = f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right) + \epsilon_c, \tag{2}$$

where ϵ_c is zero-mean noise (e.g., Pouget et al. 2000). A common choice is to assume Gaussian noise with some fixed variance. Note that this model predicts that the variance of the channel response does not change as the mean response increases. There is support for this assumption in channels that include a large population of neurons (Chen et al., 2006), but in single neurons, the variance of the spike count tends to increase in proportion to the mean (e.g., Tolhurst et al. 1983). Therefore, the fixed-variance Gaussian model is most appropriate when modeling channels of many neurons. A popular approach to modeling channels in which the variance of the response increases with the mean is to assume that r_c is Poisson distributed with mean $f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right)$ (e.g., Zemel et al. 1998). Therefore, this model assumes that the channel response has probability density function

$$P[r_c|\mathrm{S}_i] = \frac{f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right)^{r_c} \mathrm{e}^{-f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x})}}{r_c!}. \tag{3}$$

Because the variance of a Poisson distribution equals its mean, this model predicts that the variance of the channel response increases with the mean response. Note that Eqs. 2 and 3 both assume that the mean channel response satisfies Eq. 1.

Most models include multiple channels, each described by a version of Eqs. 1 and 2 or Eqs. 1 and 3, and which are combined into a random vector of responses $\boldsymbol{r} = [r_1, r_2, ..., r_{N_c}]$ that describe the response of all N_c channels to the presented stimulus. This is known as a population encoding model (Pouget et al., 2000, 2003), and $\boldsymbol{r}$ is usually referred to as a *population response*. In particular, voxel-based encoding models assume that every voxel includes a mixture of various populations of neurons, and that each population is tuned to a different attribute of the stimulus. The populations are commonly referred to as channels. For example, the most primitive visual encoding model might assume that each population or channel is tuned to a Gabor patch of a certain spatial frequency and orientation. But the populations could be tuned to anything. At the opposite extreme, they might be tuned to semantic category labels, such as rock, ocean, table, chair, or lamp. Voxel-based encoding models are most commonly used to identify brain regions that are sensitive to these attributes, so it is not unusual to build multiple encoding models for the same data that are each sensitive to a different set of stimulus attributes.

We can make this more concrete with an example of what has been termed the standard model of dimension encoding (Pouget et al., 2000, 2003). This model is typically restricted to applications in which the stimuli vary on a single dimension. Suppose the numerical value of stimulus S_i on this dimension is s_i. The model assumes Gaussian tuning functions, so in this case it predicts that

$$f_c\left(s_i, \boldsymbol{\theta}_c, \mathbf{x}\right) = r_c^{max} \exp\left[-\frac{1}{2}\left(\frac{s_i - s_c}{\omega_c}\right)^2\right], \tag{4}$$

where r_c^{max} represents the maximum response for channel c, s_c represents the value of the channel's preferred stimulus (i.e., the value of the stimulus that produces the channel's largest response), and ω_c represents the width of the tuning function. Many applications assume that all tuning functions have the same width (i.e., $\omega_c = \omega$, for all c), which is known as the homogeneous standard model. In all versions of the model, however, the channel tuning parameters are gathered together in the vector $\boldsymbol{\theta}_c^\top = [r_c^{max}, s_c, \omega_c]^\top$, where $\top$ denotes transpose. Note that in this case, the state vector $\boldsymbol{x}$ is empty. Also note that this model makes it possible to predict the mean channel responses as soon as the stimuli are selected, and therefore, before data collection begins.

Figure 1a shows the tuning functions of a large collection of channels from a typical application of this standard one-dimensional model. Note that all channels have identical shape ($r_c^{max} = r^{max}$ and $\omega_c = \omega$) and that the preferred stimuli for the various channels are evenly spaced on the stimulus dimension ($s_c = s_{c-1} + k$, for some small constant k). The shape of the tuning functions for all channels is therefore characterized by a single canonical tuning curve.

Now imagine presenting a specific stimulus S_i to the model and recording the response of all N_c channels in the population response vector $\boldsymbol{r}$. A convenient way to describe these responses is via a *population response plot*, in which neural responses are plotted on the ordinate and the numerical values of each channel's preferred stimulus are plotted on the abscissa. Figure 1b shows the population response of the model in Figure 1a to a stimulus with value 0. Each solid dot shows the response of a channel (where color matches the corresponding tuning function in panel a) in the absence of noise, and each open dot denotes a possible response of the same channels in the presence of noise.

Note that, because all channels have the same width and are equally spaced on the stimulus dimension, the expected population response has the same shape as the canonical tuning function. This property of the standard encoding model is a continuous source of confusion for both experimentalists and modelers, who sometimes confuse population response plots with tuning functions in their interpretation of encoding models. A population response function with the same shape as the canonical tuning function is not a general property of encoding models, but arises specifically from the homogeneous model (i.e., in which all tuning functions are identical, except for their preferred stimulus).

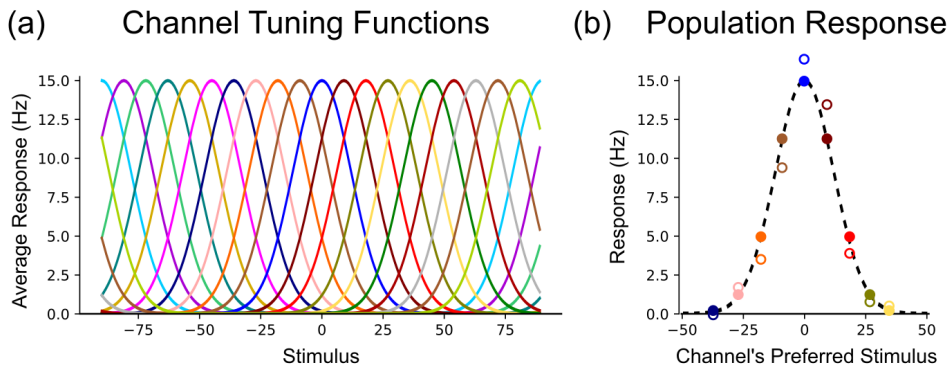

Figure 1: The standard model of dimension encoding. Panel (a) shows the tuning curves of the various channels included in the model. The peak of each tuning curve is centered at the channel's preferred stimulus value. Panel (b) shows the population response plot of this model on a hypothetical trial when a stimulus with value 0 is presented. Each solid dot shows the response of the channel of the matching color in the absence of noise, and each open dot denotes a possible response in the presence of noise.

Channel noise distributions have been estimated empirically, and there is evidence that humans use knowledge of this uncertainty during perceptual decision making (Van Bergen et al., 2015). Even so, it is common in the cognitive neuroscience literature to find applications in which channel noise is not modeled, with responses being described simply by Eq. 1. Within the general framework presented here, those applications implicitly assume Eq. 2 and Gaussian noise with a variance that is invariant across channels. When channel noise is modeled, a common assumption is that the noise is independently and identically distributed across multiple channels. In contrast, as mentioned earlier, some approaches model the channel response as Poisson distributed (i.e., as in Eq. 3), which scales the noise variance up with the mean channel response.

Of course, there are a variety of ways to construct more complex models. First, the model is easily extended to multidimensional stimuli. For example, in vision research it is common to represent images as two-dimensional matrices of pixel values, with each channel's tuning function being defined in that space. Many models represent the operation of primary visual cortex, or V1, through a large population of channels in which the tuning function of each channel is a Gabor wavelet tuned to a certain specific spatial location, orientation, and spatial frequency (e.g., Kay et al., 2008; Naselaris et al., 2009). In their structural encoding model, Naselaris et al. (2009; Kay et al., 2008, see also) assumed a total of 10,921 such channels.

The Gabor wavelet model of tuning functions is based on years of research on the response properties of neurons in V1. The tuning properties of channels in higher visual areas are less well understood. As a result, in applications that depend on a participant's perceptual or cognitive impressions of a set of images, a more generic tuning function might be more appropriate. The Gaussian tuning function of Eq. 4 is easily generalized to any arbitrary multidimensional stimuli. For example, consider a set of stimuli that vary on multiple dimensions and a channel in which the preferred stimulus is S_c. Then a multidimensional analog of Eq. 4 assumes that the channel response to stimulus S_i equals

$$f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right) = r_c^{max} \exp\left[-\frac{1}{2}\left(\frac{\Delta(\mathrm{S}_i, \mathrm{S}_c)}{\omega_c}\right)^2\right], \tag{5}$$

where $\Delta(\mathrm{S}_i, \mathrm{S}_c)$ is the distance in perceptual space between the representations of stimuli S_i and

S_c. Equation 5, which is an example of a radial basis function (e.g., Buhmann 2003), is a popular method for modeling the receptive fields of sensory units in many different modeling approaches (e.g., Ashby et al. 2007; Kruschke 1992).

A second approach to building a more complex model is to express channel tuning via a composite function: $f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right) = g_{c2}\left[g_{c1}\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right)\right]$. For example, in the Naselaris et al. (2009) model, the channel response is determined by applying a compressive nonlinearity to the output of the Gabor wavelet. If we denote the response of Gabor wavelet c to image S_i as $g_c(\mathrm{S}_i)$, then according to this model the response of channel c is

$$f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right) = \log[g_c(\mathrm{S}_i) + 1]. \tag{6}$$

The +1 just ensures that the channel response is never negative. Because log is a negatively accelerating function, this transformation models response compression at the neural level.

A third common generalization of the standard model is to assume that the channel response depends on state variables indexed in the vector $\boldsymbol{x}$. For example, $\boldsymbol{x}$ might include the responses of other channels in the population. In this case, a popular approach is to use these other responses to normalize the response of each channel:

$$f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}\right) = \frac{g_c\left(\mathrm{S}_i\right)^\nu}{\kappa^\nu + \sqrt{\sum_j \alpha_j [g_j\left(\mathrm{S}_i\right)]^\nu}}. \tag{7}$$

This is called *divisive normalization*, and it is an ubiquitous computation in cortical circuits (Carandini and Heeger, 2012). In this model, the channel response is normalized by a weighted sum of the response of all channels. The weights α_j represent the level to which other channels suppress the response of channel c, ν increases competition between channels for activation, and κ prevents division by zero.

2.2 Measurement Model

The encoding models discussed so far describe activity in each channel. However, in most applications, the individual channel responses are assumed to be unobservable. For example, in applications to fMRI, the BOLD response recorded in each voxel is assumed to be a mixture of many channel responses. Therefore, to test encoding models against empirical data, a model interface is required that specifies how the channels combine to determine the value of the dependent variable of interest (Van Bergen et al., 2015, see). This interface is called the *measurement model*.

The measurement model must solve two separate problems. First, even with state-of-the-art high resolution MRI scanners, each voxel includes many neurons, and therefore presumably, many different neural channels. Therefore, the first problem is to model how the various hypothesized channels combine to determine the amplitude of the neural activation that drives the BOLD response in each voxel.

Second, in the encoding models considered so far, the channel response $r_c(\mathrm{S}_i)$ is a single value that is presumed to represent the amplitude of neural activation in channel c when stimulus S_i is presented. In contrast, the BOLD response recorded from each voxel when stimulus S_i is presented is a time series that persists for 30 sec or so and depends in a complicated way on concentrations of oxygenated and deoxygenated hemoglobin, cerebral blood flow, and venous blood volume (Buxton, 2013). Neural activation increases the BOLD response, but the BOLD response is only an indirect measure of neural activation (Ogawa et al., 1990a,b). So the second problem in applications of encoding models to fMRI data is to link the neural activation values predicted by the models to the observed BOLD time series recorded in fMRI experiments.

This section considers each of these problems in turn.

2.2.1 Aggregating Channel Responses

Each voxel in an fMRI experiment will include several hundred thousand neurons. As a result, any voxel-based encoding model that includes multiple channels will assume that every voxel in the ROI could potentially contain all of the hypothesized channels. This is true no matter how the channels are defined, although most models assume that the number of channels, and the number of neurons within each channel are unknown. The most popular assumption is that the neural activation produced in a task-sensitive voxel in response to a stimulus presentation is a weighted linear combination of all the channels represented in that voxel, where the weights are presumed to reflect the number of neurons within the voxel that contribute to each channel. Models in this class are often referred to as linearized encoding models because the measurement model assumes that the voxel-level neural activation is a weighted linear combination of the individual channel responses. When combined with a linear model of the relationship between neural activation and the observed BOLD response, such models can use the GLM for parameter estimation – that is, to estimate the values of the unknown weights that allow the model to give the best fits to the observed BOLD responses collected from that voxel on all TRs.

We can formalize these ideas as follows. Let $a_k(\mathrm{S}_i)$ denote the aggregate neural activity in voxel k to presentation of stimulus S_i, and let w_{ck} denote the contribution of channel c to this activity. Then the voxel-based (or linearized) encoding model assumes that

$$a_k(\mathrm{S}_i) = w_{1k} + \sum_{j=2}^{N_c} w_{jk}\ r_j(\mathrm{S}_i) + \epsilon_{m,k}, \tag{8}$$

where w_{1k} is the response of one channel in voxel k that gives the same constant response to all stimuli (to account for baseline activation that might occur in a voxel containing none of the hypothesized channels), and $\epsilon_{m,k}$ is the measurement error on channel k. The most common assumption is that $\epsilon_{m,k}$ is normally distributed with mean 0 and variance σ_m^2. This is called a linearized encoding model because it makes the simplifying assumption of a linear relation between channel responses and voxel activity. Note that this model predicts that the voxel activity $a_k(\mathrm{S}_i)$ is normally distributed or approximately normally distributed (in the Poisson case) with mean

$$\mathrm{E}[a_k(\mathrm{S}_i)] = w_{1k} + \sum_{j=2}^{N_c} w_{jk} f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x}) \tag{9}$$

and in the case where the channels are independent, with variance

$$\mathrm{Var}[a_k(\mathrm{S}_i)] = \sigma_m^2 + \sum_{j=2}^{N_c} w_{jk}^2 \mathrm{Var}[r_j(\mathrm{S}_i)], \tag{10}$$

where $\mathrm{Var}[r_j(\mathrm{S}_i)]$ either equals σ_c^2 in the case of the Eq. 2 Gaussian model or $f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \mathbf{x})$ in the case of the Eq. 3 Poisson model.

Note that this model accounts for the separate contributions of the channel noise and the measurement noise ($\epsilon_{m,k}$ in Eq. 8) to the variability in $a_k(\mathrm{S}_i)$. In almost all cases, however, these will not be separately estimable. In fact, in linear models, it is well known that they are nonidentifiable. Instead, only the sum of these separate variances can be estimated (e.g., Ashby 1992). As a result, in most applications, a single noise variance will be estimated and the source of the noise will be impossible to identify. Nevertheless, we include both noise sources for completeness.

Of course, there is a separate equation like Eq. 8 for every stimulus in the ensemble. In all of these, the weights are identical because the weights are presumed to reflect the dominance of each

channel within the voxel, which does not depend on what stimulus is presented. In contrast, the channel responses reflect the dominance of each feature within the stimulus, so these will change when the stimulus changes, but should be the same in all voxels. The standard way to keep track of all this is in matrix form. For example, consider an experiment with N_s different stimuli or events. The first step is to collect all channel responses – one for every channel – in an $N_s \times N_c$ channel-response matrix $\mathbf{R}$ defined as

$$\mathbf{R} = \begin{bmatrix} \boldsymbol{r}(\mathrm{S}_1)^\top \\ \boldsymbol{r}(\mathrm{S}_2)^\top \\ \vdots \\ \boldsymbol{r}(\mathrm{S}_{N_s})^\top \end{bmatrix}. \tag{11}$$

So row i of $\mathbf{R}$ lists the population response to presentation of stimulus S_i, and column c lists the response of channel c to the presentation of each stimulus. If channel noise is modeled, then $\mathbf{R}$ is a random matrix. In most linearized encoding models, however, channel noise is not included and thus each channel is characterized by its mean response, computed as in Eq. 1.

Encoding models assume that the channels and their tuning functions are known, so the mean channel response matrix E[$\mathbf{R}$] can be computed as soon as the stimulus set is selected, and therefore before the experiment begins. Voxel-based encoding models are therefore not used to estimate channel responses, because these are assumed to be known beforehand. Applying a voxel-based encoding model to neuroimaging data instead answers three different questions. First, it can identify the ROIs where the voxel activity most closely resembles the responses predicted by the set of presumed channels. Second, it provides an estimate of the relative frequency of each channel within every voxel. And third, for any single ROI it can tell whether the observed voxel activities are more consistent with one set of presumed channels or with another set.

The channel-response matrix described in Eq. 11 accounts for the channel responses. The full set of model predictions can then be written in matrix form as

$$\begin{bmatrix} a_k(\mathrm{S}_1) \\ a_k(\mathrm{S}_2) \\ \vdots \\ a_k(\mathrm{S}_{N_s}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}(\mathrm{S}_1)^\top \\ \boldsymbol{r}(\mathrm{S}_2)^\top \\ \vdots \\ \boldsymbol{r}(\mathrm{S}_{N_s})^\top \end{bmatrix} \begin{bmatrix} w_{1k} \\ w_{2k} \\ \vdots \\ w_{N_c k} \end{bmatrix} + \begin{bmatrix} \epsilon_{m,1} \\ \epsilon_{m,2} \\ \vdots \\ \epsilon_{m,N_s} \end{bmatrix},$$

or in shorthand form as

$$\boldsymbol{a}_k = \mathbf{R}\boldsymbol{w}_k + \boldsymbol{\epsilon}_m, \tag{12}$$

where the random vector $\boldsymbol{\epsilon}_m$ has a multivariate normal distribution with mean vector $\mathbf{0}$ and variance-covariance matrix $\boldsymbol{\Sigma}_m$. Most applications assume that $\boldsymbol{\Sigma}_m = \sigma_m^2\mathbf{I}$, where $\mathbf{I}$ is the identity matrix, and they also ignore channel noise, in which case $\mathbf{R}$ is replaced by E[$\mathbf{R}$]. In these cases, the only free parameters in the model are the weights $w_{1k}, w_{2k}, ..., w_{N_c k}$ and σ_m^2. Note that under these conditions, Eq. 12 has exactly the same form as the GLM in statistics, which is usually stated as $\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$. As a result, if we assume that $\boldsymbol{a}_k$ is linearly related to the observed BOLD response, then we can estimate the unknown weights in $\boldsymbol{w}_k$ by solving the normal equations of the GLM (more on this shortly).

Equation 12 applies the encoding model to activity values from a single voxel. It is straightforward to extend the model to multiple voxels in an ROI. Adding more voxels does not change E[$\mathbf{R}$] since all voxels are exposed to the same stimulus events on every TR. Even so, the model allows two voxels to respond differently to the same stimulus because the channels might have different relative frequencies in the two voxels. So for every new voxel that is added, a new set of weights must be estimated. Mathematically, this is easily done by replacing the vector of weights $\mathbf{w}$ with a matrix $\mathbf{W}$ in which column k contains the weights associated with voxel k. The vector of voxel

activities $\boldsymbol{a}_k$ is expanded to a matrix $\mathbf{A}$ in which column k contains $\boldsymbol{a}_k$ and we also need to add a new noise vector for each new voxel. These changes lead to the multivariate encoding model

$$\begin{bmatrix}\boldsymbol{a}_1 & \boldsymbol{a}_2 & \cdots & \boldsymbol{a}_{N_v}\end{bmatrix} = \mathbf{R}\begin{bmatrix}\boldsymbol{w}_1 & \boldsymbol{w}_2 & \cdots & \boldsymbol{w}_{N_v}\end{bmatrix} + \begin{bmatrix}\boldsymbol{\epsilon}_{m,1} & \boldsymbol{\epsilon}_{m,2} & \cdots & \boldsymbol{\epsilon}_{m,N_v}\end{bmatrix},$$

or in shorthand form as

$$\mathbf{A} = \mathbf{RW} + \mathbf{E}_m. \tag{13}$$

When channel noise is ignored, this model is identical to the multivariate GLM. While each column of $\mathbf{A}$ represents a different activity profile (i.e., the vector of activities of a single voxel across stimulus conditions), each row of $\mathbf{A}$ represents a different *activity pattern,* or the vector of activities across multiple voxels in response to a single stimulus condition (Diedrichsen and Kriegeskorte, 2017). The distinction between activity profile and activity pattern at the level of voxels is analogous to the distinction between tuning function and population response at the level of neural channels.

2.2.2 Linking Neural Activation to the BOLD Response

As mentioned previously, the BOLD response is a time series. Active brain areas consume more oxygen than inactive areas, so when neural activity increases in an area, metabolic demands rise, and, as a result, oxygenated hemoglobin rushes into the area. Neural activity causes an immediate oxygen debt, and the resulting rush of oxygenated hemoglobin into the area causes the BOLD signal to rise quickly until it eventually reaches a peak at around 6 seconds after the neural activity that elicited these responses. After this peak, the BOLD signal gradually decays back to baseline over a period of 20–25 seconds (with the decay typically including a brief dip below baseline).

In contrast, the encoding models considered so far are static, in the sense that the predicted aggregate neural activity $a_k(\mathrm{S}_i)$ to presentation of stimulus S_i is a single value. All static encoding models make the same simplifying assumption that the amplitude of the BOLD response in a voxel is proportional to the aggregated neural activation that occurs in that voxel. This enormously simplifies the problem of linking the aggregate activity predicted by the model to the observed BOLD response recorded in the experiment. The only remaining problem is to estimate a single amplitude of response from the BOLD time series. Furthermore, in most experiments, each stimulus will be presented multiple times, so there will be more than one such time series for stimulus S_i. Therefore, to apply a static encoding model, a single value that represents the amplitude of the BOLD response to stimulus S_i in voxel k must be estimated from these data. This problem is known in the neuroimaging literature as deconvolution or unmixing, and a solution to it is also required in decoding methods, such as multivoxel pattern analysis (MVPA). Not surprisingly, many alternative estimators have been proposed (e.g., Mumford et al., 2012; Pedregosa et al., 2015; Turner et al., 2012).

In rapid event-related designs, which are the norm in modern fMRI research, stimuli are presented within 5 seconds or so of each other, as they are in most laboratory experiments. Since the BOLD response to neural activity might persist for 30 seconds, this means that the BOLD signals elicited by successive stimulus presentations will overlap in time. This overlap complicates the unmixing process. Mumford et al. (2012) proposed an efficient solution to this problem that they called Least Squares – Separate (LSS). If there are N_E separate stimulus presentations, then LSS reruns the standard GLM regression analysis N_E separate times on the data from each voxel. In the i^{th} of these N_E runs, the GLM includes two parameters – one regressor for the single trial on which the i^{th} stimulus was presented and a second nuisance regressor that models the response to all other stimuli. The regression weight associated with the i^{th} stimulus in this analysis is used as an estimate of the amplitude of the BOLD response in voxel k to the presentation of stimulus S_i. We will denote the BOLD times series in voxel k as $b_k(t)$ and the amplitude of this time series

on trials when stimulus S_i is presented as $\tilde{b}_k(S_i)$. This LSS method was the most effective of a variety of alternative estimation methods investigated by Mumford et al. (2012).

After the values of $\tilde{b}_k(S_i)$ are estimated for all stimuli, these can be used to populate a vector $\tilde{\boldsymbol{b}}_k^\top = [\tilde{b}_k(S_1), \tilde{b}_k(S_2), ..., \tilde{b}_k(S_{N_s})]^\top$ that describes the amplitude of the BOLD response in voxel k to all N_S stimuli used in the experiment. Similarly, after repeating this process for all voxels, we form the matrix

$$\tilde{\mathbf{B}} = [\tilde{\boldsymbol{b}}_1, \tilde{\boldsymbol{b}}_2, ..., \tilde{\boldsymbol{b}}_{N_v}]. \tag{14}$$

The assumption that the BOLD response is proportional to aggregate neural activity means that there exists some constant λ, such that $\tilde{\boldsymbol{b}}_k = \lambda \boldsymbol{a}_k$ and $\tilde{\mathbf{B}} = \lambda \mathbf{A}$, where $\boldsymbol{a}_k$ and $\mathbf{A}$ are the aggregate activity vector and matrix from Eqs. 12 and 13, respectively. Note from those equations that the voxel-based encoding model therefore predicts that

$$\tilde{\boldsymbol{b}}_k = \lambda \boldsymbol{a}_k = \mathbf{R}(\lambda \boldsymbol{w}_k) + \lambda \boldsymbol{\epsilon}_m, \tag{15}$$

and

$$\tilde{\mathbf{B}} = \lambda \mathbf{A} = \mathbf{R}(\lambda \mathbf{W}) + \lambda \mathbf{E}_m. \tag{16}$$

Therefore, the constant λ can be absorbed into the weights and error variance. In other words, the weights and error variance include an unidentifiable constant of proportionality. This causes no problems however, because the primary interest is not in the absolute value of the weights, but rather in their relation to each other. For example, note that if one weight in a voxel is twice as large as another weight, then this 2-to-1 ratio holds for any value of λ. As a result, without loss of generality, we can ignore λ during parameter estimation, which means that the multivariate voxel-based encoding model can be described by

$$\tilde{\mathbf{B}} = \mathbf{R}\mathbf{W} + \mathbf{E}_m. \tag{17}$$

As mentioned previously, most applications either ignore channel noise or assume zero-mean, additive Gaussian noise. In either case, $\mathbf{R} = \mathrm{E}[\mathbf{R}]$, $\mathbf{E}_m$ describes the sum of channel and measurement noise, and the weight matrix $\mathbf{W}$ can be estimated from the normal equations of the multivariate version of the GLM. In most applications, the stimuli are presented far enough apart in time that it is safe to assume that the BOLD responses to separate stimuli are statistically independent. For this reason, and because it is common to assume homogeneity of variance (i.e., that each $\boldsymbol{\epsilon}_{m,k}$ in Eq. 13 has a multivariate normal distribution with variance-covariance matrix $\boldsymbol{\Sigma} = \sigma_m^2 \mathbf{I}$), the Gauss-Markov Theorem applies, and therefore the uniformly minimum variance, unbiased estimator of $\mathbf{W}$ is

$$\widehat{\mathbf{W}} = (\mathbf{R}^\top \mathbf{R})^{-1} \mathbf{R}^\top \tilde{\mathbf{B}}. \tag{18}$$

Note that Eq. 18 requires that $\mathbf{R}^\top \mathbf{R}$ is nonsingular. This is possible only if $N_s > N_c$, where N_s is the number of stimuli or events and N_c is the number of hypothesized channels. So the encoding model can only be tested against data in which there are more stimulus events than hypothesized channels. This makes sense, because in each voxel, there are unknown free weight parameters. To estimate these parameters uniquely, we need more data points than parameters. Each stimulus presentation produces one data point, so unique estimation of the weights requires that $N_s > N_c$. If this condition is not possible, then an alternative is to introduce extra constraints into the estimation procedure – a technique known in statistics as regularization (e.g., Bickel and Li, 2006). For example, this is the method used by Naselaris et al. (2009).

From a Bayesian perspective, regularization is accomplished by placing a prior on $\mathbf{W}$, so that some weight estimates are favored over others. This point is important, because regularization biases inference in favor of one $\widehat{\mathbf{W}}$ over many others that produce the same distribution of observed activity profiles $\tilde{\mathbf{B}}$. Some researchers have argued that, more than a simple technicality, this is an

important theoretical decision and should be considered an important aspect of the final model (Diedrichsen, 2020; Diedrichsen and Kriegeskorte, 2017).

Mathematically, the combination of an encoding model for $\boldsymbol{r}(\mathrm{S}_i)$ and a linear measurement model is equivalent to regression by linear combination of basis functions (Hastie et al., 2009). More specifically, the model captures the non-linear relation between stimuli S_i and BOLD responses by using a set of non-linear basis functions $f_c\left(\mathrm{S}_i, \boldsymbol{\theta}_c, \boldsymbol{x}\right)$ to transform the stimuli, and then uses a linear model on the transformed space to predict the amplitude of the observed BOLD response $\tilde{b}_k$.

Figure 2 shows this more clearly with an example using the standard encoding model discussed earlier. The figure depicts hypothetical data from one voxel along with theoretical predictions of the standard encoding model that has been linked to the linear measurement model described in Eq. 8. The hypothetical data are from an experiment in which a stimulus is presented on each trial that is a random sample from some ensemble that varies on a single physical dimension. Each open circle in the cloud of points shown in the top half of the figure depicts a hypothetical response recorded in this voxel on one trial. The value of each data point on the abscissa identifies the stimulus value on that trial. We call this scatterplot the *activity profile* of this voxel (Diedrichsen and Kriegeskorte, 2017, following the nomenclature of), and the dotted line represents the mean of this activity profile (sometimes called the voxel tuning function). The channel tuning functions from the standard encoding model are represented at the bottom, each scaled by its corresponding weight parameter w_{ck}. So note that in this hypothetical voxel, the most under-represented channels are centered at the stimulus values -35 and $+50$. The sum of these scaled functions is represented by the solid line at the top, which accurately approximates the observed mean activity profile. In practice, the channel weights are estimated by fitting the solid-line prediction of the model to the observed data – a process known in statistics as linear regression with radial basis functions.

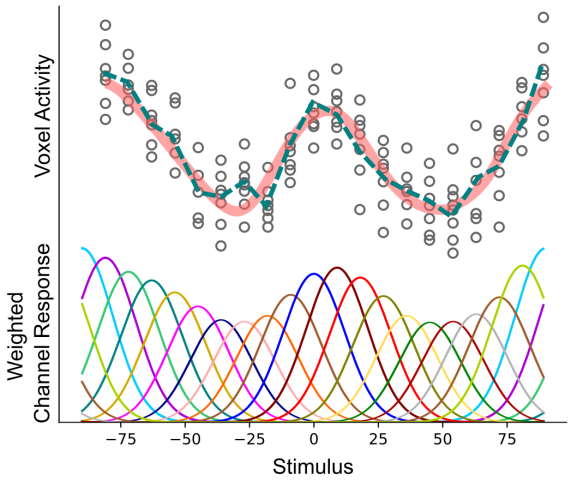

Figure 2: Hypothetical data from one voxel along with theoretical predictions of the standard encoding model. Each open circle in the top half depicts a hypothetical response from this voxel on one trial of an experiment in which the stimuli vary on a single physical dimension. The scatterplot of data is called the activity profile of this voxel, and the dotted line is its mean. The channel tuning functions from the standard encoding model are shown at the bottom, each scaled by its corresponding weight parameter. The solid line in the top half is the predicted activity profile of the standard encoding model, which equals the sum of the weighted tuning functions.

While more complex stimulus spaces and encoding models make the resulting model more

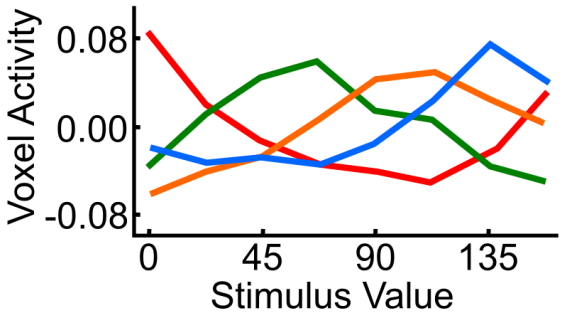

Figure 3: Mean activity profiles estimated by Serences et al. (2009).

difficult to interpret, the principle is the same: the activity profile of voxel k is modeled as a linear combination of basis functions. One issue with encoding modeling is that, in many cases, the set of basis functions will overfit the data. The reason is that the complexity and number of basis functions is selected either arbitrarily or based on theoretical considerations (e.g., the number of populations thought to underlie the voxel activity). In contrast, mean activity profiles are likely to be smooth and could probably be approximated by a small number of basis functions. For example, Figure 3 shows examples of mean activity profiles estimated by Serences et al. (2009). Note that the profiles are all unimodal and smooth, and each could probably be approximated with a single radial basis function. Although the profiles shown in this figure were averaged across many voxels, it is unlikely that much more structured variability would be found in single-voxel activity profiles, or at least not variability that can be distinguished from high levels of measurement noise common in fMRI.

2.2.3 Dynamic Encoding Models

All models considered so far are static, in the sense that they only predict the amplitude of the BOLD response to each stimulus. In contrast, many other encoding models are dynamic, including for example, dynamic causal modeling (Friston et al., 2003, DCM;). These models predict changes in neural activity over time – not just because of decay in the BOLD response, but also because of dynamic changes in neural, perceptual, and cognitive processing. Dynamic encoding models require a more detailed model, not only of how neural activity changes with time, but also of how the BOLD response depends on neural activity. In particular, they require a model that predicts the entire time-course of the BOLD response, rather than just its overall amplitude.

To begin, consider the differences between static and dynamic models in their predictions about channel responses and aggregate neural activity. Many dynamic encoding models, including DCM, do not assume that aggregate neural activity is driven by a population of separate channels. Instead, in these models, aggregate neural activity is the fundamental construct. DCM compensates for this simpler account of activation within any single voxel, by using different equations to predict neural activity in different voxels – especially voxels that are in different brain regions. In contrast, voxel-based encoding models typically apply the same model (and model equations) to all voxels. The goal in this case is to identify voxels in which the observed BOLD response agrees with these predictions.

To test any encoding model against data, we first generate a predicted activity vector for each voxel in the ROI. Let the $N_{TR} \times 1$ vector $\boldsymbol{a}_k^D$ denote the predicted neural activity in voxel k on every TR of the experiment. The superscript D (for dynamic) is to distinguish this vector from the static activity vector $\boldsymbol{a}_k$ described in Eq. 12. The two vectors are similar, in that they both

describe aggregate activity in a voxel, but note that $\boldsymbol{a}_k$ has N_S rows, whereas $\boldsymbol{a}_k^D$ has N_{TR} rows. The number of TRs in an experiment cannot be less than the number of stimuli that are presented, and in most experiments N_{TR} will be much greater than N_S. Therefore, in almost all applications $\boldsymbol{a}_k^D$ will have many more rows than $\boldsymbol{a}_k$. Row i of $\boldsymbol{a}_k$ describes the predicted aggregate activity to stimulus S_i in voxel k. In contrast, row i of $\boldsymbol{a}_k^D$ describes the predicted aggregate activity in voxel k on TR i. The static vector $\boldsymbol{a}_k$ includes an entry for every unique stimulus that predicts the same aggregate activity every time that stimulus is presented. The dynamic vector $\boldsymbol{a}_k^D$ includes an entry that predicts the aggregate neural activity on every TR of the experiment. So if stimulus S_i is presented 10 times, then $\boldsymbol{a}_k$ includes one value that predicts the same neural activity on each of these 10 presentations, whereas $\boldsymbol{a}_k^D$ will predict the effects of these 10 separate presentations on every TR of the experiment.

To test a dynamic encoding model against data from multiple voxels, we first generate predicted activity vectors for each of the N_v voxels in the ROI. The next step is to form the $N_{TR} \times N_v$ activity matrix $\mathbf{A}_D$ that includes $\boldsymbol{a}_{D,j}$ as column j. Note that this matrix is similar, but not identical to the matrix $\mathbf{A}$ in Eq. 13. They both describe aggregate activity in a set of voxels, but the columns of $\mathbf{A}_D$ are the dynamic activity vectors $\boldsymbol{a}_{D,j}$, whereas the columns of $\mathbf{A}$ are the static activity vectors $\boldsymbol{a}_j$.

If the model postulates an underlying population of channels that drive the aggregated neural activity, then a similar generalization is used to define the channel responses. In particular, the model is used to form the $N_{TR} \times N_C$ channel response matrix $\mathbf{R}_D$ that contains the predicted response of channel c on every TR of the experiment in column c and the predicted response of all channels on TR i in row i. Note that the relationship between $\mathbf{R}_D$ and the static channel response matrix $\mathbf{R}$ of Eq. 11 is similar to the relationship between $\mathbf{A}_D$ and $\mathbf{A}$. Given this dynamic channel response matrix, aggregate neural activity is predicted using a dynamic version of Eq. 12:

$$\boldsymbol{a}_k^D = \mathbf{R}_D \boldsymbol{w}_k + \boldsymbol{\epsilon}_{D,m}, \tag{19}$$

where the $N_{TR} \times 1$ random vector $\boldsymbol{\epsilon}_{D,m}$ has a multivariate normal distribution with mean vector $\mathbf{0}$ and variance-covariance matrix $\boldsymbol{\Sigma}_{D,m}$. Note that the weight vector $\boldsymbol{w}_k$ is identical in the static and dynamic models. In both cases, it specifies the relative contribution of each channel to the aggregate activity. The multi-voxel version of Eq. 19 is

$$\mathbf{A}_D = \mathbf{R}_D \mathbf{W} + \mathbf{E}_{D,m} \tag{20}$$

where $\mathbf{W}$ is defined exactly as in Eq. 13.

The next problem is to model the effects of dynamic changes in aggregate neural activity on TR-by-TR changes in the BOLD response. This is a problem that has received enormous attention in the fMRI literature. Almost all current applications of fMRI assume that the transformation from neural activation to BOLD response can be modeled as a linear, time-invariant system. Although a detailed examination clearly shows that the transformation is, in fact, nonlinear (e.g., Boynton et al., 1996), it also appears that the departures from linearity are not severe if the stimuli are of high contrast and brief exposure durations are avoided (Vazquez and Noll, 1998). These two conditions are commonly met in fMRI studies of high-level cognition.

Any linear, time-invariant system is completely characterized by its impulse response function, $h(t)$, which is the output of the system to an input that is an idealized impulse. More specifically, let $a(t)$ and $b(t)$ denote the (continuous-time) input and output of a dynamical system at time t, respectively. Then if the system is linear and time-invariant

$$b(t) = a(t) * h(t) = \int_0^\infty a(\tau) h(t - \tau) d\tau, \tag{21}$$

for any input and for all time t (e.g., Chen, 1970).

In dynamic encoding models, the input $a(t)$ is aggregate neural activity, the output $b(t)$ is the BOLD response, and the impulse response function $h(t)$ is commonly referred to as the *hemodynamic response function* (hrf). There are a variety of different methods for selecting a functional form for the hrf (e.g., Ashby, 2019). Common choices include a gamma function or a difference of gamma functions. Some researchers have also used boxcar functions with one or more ones around the peak of the hrf and zeros elsewhere (e.g., Çukur et al., 2013; Huth et al., 2012; Nishimoto et al., 2011). In all cases, however, parameters are chosen so that the resulting hrf peaks at around 6 sec and then decays back to 0 after 30 sec or so.

Dynamic encoding models make dynamic predictions about how neural activation $a(t)$ changes moment-by-moment. Therefore, in such models, Eq. 21 is used to convert model predictions to values of the observed dependent variable – that is, to values of the BOLD response $b(t)$.

Equation 21 assumes that the BOLD response is measured in continuous time. In practice, however, the BOLD response is measured only at discrete time points separated by a fixed amount of time equal to the TR. So rather than a continuous-time integral, the Eq. 21 convolution is done in discrete time. This can be accomplished using simple matrix multiplication by loading values of the hrf into the appropriate Toeplitz matrix.[1]

The Toeplitz matrix, which has order $N_{TR} \times N_{TR}$, includes a time-lagged discrete representation of the hrf in each column. To build the matrix, we begin by discretizing the hrf in a way that is similar to how we discretized the neural predictions of the model. The only difference is that any reasonable model of the hrf will include nonzero values only for 30 sec or so, whereas the functional run is likely to last 5 minutes or longer. Suppose we assume that the BOLD response to an impulse of neural activation persists for at most N_h TRs (since the hrf is an impulse response function). Then our discretized version of the hrf will be a vector $\boldsymbol{h}^\top = [h_1, h_2, ...h_{N_h}]^\top$, where $h_i = h(t = i \times TR)$. Next, we use $\boldsymbol{h}$ to build the appropriate Toeplitz matrix:

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 \\ h_2 & h_1 & 0 & \dots & 0 \\ h_3 & h_2 & h_1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ h_{N_h} & h_{N_h-1} & h_{N_h-2} & \dots & 0 \\ 0 & h_{N_h} & h_{N_h-1} & \dots & 0 \\ 0 & 0 & h_{N_h} & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \dots & h_{N_h} \end{bmatrix}. \tag{22}$$

Given this matrix, the discrete-time version of the Eq. 21 integral reduces to the simple matrix multiplication:

$$\boldsymbol{b} = a(t) * h(t) = \mathbf{H}\boldsymbol{a}_D. \tag{23}$$

Therefore, note that the dynamic encoding model predicts that the observed BOLD response in voxel k on each TR equals $\boldsymbol{b}_k = \mathbf{H}\boldsymbol{a}_{D,k}$.

The dynamic version of the voxel-based encoding model, which assumes that aggregate activity is driven by a population of channels, is generalized from Equation 20 by noting that the predicted aggregate activity matrix $\mathbf{A}_D = \mathbf{R}_D\mathbf{W}$ and therefore the predicted $N_{TR} \times N_v$ BOLD response matrix $\mathbf{B} = \mathbf{H}\mathbf{A}_D$. Combining these produces the multi-voxel, dynamic voxel-based encoding model:

$$\mathbf{B} = \mathbf{H}\mathbf{R}_D\mathbf{W} + \mathbf{E}_D, \tag{24}$$

where $\mathbf{E}_D$ is now a combination of noise at the level of neural channels, voxel activities, and BOLD responses.

[1] A Toeplitz matrix is any matrix in which all descending diagonals are filled with the same value.

The traditional GLM analysis of fMRI data, which is typically implemented in the popular fMRI data analysis software packages SPM and FSL, can be considered a special case of Eq. 24 (van Gerven, 2017), in which different channels respond to different stimulus events (e.g., each different type of stimulus, the participant's response, feedback, etc.), and each channel response is a boxcar function of zeros and ones, representing the absence and presence, respectively, of that event on each TR. Therefore, the true contribution of encoding models is not in the linearized measurement model, which was already available in the standard GLM approach, but rather in the much more detailed models of the possible computations performed by each channel.

The models we have considered so far all assume that the transformation from neural activity to BOLD response can be modeled as a linear, time-invariant system. More detailed models attempt to account for nonlinearities in the BOLD response. The most popular is the balloon model (Buxton et al., 1998), which models key biomechanical properties of the brain's vasculature. The balloon model accounts for the conflicting effects of dynamic changes in both blood oxygenation and blood volume and assumes that the blood flow out of the system depends on a balloon-like pressure within the vasculature. For example, when the blood flow is high, the walls of the blood vessels are under greater tension, and as a result they push the blood out with greater force, which reduces the rate at which oxygen is extracted from the hemoglobin. DCM, as implemented in the fMRI software package SPM (i.e., DCM10/SPM8), converts predicted neural activations to BOLD responses via a generalization of the balloon model. In contrast, most encoding models settle for a linear systems approach, and therefore instead convert predicted neural activations to BOLD responses via the convolution integral of Equation 21.

2.3 Population Receptive Fields

The population receptive field (pRF) of a voxel is a description of the region of the visual field where stimulus presentations produce an fMRI response (Dumoulin and Wandell, 2008; Wandell and Winawer, 2015). For example, panel (a) in the right column of Figure 4 shows the pRF of the traditional approach, which assumes that the pRFs of all individual neurons in a voxel can be described by a single population-level pRF. In its traditional implementation, the presented stimulus is represented by an indicator function $s(x, y) = \{0, 1\}$, where the values 0 and 1 denote the absence and presence, respectively, of any part of a stimulus at spatial coordinates (x, y). The pRF is modeled by a two-dimensional isotropic Gaussian in the same space:

$$g\left(x, y\right) = \exp\left[\frac{\left(x - x_0\right)^2 + \left(y - y_0\right)^2}{2\varsigma^2}\right], \tag{25}$$

were (x_0, y_0) is the center (i.e., mean) and ς the spread (i.e., standard deviation) of the receptive field. The predicted response of a voxel in which the pRFs of all neurons can be described by this single population-level pRF is computed by location-by-location multiplication of the stimulus value and the voxel pRF and then summing all these responses:

$$r\left(s_i\right) = \int_{x_L}^{x_U} \int_{y_L}^{y_U} s_i\left(x, y\right) g\left(x, y\right) dxdy, \tag{26}$$

where x_L, x_U, y_L, and y_U represent the lower (L) and upper (U) boundaries of the visual field along the x and y coordinates. As in most applications, the model implicitly assumes that $r\left(s_i\right)$ includes additive Gaussian neural noise. The voxel activity is assumed to be a scaled version of the population response

$$a_k\left(s_i\right) = wr\left(s_i\right), \tag{27}$$

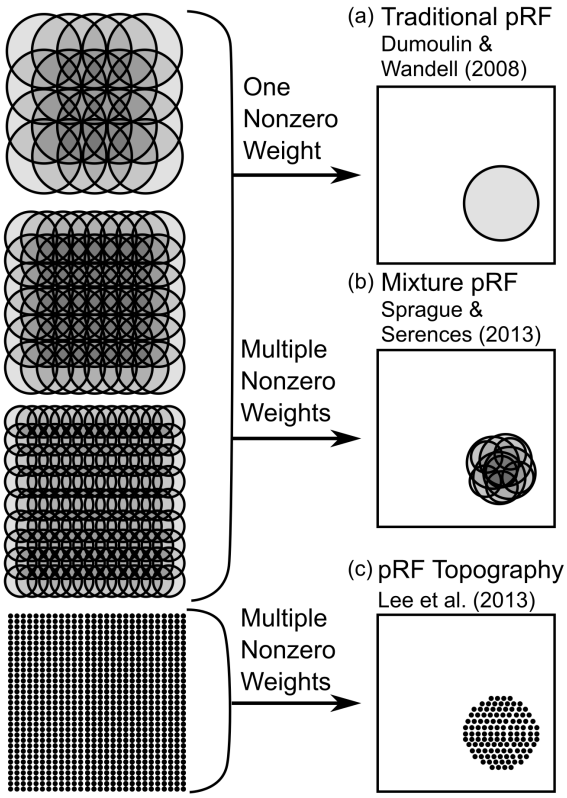

Figure 4: The population receptive field (pRF) method can be seen as an application of encoding modeling. Traditional pRFs (a) are obtained from a dense population encoding model contrained to have a single nonzero weight. Other models of the pRF (b,c) are obtained by modifying the basis set and/or the constraints imposed on the weights

, and the BOLD response is modeled as described in the previous section. Estimating the pRF of a voxel is done by finding the values of parameters x_0, y_0, and ς that that allow the model to provide the best possible fit to the observed BOLD response.
The pRF technique is usually considered an alternative to the linearized encoding modeling that is the focus of this chapter, but it can also be seen as a special case of the general encoding model framework. As shown in Figure 4a, the problem of estimating a pRF can be recast as an encoding modeling problem. First, one creates an encoding model with a large number of channels, each having a receptive field with a slightly different position and size, as illustrated in the left column of Figure 4. Second, to mimic the traditional pRF approach, one constrains all channel weights to be zero except for one, in order to accommodate the assumption that the pRFs of all neurons in the voxel can be modeled by one population-level pRF, and therefore that the data from each voxel can be modeled by a single channel. The traditional pRF approach is therefore equivalent to assuming a large number of channels that densely cover the space of possible size and location parameters, and then finding the single nonzero weight that provides the best fit to the data. The single channel with a nonzero weight has the position and size of the traditional pRF.

Of course, a more traditional encoding model that includes many channels also could be used to describe the pRF (see Figure 4b). This model would include nonzero weights for multiple channels, with each channel characterized by a receptive field of different position and size. Sprague and

Serences (2013) used such a mixture model to study the effects of spatial attention on neural representations in visual cortex. After the model is fitted to data, the pRF is equivalent to the predicted mean activity profile (the solid line curve in the top half of Figure 2). The resulting pRF is likely to be similar to the one obtained by assuming a single channel, but this encoding modeling approach has the advantage of more transparently reflecting the empirical observation that the voxel pRF is a mixture of multiple neural receptive fields of smaller size (Dumoulin and Wandell, 2008).

Other advantages of describing pRFs as applications of encoding modeling are that it encompasses other techniques proposed to obtain pRFs, it facilitates the understanding of how different techniques relate to one another, and suggests new techniques that could be useful in research. Because the linearized encoding model can be understood as linear regression with basis functions, alternative pRF models are easily obtained simply by changing the basis functions or the constraints used to estimate weights. For example, Lee et al. (2013) proposed an alternative method for estimating pRFs, illustrated in Figure **??**c, which uses Kronecker delta functions (i.e., impulses) as the basis set. In this approach, the pattern of estimated weights directly models the pRF topography.

Insights obtained from the pRF approach could also benefit encoding modeling more generally. In particular, pRFs are defined in the stimulus space and their parameters have interpretable units, which allows researchers to make meaningful comparisons across participants, conditions, and measurement instruments (Wandell and Winawer, 2015). As discussed in the next section, the parameters of a fitted encoding model can be difficult to interpret. The pRF approach, however, allows researchers instead to focus on characterizing, for each voxel, the mean activity profile predicted by a fitted encoding model (solid curve in the top half of Figure 2). Most commonly this means estimating the mode and spread of the mean activity profile, but other features of the function (support, derivatives, etc.) may also be informative. Unlike the traditional pRF approach, an encoding modeling approach could describe selectivity along any stimulus dimension, not only spatial sensitivity within visual field space.

2.4 Feature Spaces and Model Interpretation

The development so far is quite general, in the sense that it encompasses voxel-based encoding models, the standard GLM approach to constructing a statistical parametric map, population receptive fields, and model-based fMRI (developed in more detail below). What is common to these different approaches is the use of a linear measurement model with Gaussian noise (i.e., the GLM). They differ mainly in how they define a channel and a channel response [i.e., $r_c(\mathrm{S}_i)$]. The space of channel responses is sometimes called feature space (e.g., Diedrichsen, 2020), and the power and flexibility of the encoding modeling approach lies in the possibility of choosing among many different feature spaces.

For example, Naselaris et al. (2009) constructed one voxel-based encoding model in which each channel was a Gabor wavelet and another in which each channel responded to a different semantic category of objects – for example, birds, fish, or vehicles. Whereas the Gabor wavelet encoding model gave good accounts of BOLD responses in low-level visual cortical areas, the semantic encoding model gave good accounts in high-level association areas. So an encoding model approach can be used to identify brain regions that are sensitive to whatever features are hypothesized to drive the channel responses. A model based on features that do not match any set of channels in the brain should provide a poor account of BOLD responses in all ROIs.

The Gabor wavelet model was motivated by a long line of vision research on the sensitivity of V1 neurons to spatial frequency and orientation. In the case of high-level visual areas, however, the decision about how to define the channels is often more arbitrary. For example, in the case of the semantic-encoding model, the decision was made to include channels that respond to the

presence of certain categories of natural objects (e.g., birds and fish), but not others, and the object classes that were chosen had to be hand coded in every image by human observers (e.g., does this image contain a bird?). More recently, there have been a number of attempts to identify features, and therefore to define the underlying channels, by using artificial neural networks (e.g., Eickenberg et al., 2017; Güçlü and van Gerven, 2015). The general approach is to construct a multilayer neural network – commonly a deep convolutional neural network – and then train it to classify a database of natural images. After training, the output of each layer is interpreted as a different possible set of channel responses, and these are compared to the BOLD responses from different ROIs within the visual system.

For example, Güçlü and van Gerven (2015) trained a deep neural network that included 5 convolutional and 3 fully connected layers to classify images into 1 of 1000 different object categories. The network was trained on a database of around 1.2 million natural images using a supervised learning algorithm. After training was complete, each of the 8 layers of the network were used to define 8 different possible sets of channels, and therefore 8 different encoding models. Each of these 8 models was then tested against the fMRI data reported by Naselaris et al. (2009) by using an output model similar to Eq. 12. Overall, the models gave good accounts of visual responses across the entire ventral stream. Furthermore, the BOLD responses in early visual areas were best accounted for by early network layers, whereas in higher-level (i.e., downstream) visual areas, the BOLD responses were best fit by higher-level network layers.

The neural network used in this application included some features that were inspired by neural processing in the human brain (e.g., convolutional layers). But the model has much closer ties to the machine-learning literature than to neuroscience. Essentially, it can be viewed as an attempt to build an optimal model of object classification. The fact that it gives a good account of BOLD responses in visual cortex as humans view images of natural scenes suggests that the human visual system may have evolved to optimize object classification.

In sum, the feature space can be the response of filters to images, the responses of units in a deep neural network, variables in an abstract cognitive model, labels applied by researchers to their stimuli, etc.. This flexibility allows researchers to propose multiple competing feature spaces to explain neural activity in a particular brain region, and use model selection techniques (Zucchini, 2000) to choose one that describes the data best without overfitting. Ideally, the set of competing models would include only feature spaces that are theoretically relevant, preferably supported by evidence from past research.

Unfortunately, the flexibility and power of encoding models also leads to a number of issues of model interpretation.

The first problem is that sometimes it is unclear whether the feature space is a representation of the stimuli S_i or of the neural channel responses r_c. Many encoding models provide a separate notation for stimuli and channel responses, together with equations indicating how to compute channel responses given the presentation of a stimulus. On the other hand, some applications have used a set of hand-coded stimulus labels as the feature set (e.g., Çukur et al., 2013; Huth et al., 2012), with binary indicator variables used to represent such labels. In this case, it is unclear whether those variables are assumed to represent the presence of a stimulus or the response of a channel that is dedicated to the detection that stimulus. If one assumes that the feature space is a representation of the stimuli, then the linear measurement model assumes a linear mapping, not only from channels to measurements, but also from stimuli to neural responses. Both would be described by the estimated parameter matrix $\widehat{\mathbf{W}}$ (i.e., from Eq. 18). On the other hand, if one assumes that the labels are a representation of channel responses (i.e., populations of neurons that are active when the stimulus feature is presented), then there is an unknown transformation between S_i and r_c, which is likely nonlinear and is not explicitly modeled. The way in which most researchers discuss their results suggests that the latter interpretation is most common. For example, when Naselaris et al. (2009) compared the Gabor wavelet model against the semantic

model that was constructed by hand-coding labels in each image, they implicitly assumed that both models were identical except for the type of features to which the underlying channels were tuned. What this type of comparison does not take into account is the quality of the encoding models themselves. That is, only the Gabor wavelet model provides an explicit mechanistic description of how each channel responds to any possible stimulus.

The second issue has to do with the interpretation of the weight matrix $\widehat{\mathbf{W}}$. It is tempting to interpret estimated weights as providing information about the relative importance of different channels in the activity of a given voxel. This was the interpretation we assigned each weight when building the model (e.g., see Eq. 8). However, those were forward inferences, whereas interpreting entries in $\widehat{\mathbf{W}}$ after model fitting is a backward inference. And in the case of encoding models at least, backward inferences are tricky. There are multiple reasons why the entries in $\widehat{\mathbf{W}}$ might not provide the expected weight information (Kriegeskorte and Douglas, 2019). For example, in most cases, channels are not chosen to provide responses that are independent of each another, so multicollinearity among the channel responses may occur. Under these circumstances, weights are difficult to interpret because they do not reflect the effect of each channel independently from all others. In addition, some models are over-parameterized, in the sense that many different weight matrices describe the data equally well (i.e., so $\mathrm{R}^\top\mathrm{R}$ in Eq. 18 is singular). In practice, such identifiability problems are solved using regularization, but this reflects the choice of a particular prior over weights (Diedrichsen and Kriegeskorte, 2017). A channel with a large weight under one prior could have no weight under a different prior, so interpretation of weights should take into account what assumptions about the measurement model are implemented by the chosen prior.

A third, related issue has to do with interpreting the success of an encoding model to describe data from a given voxel as evidence that the feature space of the model is represented in the voxel. This is called the feature fallacy error because, for any given feature space used to describe voxel activities, there are an infinite number of other feature spaces that will make the exact same predictions, given that the matrix of weights $\widehat{\mathbf{W}}$ is modified accordingly (e.g., by choice of an appropriate prior; Diedrichsen 2020; Diedrichsen and Kriegeskorte 2017).

Gardner and Liu (2019) recently showed why this is the case for the standard linearized encoding model described by Eq. 13. For example, consider a model, call it Model 1, in which the predicted activity matrix $\mathbf{A}$ equals

$$\mathbf{A} = \mathbf{R}_1\mathbf{W}_1, \tag{28}$$

where $\mathbf{R}_1$ is the expected value of the channel response matrix. Now consider a second model, Model 2, that postulates a different set of expected channel responses $\mathbf{R}_2$ that are linearly related to the Model 1 responses via

$$\mathbf{R}_2 = \mathbf{R}_1\mathbf{P}, \tag{29}$$

where $\mathbf{P}$ is some $N_c \times N_c$ nonsingular matrix. Therefore, note that the predicted aggregated activity matrix for Model 2 equals

$$\mathbf{A}_2 = \mathbf{R}_2\mathbf{W}_2 = \mathbf{R}_1\mathbf{P}\mathbf{W}_2. \tag{30}$$

Now if $\mathbf{W}_2 = \mathbf{P}^{-1}\mathbf{W}_1$, it follows that

$$\mathbf{A}_2 = \mathbf{R}_1\mathbf{P}\mathbf{P}^{-1}\mathbf{W}_1 = \mathbf{R}_1\mathbf{W}_1 = \mathbf{A}_1, \tag{31}$$

and therefore, both models predict exactly the same aggregated activity matrix, even though they postulate different channel responses and different weights. Diedrichsen and Kriegeskorte (2017) argued that similar model identifiability problems arise even when weights are estimated using regularization rather than by solving the normal equations (as in Eq. 18).

The identifiability and model mimicry problems that are endemic to encoding models are likely not restricted to models that span the exact same linear subspace. This becomes clear if we refer back to the Figure 2 example, which we used to illustrate that encoding models are a form of linear regression with radial basis functions. The radial basis functions illustrated in the bottom part of Figure 2 are not the only ones that could provide a good fit to the activity profile shown in the top part of the figure. Given enough channels, a model in which the basis functions are polynomials, splines, or even simple step functions could provide an arbitrarily good fit (Hastie et al., 2009, see).

What all this means is that one must be extremely careful when interpreting the success of an encoding model in terms of its basis functions or features. Sometimes a particular set of features is theoretically important, neurobiologically motivated, or simply easier to interpret. All of these are good reasons to prefer one basis set over others. At the same time, however, it is essential to acknowledge that the fit and predictive performance of a model do not guarantee, by themselves, that an ROI encodes stimuli using that specific basis set.

3 Model Inversion

Although encoding models provide the best opportunity to make causal inferences from fMRI data (Weichwald et al., 2015), decoding methods offer their own distinct advantages (e.g., Naselaris et al. 2011). One is that they allow decoding accuracy to be compared directly to human behavioral performance in each ROI. For example, Walther et al. (2009) compared the confusions that human observers made when categorizing natural scenes with the confusions made by an MVPA classifier in a variety of different visual ROIs. Although the human observers made fewer errors, the pattern of confusions made by the MVPA classifier in the parahippocampal place area was similar to the pattern of confusions made by the humans, whereas the pattern of confusions made by the classifier in V1 was not correlated (at least, not significantly) with the pattern made by the humans. Thus, this result supports a model in which the parahippocampal place area plays a key role in scene classification behavior.

Carlson and colleagues extended this approach by assuming that the observer's response time on each trial is related to the distance of the activity pattern to the best-fitting linear bound of an MVPA classifier (Carlson et al., 2013; Grootswagers et al., 2018; Ritchie and Carlson, 2016; Ritchie et al., 2015). The assumption that response time is inversely related to the distance between the percept and a decision bound has a long history in mathematical psychology (e.g., Ashby and Maddox, 1994; Murdock, 1985). Thus, if a particular brain region stores information that is extracted for behavioral performance, then it is likely that distances-to-bound obtained from a classifier trained on data from that region will correlate with response times and similar behavioral measures. Using this approach, the Carlson group has shown that brain regions that provide information that is read out for behavior are only a subset of the brain regions that contain decodable information.

Decoding methods are also popular because they provide the basis of the popular claims that fMRI can be used for mind reading (Haynes and Rees, 2006). In these applications, the BOLD responses are decoded to predict the stimulus event that occurred. Many exciting possibilities have been proposed –from communicating with patients who were diagnosed to be in a vegetative state, to lie detection, to enabling people to control external devices via thought (DeCharms, 2008).

Researchers who develop and test encoding models can exploit many of the advantages of decoding approaches via model inversion, which is the process of constructing a decoding scheme by inverting an encoding model. Perhaps the most immediate benefit of this process is that it allows unique tests of the encoding model that would otherwise be impossible. For example, a valid encoding model that accurately predicts how the BOLD response differs when different stimuli are presented should also be able to predict which stimulus was presented simply by examining

the BOLD response on each trial. In mathematical psychology, the validity of a model is typically assessed by examining its ability to predict what response was made (and perhaps also the response time), given knowledge of the stimulus. An inverted encoding model allows tests in the opposite direction – that is, it allows a test of the model's ability to predict what stimulus was presented, given knowledge of the response.

An encoding model predicts the aggregate activity in a voxel given knowledge of the stimulus (e.g., see Eq. 8). More specifically, a complete encoding model should predict the probability density function of aggregate activity in voxel k on trials when stimulus S_i is presented – that is, $P(a_k|\mathrm{S}_i)$. In this approach, Bayes' rule is used to to invert the model:

$$P(\mathrm{S}_i|a_k) \propto P(a_k|\mathrm{S}_i)P(\mathrm{S}_i), \tag{32}$$

where $P(\mathrm{S}_i)$ is the prior probability that stimulus S_i is presented. When stimuli are modeled in a physical stimulus space, such as the pixel space used to construct each stimulus, model inversion allows for full reconstruction of the presented stimulus. Of course, decoding is possible without the use of an explicit encoding model, as in MVPA, by training machine-learning algorithms to extract information about stimuli from activity patterns (Pereira et al., 2009, see).

The Eq. 32 decoding scheme operates directly on the model's predicted aggregate activities. As we saw earlier however, many models predict that aggregate activity is determined by the responses of a population of underlying channels (e.g., as in Eq. 8). These hypothesized channels have important consequences for model inversion. In particular, in addition to using the observed BOLD response to make inferences about what stimulus was presented (i.e., stimulus decoding), model inversion often makes it possible to use the observed BOLD response to make inferences about the channel responses, which typically are unobservable. Estimating the channel responses from a decoding scheme is a form of *population response reconstruction*.

Of course, if the channel responses are observable, then they could also be used for stimulus decoding. In other words, one could predict the presented stimulus either from the aggregated activity (i.e., the BOLD response) or from the channel responses. It is very important, however, to keep the distinction between these two forms of stimulus decoding in mind when interpreting the results of encoding and decoding studies. For example, the act of perception is a form of stimulus decoding because the brain must use neural activity to make inferences about the presented stimulus. But this decoding process must use channel responses. In fMRI experiments, the aggregate activity is the total neural activity in tens of thousands of neurons located in an arbitrarily defined cube of the brain. The neurons in this cube likely project to a variety of different targets, and therefore the downstream neurons are driven by the channels, not by the aggregated activity. Conversely, note that the fMRI experimenter has indirect access to the aggregated activity (i.e., via the BOLD response), but typically has no access to the responses of individual channels. Therefore, whereas the brain can only decode the stimulus from the channel responses, the experimenter can only decode the stimulus from the aggregated activity. Despite this important difference, it is common to find conflation of $\mathbf{r}$ (the channel response to stimulus S_i) and $\mathbf{a}_i$ (the aggregate activity in response to stimulus S_i) (e.g., Bobadilla-Suarez et al., 2020; Diedrichsen and Kriegeskorte, 2017), which may lead to incorrect theoretical conclusions.

During model inversion, researchers usually distinguish between training and testing data. The standard approach is to first use a set of training data from some ROI to fit the encoding model (i.e., estimate all free parameters). Next, the encoding model is inverted to create a decoding scheme. Finally, the decoding method is tested against new validation data from the same ROI.

To begin, let $\tilde{\mathbf{B}}_{\text{train}}$ and $\tilde{\mathbf{B}}_{\text{test}}$ denote the data matrices collected in the ROI during training and testing, respectively. Both matrices have order $N_s \times N_v$ and, as described by Eq. 14, they contain the amplitude of the BOLD response to all N_S stimuli in all N_v voxels. Row i summarizes the BOLD response to stimulus S_i in every voxel, and column k summarizes the response in voxel k to every stimulus. Now consider encoding models in which aggregate activity is assumed to depend

on responses from an underlying population of channels. In these models, the channel-response matrix $\mathbf{R}$ depends on exactly which stimuli are presented and on their order of presentation. The training and testing data might come from trials that present the same stimuli, but even in this case the order of stimulus presentation will typically differ. Therefore the channel-response matrices for training and testing will differ. Denote these two matrices by $\mathbf{R}_{\text{train}}$ and $\mathbf{R}_{\text{test}}$, respectively. Although encoding models assume the expected values of these two matrices will differ, they assume that the matrix of channel weights $\mathbf{W}$ will be the same during training and testing. This is because $\mathbf{W}$ depends on the relative frequencies of the different channels in the voxels within the search set, but not on the stimuli that are presented (i.e., see Eq. 13).

3.1 Population Response Reconstruction

Given that the population responses of the hypothesized channels are not directly observable with fMRI, an interesting application of model inversion is to estimate these responses (Brouwer and Heeger, 2009). In fact, this one application is what researchers in the literature usually refer to as "inverted encoding modeling" or IEM (e.g., Gardner and Liu, 2019; Liu et al., 2018; Sprague et al., 2018, 2019).

According to the multivariate encoding model described in Eq. 17, the predicted (i.e., mean) BOLD amplitude during training equals $\hat{\tilde{\mathbf{B}}}_{\text{train}} = \text{E}[\mathbf{R}_{\text{train}}]\widehat{\mathbf{W}}$. Note that $\hat{\tilde{\mathbf{B}}}_{\text{train}}$ and $\tilde{\mathbf{B}}_{\text{train}}$ are different. The former is the predicted BOLD response according to the model, whereas the latter is the observed BOLD response. Now to fit the encoding model to the training data, we first compute $\text{E}[\mathbf{R}_{\text{train}}]$ from the model, and then use $\tilde{\mathbf{B}}_{\text{train}}$ to compute $\widehat{\mathbf{W}}$ (from Eq. 18). Our goal is now to use the $\widehat{\mathbf{W}}$ matrix we estimated from the training data and the observed voxel activities during testing (i.e., $\tilde{\mathbf{B}}_{\text{test}}$) to estimate the matrix of expected population responses $\text{E}[\mathbf{R}_{\text{test}}]$, which we abbreviate as $\hat{\mathbf{R}}_{\text{test}}$. If we know these channel responses then we can infer which stimulus was presented simply by comparing the estimated channel responses (i.e., the rows of $\widehat{\mathbf{R}}_{\text{test}}$) to each row of the original expected channel-response matrix $\text{E}[\mathbf{R}_{\text{train}}]$ (see Eq. 14), assuming that the stimuli presented during testing were all presented one or more times during training.

At testing, the encoding model predicts that the BOLD responses should equal

$$\hat{\tilde{\mathbf{B}}}_{\text{test}} = \widehat{\mathbf{R}}_{\text{test}}\widehat{\mathbf{W}}. \tag{33}$$

Our goal is to solve for $\widehat{\mathbf{R}}_{\text{test}}$. Unfortunately however, since at this stage of the analysis $\widehat{\mathbf{R}}_{\text{test}}$ is unknown, so is $\hat{\tilde{\mathbf{B}}}_{\text{test}}$. If we did know $\hat{\tilde{\mathbf{B}}}_{\text{test}}$, then we could just solve for $\widehat{\mathbf{R}}_{\text{test}}$. Ester et al. (2015) proposed estimating $\hat{\tilde{\mathbf{B}}}_{\text{test}}$ with the observed data $\tilde{\mathbf{B}}_{\text{test}}$, and then solving the resulting equation for $\widehat{\mathbf{R}}_{\text{test}}$. This process produces the following estimator: [2]

$$\widehat{\mathbf{R}}_{\text{test}} = \hat{\tilde{\mathbf{B}}}_{\text{test}}\widehat{\mathbf{W}}^{\top}\left(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top}\right)^{-1}. \tag{37}$$

Note that $\widehat{\mathbf{W}}$ has order $N_c \times N_v$, so $\left(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top}\right)^{-1}$ exists only if $N_v \geq N_c$ – that is, only if there are at least as many voxels in the ROI or searchlight as there are channels. Adding more voxels

[2] If we estimate the predicted matrix $\hat{\tilde{\mathbf{B}}}_{\text{test}}$ with the observed data matrix $\tilde{\mathbf{B}}_{\text{test}}$, then Eq. 33 becomes

$$\tilde{\mathbf{B}}_{\text{test}} = \widehat{\mathbf{R}}_{\text{test}}\widehat{\mathbf{W}}. \tag{34}$$

Multiplying both sides by $\widehat{\mathbf{W}}^{\top}(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top})^{-1}$ produces

$$\tilde{\mathbf{B}}_{\text{test}}[\widehat{\mathbf{W}}^{\top}(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top})^{-1}] = \widehat{\mathbf{R}}_{\text{test}}\widehat{\mathbf{W}}[\widehat{\mathbf{W}}^{\top}(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top})^{-1}], \tag{35}$$

which implies

$$\widehat{\mathbf{R}}_{\text{test}}(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top})(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top})^{-1} = \tilde{\mathbf{B}}_{\text{test}}\widehat{\mathbf{W}}^{\top}(\widehat{\mathbf{W}}\widehat{\mathbf{W}}^{\top})^{-1}, \tag{36}$$

from which Eq. 37 easily follows.

to the ROI adds more data (i.e., each new voxel adds a column to $\mathbf{B}$), but the size of the search volume does not affect the size of $\mathbf{R}$ (since $\mathbf{R}$ has order $N_s \times N_c$). So the more voxels there are in the search volume, the more data we have to estimate the rows of $\mathrm{E}[\mathbf{R}_{test}]$.

As an example of how Eq. 37 is applied, Ester et al. (2015) used this approach to study visual representations during the delay period of a working-memory task in which subjects had to remember the orientation of a briefly presented Gabor pattern. The encoding model assumed 9 different orientation channels. They used a leave-one-run-out cross-validation procedure (e.g., see Ashby, 2019) in which they fit the encoding model to the data from all but one functional run by estimating the weight matrix $\mathbf{W}$ from these data using Eq. 18. Next, they used the data from the single withheld functional run to invert the encoding model – that is, to estimate $\mathrm{E}[\mathbf{R}_{test}]$ from Eq. 37, which provided an estimate of the channel responses during the delay period of each trial of the withheld run. In brain regions that maintain a visual representation of the stimulus during the delay period, the estimated channel responses should peak at the to-be-remembered orientation, whereas in any other region, the channel responses should all be roughly the same. Using this approach Ester et al. (2015) were able to identify a broad network of frontal, parietal, and occipital regions that maintained a high-fidelity visual representation during the delay period.

This method has also been used to study how psychological factors such as attention (Garcia et al., 2013; Sprague and Serences, 2013), working memory (Ester et al., 2013), or learning (Byers and Serences, 2014; Ester et al., 2020) influence population responses. In these studies $\widehat{\mathbf{W}}$ is estimated from training data, and then separate population responses are estimated from data collected in two or more test conditions using Eq. 37, each run under different levels of the psychological factor (e.g., with and without attention). Finally, these separate estimates are all compared.

Recall that row i of $\mathbf{R}$ lists the response of each channel in the population to presentation of stimulus S_i. If the tuning functions all have the same shape during both training and testing (i.e., the model is homogeneous), then each row of $\mathbf{R}$ should peak at the channel most sensitive to S_i and then decay as predicted by the channel tuning function $f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \boldsymbol{x})$ (i.e., see Eq. 1). To estimate this function, it is common to shift the rows in $\widehat{\mathbf{R}}_{\mathrm{test}}$ so that the peak of the response is in the same place across all channels (this is usually facilitated by the use of circular dimensions, such as orientation or color), followed by averaging of responses across rows. However, this method will fail if tuning is not homogeneous, which could happen for instance, if the test condition influences some channels more than others (Hays and Soto, 2020).

There has been much recent controversy regarding the correct interpretation of population responses that are estimated by inverting an encoding model (e.g., Gardner and Liu, 2019; Liu et al., 2018; Sprague et al., 2018, 2019). What does it mean to find, for example, that attention narrows the estimated population responses, or that it increases their amplitude? When the standard encoding model is assumed, a change in the channel tuning function $f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \boldsymbol{x})$ produces a corresponding change in the population responses. However, the converse is not necessarily true: if Eq. 37 is used to estimate the population responses, then a change in those estimates across conditions does not imply a corresponding change in the channel tuning functions.

For example, Liu et al. (2018) reported evidence that the Eq. 37 estimates of the population responses can be biased by noise. They ran an experiment in which gratings were presented at one of two different contrasts. Single-unit electrophysiology shows that orientation tuning is contrast invariant, so the width of the orientation channels should be the same for the two contrasts. Therefore, the population responses estimated via Eq. 37 should be contrast invariant. In violation of this prediction, Liu et al. (2018) found that the estimated population response widths were greater for the low-contrast gratings and they reported results of simulations supporting the hypothesis that this apparent bias was the result of decrements in signal-to-noise ratio that occur when contrast is reduced.

Sprague et al. (2018) defended the inverted encoding model approach of Eq. 37 by correctly

pointing out that its goal is to make inferences about population responses $\mathbf{r}$, not about individual tuning functions $f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \boldsymbol{x})$. However, there seems to be lack of clarity regarding the correct interpretation of an estimated population response. In terms of brain processing, channel responses are important because they are the input for downstream neurons that are part of the decoding network that makes perception possible (and more generally, any behavior). Any narrowing of the tuning function that might be caused, for example, by attention, therefore provides more precise downstream information for decoding. For this reason, Liu et al. (2018) are also correct when they point out that the information available for stimulus decoding is better characterized by the posterior distribution over stimuli $P(\mathrm{S}_i|a_k)$ (i.e., see Eq. 32) than by any reference to population responses (Van Bergen et al., 2015).

A focus on $P(\mathrm{S}_i|a_k)$ would also avoid a common issue in the literature, which is that many researchers interpret estimated population responses by reference and comparison to tuning functions from single-cell recordings, rather than by focusing on what population responses would mean for downstream processing. This is likely the result of how foreign the concept of a population response is to an experimental neuroscientist. Electrophysiologists rarely measure the response of multiple neurons or populations to a single stimulus. Instead, they typically measure the response of a single neuron or small number of neurons to many stimuli. For this reason, when Sprague et al. (2018) discuss population responses, a casual reader could misinterpret their use of "population-level channel response functions" as something like the channel tuning functions $f_c(\mathrm{S}_i, \boldsymbol{\theta}_c, \boldsymbol{x})$, rather than to their intended meaning as a pattern of distributed activity across channels (i.e., $\mathbf{r}$).

On the other hand, a focus on $P(\mathrm{S}_i|a_k)$ does not solve all the issues with model inversion highlighted by the Liu et al. (2018) results. In particular, inversion of an encoding model that does not capture some of the data-generating mechanisms will often lead to the wrong conclusions. In the Liu et al. (2018) study, the mechanism left out of the model was the influence of contrast on signal-to-noise ratio. Unfortunately, whether one inverts the model to obtain estimates of $\mathbf{r}$ or $P(\mathrm{S}_i|a_k)$, such estimates will be biased when the encoding model is grossly incorrect.

Although Eq. 37 provides biased estimates of channel tuning functions, it nevertheless is widely used because an important goal of experimental neuroscience is to make inferences about channel tuning functions from neuroimaging data. The obvious way to do this would be to estimate the parameters of the tuning functions via model fitting to the data (e.g., using adaptive basis functions), and then make these the target of inference rather than the population responses. A problem with this solution is that encoding models are already complex, so adding free parameters is likely to increase the identifiability problems that already exist. Sadil et al. (2021) recently addressed this issue by constraining the one-dimensional encoding model (see Eq. 4) in multiple ways. First, they assumed that tuning functions for all channels are identical except for their preferred stimulus (i.e., homogeneous population code). Second, they avoided the many free weight parameters that characterize standard encoding models (as in the Eq. 13 model) by assuming that the weights in each voxel follow a Gaussian-like curve centered at the stimulus value (e.g., orientation) that is preferred by the dominant channel in that voxel. Third, they limited the number of ways that the model predictions could be modified by some psychological or experimental factor (e.g., reducing stimulus contrast). In addition, they adopted a Bayesian framework that allowed them to introduce inferential biases through their chosen prior.

Inverted encoding modeling also falls victim to the feature fallacy error (Diedrichsen, 2020; Diedrichsen and Kriegeskorte, 2017). As explained earlier, an infinite number of channel response matrices can be chosen that produce exactly the same fit to the data (Gardner and Liu, 2019). Although these different channel responses all predict the same aggregate activity (see Eqs. 28 – 31), their population response profiles can have dramatically different shapes. This highlights the fact that inverted encoding is only useful when the obtained estimates of the population responses are interpreted with specific reference to the tuning functions and other features of the model that

was inverted (Sprague et al., 2019).

3.2 Stimulus Decoding and Reconstruction

The most common application of model inversion is not to estimate population responses, but either to decode stimulus values or to provide a full reconstruction of the presented stimulus.

For example, the Eq. 37 decoding scheme is easily extended to stimulus decoding – that is, from the problem of estimating the expected population response matrix E[$\mathbf{R}$] to the problem of testing the ability of the model to identify the stimuli that were presented during the test phase. The rows of the $\widehat{\mathbf{R}}_{\text{test}}$ matrix that results from Eq. 37 will not exactly equal any of the rows of the expected channel-response matrix E[$\mathbf{R}_{train}$] that we constructed when building the Eq. 13 encoding model (e.g., because of noise). So to use Eq. 37 to complete the loop back to the stimulus, we need a classification scheme that will assign a single stimulus to each row of $\widehat{\mathbf{R}}_{\text{test}}$. Under the assumption that the noise vector $\boldsymbol{\epsilon}_m$ in Eq. 12 has a multivariate normal distribution in every voxel with mean vector $\mathbf{0}$ and variance-covariance matrix $\boldsymbol{\Sigma} = \sigma^2\text{I}$, it turns out that for each row in $\widehat{\mathbf{R}}_{\text{test}}$, the optimal classification strategy is to compute the correlation with every row in E[$\mathbf{R}_{train}$] and then associate that row in $\widehat{\mathbf{R}}_{\text{test}}$ with the row in E[$\mathbf{R}_{train}$] where the correlation is highest (assuming that the prior probabilities $P(\text{S}_i)$ are equal for all stimuli; e.g., Fukunaga, 2013). Row i of E[$\mathbf{R}_{train}$] contains the expected response of each channel to the presentation of stimulus S_i. Therefore, we can denote this row by $\mathbf{r}_{\text{train}}(\text{S}_i)^\top$ (i.e., see Eq. 12). Row m of $\widehat{\mathbf{R}}_{test}$ was generated by the m^{th} event, but of course, we do not know which stimulus caused this event. So denote row m of $\widehat{\mathbf{R}}_{\text{test}}$ by $\widehat{\mathbf{r}}_{\text{test}}(\text{E}_m)^\top$. Then the optimal decoding scheme uses the following classification rule:

Classify the m^{th} event of the testing data as a stimulus S_i event if

$$\text{corr}\left[\widehat{\mathbf{r}}_{\text{test}}(\text{E}_m), \mathbf{r}_{\text{train}}(\text{S}_i)\right] = \max_{j=1,N_s} \text{corr}\left[\widehat{\mathbf{r}}_{\text{test}}(\text{E}_m), \mathbf{r}_{\text{train}}(\text{S}_j)\right]. \tag{38}$$

As indicated earlier, an encoding model is not necessary to perform stimulus decoding from fMRI data. This can also be achieved by training a machine-learning algorithm to extract information about stimuli from activity patterns. This type of non-parametric decoding appears in the literature more frequently than decoding by inverting an encoding model, but it has been argued that machine-learning approaches provide more limited opportunities to make inferences about underlying computational mechanisms (Kriegeskorte and Douglas, 2019; Naselaris et al., 2011). In other words, a common assumption in the field is that although nonparametric decoding analyses can reveal *what* information is encoded in a given brain region, they can not reveal information about *how* that information is encoded. On the other hand, experimental and modeling work reveals this to be at least partially incorrect.

For example, an important question in sensory neuroscience is whether a neural population encodes a stimulus property in a way that is invariant to some irrelevant stimulus change; that is, with encoding being the same across changes in an irrelevant feature. The opposite of such invariant encoding would be context-specific or configural encoding, in which the way a stimulus property is encoded by a population depends on the value of a second property. Both invariant and configural representations are important for discussions of how the brain represents objects and generalizes knowledge about them. Cognitive neuroscientists have used a variation of decoding analyses, called cross-decoding (or cross-classification, see Allefeld and Haynes, 2014; Anzellotti and Caramazza, 2014; Kaplan et al., 2015), to attempt to make inferences about invariant encoding in particular brain regions. The first step in cross-decoding is to train a classifier to decode a particular stimulus feature, such as the shape of an object, from patterns of fMRI activity observed across voxels. The second step is to test the trained classifier with new patterns of fMRI activity, this time obtained from presentation of the same stimuli, but changed in an irrelevant property, such as rotation in depth.

Theoretical and modeling work has shown that cross-decoding can indeed be used to make valid inferences about *how* stimuli are encoded in a particular area from neuroimaging data, without making any assumptions about specific aspects of the encoding model (Soto et al., 2018). However, cross-decoding provides evidence *against* the null hypothesis of context-specific encoding (i.e., generalization of decoding performance shows that encoding is not completely context-specific), and not evidence *for* the alternative of invariance. In addition, the test is prone to false positives because the measurement model can increase invariance in the transformation from neural to voxel space. Testing the null of invariance in addition to cross-decoding allows one to reach more precise and valid conclusions about the underlying representations. These theoretical insights have been verified through experimental and simulation work (Soto and Narasiwodeyar, 2021). It is likely that other general features of encoding can be inferred using non-parametric decoding, but more research is needed in this area.

In addition to simple decoding of the identity of a stimulus, model inversion can also be used for full stimulus reconstruction, thereby providing a method to visualize what has been encoded in the brain on a given trial. For example, Naselaris et al. (2009) used the structural model illustrated in Figure 1 and a Bayesian framework to reconstruct an image with the maximum posterior probability of having produced the measured BOLD activity. Their Bayesian framework allowed them to compare reconstruction under a variety of prior distributions over the images ($P(\mathrm{S}_i)$ in Eq. 32). They found that reconstruction with a flat prior, which uses only information from voxel activities captured in the encoding model, was insufficient to reveal the identity of objects in the reconstructed images. A more informative prior that included some well-known statistical information about natural images (a $1/f$ amplitude spectrum and sparsity in the Gabor-wavelet domain) produced more natural-looking images, but still was unable to provide information about object identity. Finally, they attempted to better capture the prior distribution over natural images by sampling from it: they used a database of six million images as a prior, so that each image in the set had a prior probability of $\left(6 \times 10^6\right)^{-1}$, and any image outside this set had a prior probability of zero. This prior enabled them to reconstruct both the spatial structure and semantic content of the original images. A similar approach was used to reconstruct videos presented to participants from fMRI data (Nishimoto et al., 2011).

More recent research in this area leverages the power of deep learning for image reconstruction (e.g., Ren et al., 2021; Seeliger et al., 2018; Shen et al., 2019), achieving reconstructions that could be recognized by humans without the need to sample explicitly from some pool of natural images.

4 Representational Similarity Analysis

Representational similarity analysis (RSA) is a multivariate method that extracts similarity structures from BOLD activity (Kriegeskorte et al., 2008). It identifies activation patterns that are similar and others that are dissimilar. A fundamental assumption is that two data sets that exhibit a comparable similarity structure must share a deeper homology in how the systems that generated those data represent and process events in the world. Perhaps the greatest strength of RSA is that a common approach can be used to extract similarity structures from many different modalities, allowing links to be drawn between vastly different levels of analysis. For example, consider a mathematical model of some perceptual or cognitive task that makes no neuroscience predictions per se, but instead assumes that performance depends on some hypothetical intervening variable, such as working memory load, attention, or reward prediction error. Next, suppose that for each pair of possible trial types, we use the model to compute a predicted similarity by comparing its predicted values on the intervening variable on the two types of trials. We can then compare these predicted similarities to the similarity structure that RSA extracts from the BOLD data. If the similarities predicted by the model and the similarity structure derived from the BOLD responses in some ROI are qualitatively similar, then RSA concludes that this ROI may

play a key role in computing the value of the hypothesized intervening variable.

RSA is conceptually simple. The first step is to compute a representational dissimilarity matrix (RDM), which includes a row and column for every event, condition, ROI, or task, depending on what type of similarity structure we want to construct. For the present purposes, there are three obvious possibilities. One is that the RDM will include dissimilarities between all possible pairs of activity patterns estimated from a voxel-based encoding model (i.e., rows of $\widehat{\mathbf{A}}$). Another possibility is that the RDM is estimated directly from the BOLD data in some ROI for the same events that were used to create the activity patterns. Finally, a third possibility is that the RDM is constructed from some other type of mathematical model – for example, a traditional model of perceptual or cognitive processing from the mathematical psychology literature. However the RDM is created, it is assumed to include numerical data that define the similarity structure describing how the various events are related.

The RDM is sometimes used to build a similarity structure using some form of multidimensional scaling. But in most applications, two RDMs of the same task are directly compared. For example, RSA is often used to test the validity of an encoding model by testing statistically whether the RDM predicted by the model is consistent with an empirical RDM estimated is some ROI from our fMRI data.

4.1 Estimating an RDM

An RDM is estimated by computing the dissimilarity in the model predictions or data for all possible pairs of stimulus types (or more generally, event types). If there are N_S different stimuli, then these dissimilarities are collected in an RDM of order $N_S \times N_S$. The entry in row i and column j is the observed (in the case of BOLD data) or predicted (in the case of a model) dissimilarity between the response to stimulus types i and j. Denote this dissimilarity by $d(\mathrm{S}_i, \mathrm{S}_j)$.

In the case of BOLD data, $d(\mathrm{S}_i, \mathrm{S}_j)$ is computed by comparing rows of the BOLD activity matrix $\tilde{\mathbf{B}}$. Recall that $\tilde{\mathbf{B}}$ is an $N_S \times N_v$ matrix in which row i and column k contains the estimated amplitude of the BOLD response to stimulus S_i in voxel k of the ROI. Therefore, row i is a vector describing the response of the ROI to stimulus S_i. In the case of voxel-based encoding models, the predicted aggregate activity vector $\mathbf{A}$ replaces $\tilde{\mathbf{B}}$. In the case of more traditional mathematical psychology models, the RDM is computed by comparing predictions of the model – usually on the intervening variable of interest – on all possible pairs of stimulus trials.

Let $\boldsymbol{a}_i^\top$ denote the i^{th} row of the aggregate activity matrix $\mathbf{A}$ predicted by some voxel-based encoding model. Then $d(\mathrm{S}_i, \mathrm{S}_j)$ is an estimate of the dissimilarity of $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$. The concept of similarity is fundamentally important in almost every scientific field. And across these different fields, similarity and dissimilarity are defined in many different ways. In RSA, the choice of the best dissimilarity measure is still an area of active research (Bobadilla-Suarez et al., 2020). Most applications however, have used one of three different measures: one minus the Pearson correlation, a Euclidean measure, or a Mahalanobis-distance measure.

As the name suggests, one minus the Pearson correlation equals

$$d_{\mathrm{P}}(\mathrm{S}_i, \mathrm{S}_j) = 1 - r(\boldsymbol{a}_i, \boldsymbol{a}_j), \tag{39}$$

where $r(\boldsymbol{a}_i, \boldsymbol{a}_j)$ is the Pearson correlation between the entries in $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$. The Euclidean measure is defined as the squared Euclidean distance between $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$:

$$d_{\mathrm{E}}(\mathrm{S}_i, \mathrm{S}_j) = (\boldsymbol{a}_i - \boldsymbol{a}_j)^\top (\boldsymbol{a}_i - \boldsymbol{a}_j). \tag{40}$$

Mahalanobis dissimilarity is based on the assumption that the underlying data are samples from a multivariate normal distribution. The Mahalanobis dissimilarity between activity vectors $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$, which is defined as the squared Mahalanobis distance between the vectors, equals

$$d_{\mathrm{M}}(\mathrm{S}_i, \mathrm{S}_j) = (\boldsymbol{a}_i - \boldsymbol{a}_j)^\top \, \widehat{\boldsymbol{\Sigma}}^{-1} \, (\boldsymbol{a}_i - \boldsymbol{a}_j), \tag{41}$$

where $\widehat{\Sigma}^{-1}$ is an estimate of the (spatial) variance-covariance matrix of the activity vectors.

One weakness of all these measures is that if two activity vectors are identical at the population level, and therefore their distance apart is zero, then noise can only increase the distance between them. Therefore, under the null hypothesis that two event types elicit identical activity patterns, all of these distance measures will produce biased estimates of the true difference. One solution to this problem is to use cross-validated Mahalanobis distance, or crossnobis distance (Allefeld and Haynes, 2014). The crossnobis distance is computed by dividing the data into Q independent partitions, and using a leave-one-partition-out scheme. Let $\boldsymbol{a}_i(q)$ denote the i^{th} activity pattern computed from the data in partition q, and $\boldsymbol{a}_i(\neg q)$ denote the same activity computed from the data in all partitions other than q. Then the cross-validated Mahalanobis distance – that is, the crossnobis distance – between the activity vectors associated with stimuli S_i and S_i equals

$$d_{\mathrm{CN}}(\mathrm{S}_i, \mathrm{S}_j) = \frac{1}{Q} \sum_{q=1}^{Q} [\boldsymbol{a}_i(q) - \boldsymbol{a}_j(q)]^\top \, \widehat{\boldsymbol{\Sigma}}^{-1} \, [\boldsymbol{a}_i(\neg q) - \boldsymbol{a}_j(\neg q)]. \tag{42}$$

Note that because $[\boldsymbol{a}_i(k) - \boldsymbol{a}_j(k)]$ and $[\boldsymbol{a}_i(\neg k) - \boldsymbol{a}_j(\neg k)]$ are computed from different data partitions, the crossnobis distance $d_{\mathrm{CN}}(\mathrm{S}_i, \mathrm{S}_j)$ could be either positive or negative. In contrast, of course, regular Euclidean and Mahalanobis distance must both always be non-negative. The advantage of crossnobis distance is that it eliminates bias. More specifically, under the null hypothesis that two events elicit the same pattern of activation, the mean crossnobis distance between the resulting activity vectors is zero, whereas with regular Euclidean or Mahalanobis distance, this mean is greater than zero (Allefeld and Haynes, 2014). Furthermore, Walther et al. (2016) compared all of these measures on simulated and real fMRI data. The most reliable method was crossnobis distance. Even so, the choice of the best dissimilarity measure is still an area of active research. While crossnobis distance has the appealing property of being unbiased and has been shown to be more reliable than other measures, some researchers have recently argued that the one-minus-Pearson-correlation measure is preferable (Bobadilla-Suarez et al., 2020).

5 Testing Encoding Models Against Behavioral Data

The introduction to this chapter claimed that many of the identifiability problems that plague computational models of behavior could be alleviated by extending tests of the models to fMRI data. However, we also saw that encoding models have their own identifiability problems that complicate their interpretation. Even so, it now seems clear that an integrative approach, in which behavioral and neuroimaging data are both addressed within the same modeling framework, would be beneficial in both mathematical psychology and computational neuroimaging (Soto, 2019).

There are at least three ways in which encoding models can be tested against behavioral data. First, we can use encoding models that are grounded in neuroscience to predict behavioral data. Second, we can fit a cognitive model to behavioral data, build an encoding model in which the encoding channels compute the intervening variables hypothesized by the cognitive model, and then test the resulting encoding model against fMRI data. This approach is known as model-based fMRI (O'Doherty et al., 2007). Third, we can jointly model fMRI and behavioral data in a truly integrative approach that constrains inferences about a single model with both types of data. We now briefly describe each of these approaches.

5.1 Encoding/Decoding Observer Models

One way to build an encoding model that makes simultaneous neural and behavioral predictions is to generalize any of the voxel-based encoding models described earlier in a way that allows them to make behavioral predictions. In all of those models, the population neural response vector $\mathbf{r}$

is assumed to be available to downstream neurons to decode useful behavioral information about the stimulus. So to make behavioral predictions, two additional problems must be solved. First, a choice must be made about which of a variety of possible decoding schemes is incorporated into the model (e.g., Lehky et al., 2013; Pouget et al., 1998; Salinas and Abbott, 1994; Seung and Sompolinsky, 1993). Second, assumptions must be made about how the model uses the decoded stimulus information to select a response. We refer to encoding models that add a decoding scheme and response selection assumptions as *encoding/decoding observer models*.

As an illustration of this approach, consider a simple identification task in which the stimuli vary on a single physical dimension (e.g., as in Eq. 4). For example, the stimuli might all be Gabor patterns that vary only on orientation or spatial frequency. The question of which decoding scheme to use is complicated somewhat by the fact that some schemes lead to an inherent ambiguity in whether an observed behavioral change is due to encoding versus decoding changes (Gold and Ding, 2013). Confronted with this dilemma, many modelers have assumed optimal decoding via maximum likelihood estimation (e.g., Dakin et al., 2005; Deneve et al., 1999; Hays and Soto, 2020; Ling et al., 2009; May and Solomon, 2015; Paradiso, 1988; Series et al., 2009; Soto et al., 2021). This assumption leads to the decoding scheme in which observation of the neural response vector $\mathbf{r}$ causes the model to infer that the value of the presented stimulus was $\hat{s}$, where:

$$\hat{s} = \arg\max_s \hat{P}(s|\mathbf{r}, \boldsymbol{\theta}), \tag{43}$$

and as usual, $\boldsymbol{\theta}$ is a vector of channel tuning parameters.

If neural noise is independent across channels, then

$$P(s|\mathbf{r}, \boldsymbol{\theta}) = \prod_{c=1}^{N_c} P(s|r_c, \boldsymbol{\theta}), \tag{44}$$

and therefore, the log-likelihood is maximized when:

$$\hat{s} = \arg\max_s \sum_{c=1}^{N_c} \ln \hat{P}(s|r_c, \boldsymbol{\theta}). \tag{45}$$

There is usually a single optimal solution for a well-posed statistical problem such as this, which therefore avoids the ambiguities mentioned above about whether behavioral changes are caused by encoding or decoding mechanisms. An additional advantage is that the asymptotic properties of maximum likelihood estimates are well known. In particular, maximum likelihood estimators are asymptotically normal, and if noise is independent and identically distributed across channels, then the maximum likelihood estimator $\hat{s}$ of the true stimulus value s_0 has an asymptotic normal distribution with mean s_0 and variance

$$\sigma_{\hat{s}}^2 = [n\, I(s_0)]^{-1},$$

where $I(s_0)$ is the Fisher information, and n is the number of channels (e.g., Van der Vaart 2000).

Note that this variance can be directly computed if an analytical form for the Fisher information is known, which is the case for the standard encoding model with Gaussian tuning functions that all have identical width ω (i.e., see Eq. 4). When, in addition, neural noise is Poisson and independent, the Fisher information is given by (Dayan and Abbott, 2001; Pouget et al., 1998; Seung and Sompolinsky, 1993):

$$\begin{aligned} I(s) &= \sum_{c=1}^{N} \frac{[f_c'(s)]^2}{f_c(s)} \\ &= \sum_{c=1}^{N} \frac{r^{max}(s - s_c)^2}{\omega^4} \exp\left[-\frac{1}{2}\left(\frac{s - s_c}{\omega}\right)^2\right], \end{aligned} \tag{46}$$

where $f_c(s)$ is the Gaussian tuning function of Eq. 4 and $f_c'(s)$ is its derivative with respect to s. For Gaussian neural noise with fixed variance σ_r^2, the Fisher information is given by (Pouget et al., 1998):

$$\begin{aligned} I(s) &= \frac{1}{\sigma_r^2} \sum_{c=1}^{N} f_c'(s)^2 \\ &= \frac{1}{\sigma_r^2} \sum_{c=1}^{N} \frac{r^{max} (s - s_c)^2}{\omega^4} \exp\left[-\left(\frac{s - s_c}{\omega} \right)^2 \right]. \end{aligned} \tag{47}$$

When $I(s)$ is unknown, which is likely to be the case for many encoding models, $\sigma_{\hat{s}}^2$ can be directly estimated through Monte Carlo simulation (e.g., Dakin et al., 2005; Ling et al., 2009; Hays and Soto, 2020).

Another advantage of assuming that decoding is optimal is that it allows encoding/decoding observer models to be linked to psychophysical measures in a straightforward manner. For example, the distribution of $\hat{s}$ could be interpreted as the the distribution of perceptual evidence assumed by Gaussian signal detection theory (Ashby & Wenger, Chapter 7, this volume; Green and Swets 1966; Macmillan and Creelman 2005), which links the encoding/decoding observer model to popular measures such as d' and sensory thresholds. For example, consider a two-stimulus identification task with stimuli that have values s_1 and s_2. Suppose we use these asymptotic results to compute the mean $\mu_{\hat{s}}$ and variance $\sigma_{\hat{s}}^2$ of the distribution of estimates for each stimulus, either through analytical expressions or Monte Carlo simulation. From these values, it is easy to compute the model's predicted d' for the identification task (Soto et al., 2021):

$$d' = \frac{\mu_{\hat{s}_1} - \mu_{\hat{s}_2}}{\sqrt{.5 \left(\sigma_{\hat{s}_1}^2 + \sigma_{\hat{s}_2}^2 \right)}}.$$

Note that $I(s)$ is a function of the stimulus value, so the variance of decoded values might change when different stimuli are presented. However, most researchers assume that it remains the same across values of the decoded variable, in line with the equal-variance signal detection model.

The methods used to create encoding/decoding observer models allow behavioral predictions to be generated from almost any encoding model that has either been fitted to neural data or constrained by it. For example, Goris et al. (2013) showed that an encoding/decoding observer model constrained by what is known about encoding of spatial frequency in primary visual cortex does an excellent job at predicting pattern detection behavior. In principle, any well-defined encoding model can serve as a model of behavior with relatively minor adjustments.

Equations 28 – 31 showed that many different sets of encoding channels make identical predictions. This can make it difficult to draw strong inferences about why some change occurred in a population response. One way to resolve these ambiguities is to explore various alternatives by formulating them as hypotheses that make distinct behavioral predictions in some psychophysical task. Simulation work has shown that when combined with inverted encoding modeling, only a couple of psychophysical experiments are sufficient to arbitrate between major hypotheses about changes in neural encoding (Hays and Soto, 2020).

Signal detection theory has been an invaluable model, not only in perceptual tasks, but also in cognitive tasks such as recognition memory (e.g., Wixted, 2007), causal and contingency learning (e.g., Siegel et al., 2009), generalization (Blough, 1967, e.g.), and metacognition (e.g., Maniscalco and Lau, 2012, 2014). For this reason, the methods that have been successfully used to link encoding models to psychophysics in the vision literature might prove useful in other research areas as well.

5.2 Model-Based FMRI

All of the encoding models considered so far were designed specifically with the goal of modeling fMRI data. But fMRI data can also be used to provide unique tests of cognitive-based mathematical models that are more traditional within mathematical psychology. The methods that have been developed to test the validity of purely behavioral computational models against fMRI data are known as *model-based fMRI* (O'Doherty et al., 2007).

Purely behavioral models are those that make no neuroscience predictions. Instead, they typically make predictions about how a participant will respond to a stimulus by appealing to some hypothetical constructs or latent (intervening) variables, such as, for example, memory, attention, or similarity. The models are tested against behavioral data by examining their ability to account for dependent variables such as response accuracy and response time. A good fit provides only indirect support for the model and its hypothesized latent variables – in part, because of the identifiability problems described earlier. Model-based fMRI provides an opportunity to improve model identifiability by offering a method to examine the latent variables more directly. The basic idea is to estimate the free parameters of the model by fitting it to the available behavioral data – in exactly the same way that the model is typically applied. Next, the parameter estimates that result are used to derive predictions from the model about one or more latent variables, and finally these predictions are compared to the observed BOLD responses from various brain regions (e.g., by using the GLM). For example, consider an exemplar model that predicts trial-by-trial categorization responses are determined by certain specific similarity computations. In model-based fMRI, the critical similarity value predicted by the model is computed on every trial and then correlated with trial-by-trial observed BOLD responses, either across the whole brain or in specific brain regions. Finding a region where the correlation is high accomplishes two goals. First, it provides empirical support for the model that is impossible with purely behavioral data because it suggests that changes in neural activity in some brain region are consistent with changes in a latent variable that the model predicts is critical to the task under study. Second, a good fit identifies brain regions that might possibly mediate the processes hypothesized by the model. Since the models are perceptual or cognitive, this allows an important first step in extending them to the neural level.

In the ideal application, the constructs that are tested against fMRI data change significantly from trial to trial. For example, consider a model that assumes participants compare the presented stimulus to some internally constructed decision criterion and give one response if the criterion is exceeded and a different response if it is not (e.g., as in signal detection theory). A model that predicts the numerical value of this criterion on every trial could be tested against fMRI data by correlating the predicted criterion value against the BOLD response observed in different brain regions. However, if the experimental design is such that the model predicts only slow changes in the criterion during the scanning session, then these correlations will not provide strong tests of the model because the predicted BOLD responses in criterion-setting regions will be similar to the BOLD responses in task-inactive brain regions.

After some model-predicted hypothetical constructs are selected that vary significantly from trial to trial, a typical model-based fMRI analysis would include the following steps. First, the model is fit to the behavioral data collected during the functional run separately for each participant. The primary purpose of this step is to estimate the free parameters in the model. Since the model being tested is purely behavioral, it makes no predictions about neural activations or BOLD responses, and as a result, its parameters should only be estimated by fitting to behavioral data.

The second step is to use the parameter estimates from step one to compute numerical values of the intervening variables from the model that were identified earlier to test against the fMRI data. The goal here is to identify brain regions in which changes in the BOLD responses are predicted by changes in the variables. In the case of the exemplar model, obvious candidates include the

predicted summed similarity of the presented stimulus to each of the contrasting categories.

Step three is to construct a model of the BOLD response from each of the selected model variables. The standard approach is to first construct a boxcar function of square waves for each variable. The height of this function is set to zero when the variable is predicted to be inactive and to the value of the variable when it is active. For example, in the case of the exemplar model's predcited summed similarity to some category A, the boxcar function would equal zero between trials and its height would equal the predicted summed similarity to exemplars from category A during the time beginning with each stimulus onset and ending with the participant's response. After this boxcar function is built, predicted BOLD responses are computed by convolving the boxcar function with some model of the hrf (as in Eq. 21).

Step four is to correlate each of these predicted BOLD responses with the observed BOLD response in every voxel via the GLM. Voxels where the correlation is high are identified as being sensitive to that variable (for a more thorough description of all these steps see, e.g., Ashby 2019).

In summary, a model-based fMRI analysis of this type: 1) tests the model against a new dependent variable (i.e., the BOLD response); 2) potentially makes the model's latent variables observable; 3) identifies brain regions sensitive to the model's latent variables; and 4) provides valuable data that could be used to develop a neurocomputational version of the model.

5.3 Joint Neural and Behavioral Modeling

Encoding/decoding observer models are neural models in which some assumptions are added that allow tests against behavioral data. In contrast, model-based fMRI is an approach in which assumptions are added to purely behavioral models that allow tests against fMRI data. A third way in which encoding models can be tested against behavioral data is to build models that directly account for both neuroscience and behavioral data. There are two general approaches to joint modeling of this kind – one based in neuroscience and one based in statistics. Their main advantage is that they use variation in both behavioral and neural data to jointly and equally constrain inferences about encoding models.

The neuroscience approach comes from the emerging field of computational cognitive neuroscience (CCN), which is a new field that lies at the intersection of computational neuroscience, machine learning, and neural network theory (i.e., connectionism). The goal here is to build biologically detailed neural network models in which the simulated regions and their interconnections are faithful to known neuroanatomy. The units that define the network are either simulated spiking neurons or populations of similar neurons (e.g., a cortical column), in which case the primary dependent variables are the firing rates of each population. Theoretically at least, CCN models can account for all levels of a behavioral phenomenon from single-neuron spiking up to behavior (Ashby, 2018; O'Reilly and Munakata, 2000). In particular, a good CCN model should predict how neural activity changes in a variety of different brain regions as the subject performs the task under study, and at the same time make predictions about the most widely studied behavioral dependent variables, including response accuracy and response time. In general, testing CCN models against fMRI data follows the same basic steps as in model-based fMRI. For a description of the special issues that arise due to the extra neuroscience details of CCN models, see Ashby (2019).

The statistical approach to joint modeling uses a hierarchical Bayesian inferential framework to model the statistical relations between neural and behavioral measures directly within a single model (Palestro et al., 2018; Turner, 2015; Turner et al., 2013). To keep the presentation concrete, consider an identification experiment in which participants are presented with one of two stimuli on each trial, S_1 and S_2, and their task is to report which of the two stimuli was presented. Model performance in this task will depend on the specific stimuli that are presented and their base rates, which can be collected in the set $\mathcal{S} = \{S_1, S_2, P(S_1), P(S_2)\}$. The neural dependent variables are the amplitudes of the BOLD responses to the two stimuli, collected in the 2×1 vector $\tilde{\boldsymbol{b}}$, and

the behavioral dependent variables are the proportion of correct responses on S_1 trials and on S_2 trials, which can be collected in a 2×1 vector $\boldsymbol{o}$. Finally, we assume that the BOLD responses are related to the channel responses according to the linearized encoding model of Eq. 15.

To build a joint model, we begin by computing the likelihood of the fMRI data, $P(\tilde{\boldsymbol{b}}|\mathbf{R}, \boldsymbol{\beta}, \mathcal{S})$, where $\boldsymbol{\beta}$ represents a vector of parameters from the neural measurement model. For example, in the linearized encoding model, $\boldsymbol{\beta}$ would include the weight parameters in $\boldsymbol{w}$ as well as the variance-covariance matrix of measurement noise $\boldsymbol{\Sigma}_m$. Second, we compute the likelihood of the behavioral data, $P(\boldsymbol{o}|\mathbf{R}, \boldsymbol{\gamma}, \mathcal{S})$, where $\boldsymbol{\gamma}$ is a vector of parameters from the behavioral measurement model. Both of these likelihoods depend directly on the random population response $\mathbf{R}$, which has a distribution $P(\mathbf{R}|\boldsymbol{\theta}, \mathcal{S})$ specified either by Eq. 2 or 3, and that depends on the encoding model parameters and the stimulus set $\mathcal{S}$ (we omit state variables for simplicity). Finally, the model should formalize prior distributions over all the parameters included in $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$, which would depend on hyperparameters $\boldsymbol{\Omega}$. With this, the model is fully specified and the joint posterior distribution of the model parameters can be expressed as:

$$P\left(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\gamma}|\tilde{\boldsymbol{b}}, \boldsymbol{o}\right) \propto P\left(\tilde{\boldsymbol{b}}|\mathbf{R}, \boldsymbol{\beta}, \mathcal{S}\right) P\left(\boldsymbol{o}|\mathbf{R}, \boldsymbol{\gamma}, \mathcal{S}\right) P\left(\mathbf{R}|\boldsymbol{\theta}, \mathcal{S}\right) P\left(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\gamma}|\boldsymbol{\Omega}\right). \tag{48}$$

In general, this distribution can be approximated using any of a wide range of available sampling algorithms (Gilks et al., 1996, see).

Under the assumption that the BOLD responses are related to the channel responses according to the linearized encoding model of Eq. 15, then the likelihood of the BOLD amplitude $P\left(\tilde{\boldsymbol{b}}|\mathbf{R}, \boldsymbol{\beta}, \mathcal{S}\right)$ is multivariate Gaussian with mean $\mathrm{E}[\mathbf{R}]\boldsymbol{w}$ (i.e., see Eq. 15) and variance-covariance matrix $\boldsymbol{\Sigma}_m$. The priors over $\boldsymbol{w}$ and $\boldsymbol{\Sigma}_m$ can be chosen to match the regularization algorithms used in past applications of encoding modeling (Diedrichsen and Kriegeskorte, 2017), or to be conjugate for the likelihood function, which facilitates inference. The likelihood of the behavioral data $P(\boldsymbol{o}|\mathbf{R}, \boldsymbol{\gamma}, \mathcal{S})$ can be obtained by linking the encoding model to signal detection theory in the way described earlier in this section. In this approach, an optimal decoder is used to obtain estimates of the noise in the decoded stimuli. With the addition of a threshold parameter, one can obtain the likelihood of each possible response on a given trial from the cumulative normal distribution. As before, priors can be chosen following previous applications of signal detection theory that have used a Bayesian framework, or to be conjugate to the likelihood function. Finally, the distribution of population responses $P(\mathbf{R}|\boldsymbol{\theta}, \mathcal{S})$ will depend on our choice of tuning functions and neural noise, and priors can be chosen to be conjugate to that distribution, or based on previous applications (Sadil et al., 2021; Van Bergen et al., 2015).

6 Conclusions

Mathematical psychologists build and test mathematical models of perceptual, cognitive, and motor behaviors. A common goal is to develop models that describe the underlying processes that are presumed to mediate the behavior under study. When tested in the traditional way – that is, against behavioral measures such as response accuracy and response time – these processes are almost always unobservable. One common barrier that limits progress in this field is that models postulating very different psychological processes can often provide a similarly good quantitative fit to the behavioral data. For example, because of such nonidentifiabilities, many subfields are still debating the validity of competing models that were proposed 40 and 50 years ago.

Testing these models against fMRI BOLD data offers the hope of greatly improving model identifiability. And, because of model-based fMRI, this is true even for models that include no neuroscience detail. Any model that makes predictions about psychological processes that are unobservable with behavioral data could benefit from testing via model-based fMRI, at least so

long as those predictions change significantly trial-by-trial. For example, if two competing models account for behavioral data about equally well, then we should favor the model that makes predictions about trial-by-trial changes in some psychological process that track changes in the BOLD response of some brain region, over the model that makes process predictions that are not mirrored by BOLD data.

As another example that does not depend on model-based fMRI, suppose some cognitive theory predicts that the same perceptual and cognitive processes mediate performance in two different tasks. Then this theory should predict similar patterns of activation in an fMRI study of the two tasks, even if the theory makes no predictions about what those activation patterns should look like. If an RSA concludes that the activation patterns in the two tasks are qualitatively different, then the theory probably needs some significant revision.

Although the number likely decreases every year, there are still many cognitive scientists who are deeply skeptical of fMRI – some even characterizing it as a new form of phrenology (Dobbs, 2005; Uttal, 2001). Even so, recent methodological advancements, such as model-based fMRI and RSA, show that fMRI can provide useful and powerful new tests of models – even purely cognitive models – that would have been considered a fantasy just a few decades ago.

7 Related Literature

For a thorough description of virtually all statistical methods for analyzing fMRI BOLD data – including traditional GLM approaches, as well as encoding and decoding methods, RSA, and DCM – see Ashby (2019).

An introduction to encoding and decoding from a computational neuroscience perspective can be found in Pouget et al. (2003) and Dayan and Abbott (2001). For an introduction to applications of encoding models to neuroimaging, see van Gerven (2017).

Decoding analyses of neuroimaging data using machine-learning algorithms (e.g., MVPA) rather than explicit encoding modeling are covered by Pereira et al. (2009). Kriegeskorte and Diedrichsen (2019) summarize recent work on RSA and its relation to encoding modeling (see also Diedrichsen and Kriegeskorte 2017). May and Solomon (2015) describe encoding/decoding observer modeling in detail, and O'Doherty et al. (2007) does the same for model-based fMRI. Palestro et al. (2018) give a tutorial introduction to joint modeling of neural and behavioral data using a hierarchical Bayesian framework.

8 Acknowledgments

We thank Thomas Sprague, Justin Gardner, and Joshua Ryu for their helpful comments on the manuscript.

References

Allefeld, C. and Haynes, J. D. (2014). Searchlight-based multi-voxel pattern analysis of fMRI by cross-validated MANOVA. *NeuroImage*, 89:345–357.

Anzellotti, S. and Caramazza, A. (2014). The neural mechanisms for the recognition of face identity in humans. *Frontiers in Psychology*, 5:672.

Ashby, F. G. (1992). Multidimensional models of categorization. In Ashby, F. G., editor, *Multidimensional models of perception and cognition*, pages 449–483. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Ashby, F. G. (2018). Computational cognitive neuroscience. In Batchelder, W., Colonius, H., Dzhafarov, E., and Myung, J., editors, *New handbook of mathematical psychology, Volume 2*, pages 223–270. New York: Cambridge University Press.

Ashby, F. G. (2019). *Statistical analysis of fMRI data, Second edition*. Cambridge MA: MIT press.

Ashby, F. G., Ennis, J. M., and Spiering, B. J. (2007). A neurobiological theory of automaticity in perceptual categorization. *Psychological Review*, 114(3):632–656.

Ashby, F. G. and Maddox, W. T. (1994). A response time theory of separability and integrality in speeded classification. *Journal of Mathematical Psychology*, 38(4):423–466.

Bickel, P. J. and Li, B. (2006). Regularization in statistics. *Test*, 15(2):271–344.

Blough, D. S. (1967). Stimulus generalization as signal detection in pigeons. *Science*, 158(3803):940–941.

Bobadilla-Suarez, S., Ahlheim, C., Mehrotra, A., Panos, A., and Love, B. C. (2020). Measures of neural similarity. *Computational Brain & Behavior*, 3:369–383.

Boynton, G. M., Engel, S. A., Glover, G. H., and Heeger, D. J. (1996). Linear systems analysis of functional magnetic resonance imaging in human V1. *Journal of Neuroscience*, 16(13):4207–4221.

Brouwer, G. J. and Heeger, D. J. (2009). Decoding and reconstructing color from responses in human visual cortex. *The Journal of Neuroscience*, 29(44):13992–14003.

Buhmann, M. D. (2003). *Radial basis functions: Theory and implementations*, volume 12. Cambridge, MA: Cambridge University Press.

Buxton, R. B. (2013). The physics of functional magnetic resonance imaging (fMRI). *Reports on Progress in Physics*, 76(9):096601.

Buxton, R. B., Wong, E. C., and Frank, L. R. (1998). Dynamics of blood flow and oxygenation changes during brain activation: The balloon model. *Magnetic Resonance in Medicine*, 39(6):855–864.

Byers, A. and Serences, J. T. (2014). Enhanced attentional gain as a mechanism for generalized perceptual learning in human visual cortex. *Journal of Neurophysiology*, 112(5):1217–1227.

Carandini, M. and Heeger, D. J. (2012). Normalization as a canonical neural computation. *Nature Reviews Neuroscience*, 13(1):51–62.

Carlson, T. A., Ritchie, J. B., Kriegeskorte, N., Durvasula, S., and Ma, J. (2013). Reaction time for object categorization is predicted by representational distance. *Journal of Cognitive Neuroscience*, 26(1):132–142.

Chen, C. T. (1970). *Introduction to linear systems theory*. New York: Holt, Rinehart and Winston.

Chen, Y., Geisler, W. S., and Seidemann, E. (2006). Optimal decoding of correlated neural population responses in the primate visual cortex. *Nature Neuroscience*, 9(11):1412–1420.

Çukur, T., Nishimoto, S., Huth, A. G., and Gallant, J. L. (2013). Attention during natural vision warps semantic representation across the human brain. *Nature Neuroscience*, 16(6):763–770.

Dakin, S. C., Mareschal, I., and Bex, P. J. (2005). Local and global limitations on direction integration assessed using equivalent noise analysis. *Vision Research*, 45(24):3027–3049.

Dayan, P. and Abbott, L. F. (2001). *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. MIT Press.

DeCharms, R. C. (2008). Applications of real-time fMRI. *Nature Reviews Neuroscience*, 9(9):720–729.

Deneve, S., Latham, P. E., and Pouget, A. (1999). Reading population codes: a neural implementation of ideal observers. *Nature Neuroscience*, 2(8):740–745.

Diedrichsen, J. (2020). Representational models and the feature fallacy. In Poeppel, D., Mangun, G. R., and Gazzaniga, M. S., editors, *The Cognitive Neurosciences, 6th Edition*, pages 669–678. Cambridge, MA: MIT Press.

Diedrichsen, J. and Kriegeskorte, N. (2017). Representational models: A common framework for understanding encoding, pattern-component, and representational-similarity analysis. *PLOS Computational Biology*, 13(4):e1005508.

Dobbs, D. (2005). Fact or phrenology? *Scientific American Mind*, 16(1):24–31.

Dumoulin, S. O. and Wandell, B. A. (2008). Population receptive field estimates in human visual cortex. *NeuroImage*, 39(2):647–660.

Eickenberg, M., Gramfort, A., Varoquaux, G., and Thirion, B. (2017). Seeing it all: Convolutional network layers map the function of the human visual system. *NeuroImage*, 152:184–194.

Ester, E. F., Anderson, D. E., Serences, J. T., and Awh, E. (2013). A neural measure of precision in visual working memory. *Journal of Cognitive Neuroscience*, 25(5):754–761.

Ester, E. F., Sprague, T. C., and Serences, J. T. (2015). Parietal and frontal cortex encode stimulus-specific mnemonic representations during visual working memory. *Neuron*, 87(4):893–905.

Ester, E. F., Sprague, T. C., and Serences, J. T. (2020). Categorical biases in human occipitoparietal cortex. *Journal of Neuroscience*, 40(4):917–931.

Fracasso, A., Dumoulin, S. O., and Petridou, N. (2021). Point-spread function of the BOLD response across columns and cortical depth in human extra-striate cortex. *Progress in Neurobiology*, page 102034.

Friston, K. J., Harrison, L., and Penny, W. (2003). Dynamic causal modelling. *NeuroImage*, 19(4):1273–1302.

Fukunaga, K. (2013). *Introduction to statistical pattern recognition, Second edition*. San Diego, CA: Academic Press.

Garcia, J. O., Srinivasan, R., and Serences, J. T. (2013). Near-real-time feature-selective modulations in human cortex. *Current Biology*, 23(6):515–522.

Gardner, J. L. and Liu, T. (2019). Inverted encoding models reconstruct an arbitrary model response, not the stimulus. *eNeuro*, 6(2):e0363–18.2019.

Gilks, W. R., Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov chain Monte Carlo in practice*. Chapman & Hall, London.

Gold, J. I. and Ding, L. (2013). How mechanisms of perceptual decision-making affect the psychometric function. *Progress in Neurobiology*, 103:98–114.

Goris, R. L. T., Putzeys, T., Wagemans, J., and Wichmann, F. A. (2013). A neural population model for visual pattern detection. *Psychological Review*, 120(3):472–496.

Green, D. M. and Swets, J. A. (1966). *Signal detection theory and psychophysics*. Wiley, New York.

Grootswagers, T., Cichy, R. M., and Carlson, T. A. (2018). Finding decodable information that can be read out in behaviour. *NeuroImage*, 179:252–262.

Güçlü, U. and van Gerven, M. A. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014.

Hastie, T., Tibshirani, R., and Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. NY: Springer.

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

Hays, J. S. and Soto, F. A. (2020). Changes within neural population codes can be inferred from psychophysical threshold studies. *bioRxiv*, page 2020.03.26.010900.

Huth, A. G., Nishimoto, S., Vu, A. T., and Gallant, J. L. (2012). A continuous semantic space describes the representation of thousands of object and action categories across the human brain. *Neuron*, 76(6):1210–1224.

Kaplan, J. T., Man, K., and Greening, S. G. (2015). Multivariate cross-classification: Applying machine learning techniques to characterize abstraction in neural representations. *Frontiers in Human Neuroscience*, 9:151.

Kay, K. N., Naselaris, T., Prenger, R. J., and Gallant, J. L. (2008). Identifying natural images from human brain activity. *Nature*, 452(7185):352–355.

Kriegeskorte, N. and Diedrichsen, J. (2019). Peeling the onion of brain representations. *Annual Review of Neuroscience*, 42:407–432.

Kriegeskorte, N. and Douglas, P. K. (2019). Interpreting encoding and decoding models. *Current Opinion in Neurobiology*, 55:167–179.

Kriegeskorte, N., Mur, M., and Bandettini, P. (2008). Representational similarity analysis–Connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2:4.

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review*, 99(1):22–44.

Lee, S., Papanikolaou, A., Logothetis, N. K., Smirnakis, S. M., and Keliris, G. A. (2013). A new method for estimating population receptive field topography in visual cortex. *NeuroImage*, 81:144–157. Publisher: Elsevier.

Lehky, S. R., Sereno, M. E., and Sereno, A. B. (2013). Population coding and the labeling problem: extrinsic versus intrinsic representations. *Neural Computation*, 25(9):2235–2264.

Ling, S., Liu, T., and Carrasco, M. (2009). How spatial and feature-based attention affect the gain and tuning of population responses. *Vision Research*, 49(10):1194–1204.

Liu, T., Cable, D., and Gardner, J. L. (2018). Inverted encoding models of human population response conflate noise and neural tuning width. *Journal of Neuroscience*, 38(2):398–408.

Macmillan, N. A. and Creelman, C. D. (2005). *Detection theory: A user's guide*. Lawrence Erlbaum Associates, Mahwah, NJ, 2nd edition.

Maniscalco, B. and Lau, H. (2012). A signal detection theoretic approach for estimating metacognitive sensitivity from confidence ratings. *Consciousness and Cognition*, 21(1):422–430.

Maniscalco, B. and Lau, H. (2014). Signal detection theory analysis of type 1 and type 2 data: Meta-d', response-specific meta-d', and the unequal variance SDT model. In Fleming, S. M. and Frith, C. D., editors, *The Cognitive Neuroscience of Metacognition*, pages 25–66. Springer Berlin Heidelberg, Berlin, Heidelberg.

May, K. A. and Solomon, J. A. (2015). Connecting psychophysical performance to neuronal response properties I: Discrimination of suprathreshold stimuli. *Journal of Vision*, 15(6):8.

Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K.-M., Malave, V. L., Mason, R. A., and Just, M. A. (2008). Predicting human brain activity associated with the meanings of nouns. *Science*, 320(5880):1191–1195.

Mumford, J. A., Turner, B. O., Ashby, F. G., and Poldrack, R. A. (2012). Deconvolving BOLD activation in event-related designs for multivoxel pattern classification analyses. *NeuroImage*, 59(3):2636–2643.

Murdock, B. B. (1985). An analysis of the strength-latency relationship. *Memory & Cognition*, 13(6):511–521.

Naselaris, T., Kay, K. N., Nishimoto, S., and Gallant, J. L. (2011). Encoding and decoding in fMRI. *NeuroImage*, 56(2):400–410.

Naselaris, T., Prenger, R. J., Kay, K. N., Oliver, M., and Gallant, J. L. (2009). Bayesian reconstruction of natural images from human brain activity. *Neuron*, 63(6):902–915.

Nishimoto, S., Vu, A. T., Naselaris, T., Benjamini, Y., Yu, B., and Gallant, J. L. (2011). Reconstructing visual experiences from brain activity evoked by natural movies. *Current Biology*, 21(19):1641–1646.

Norman, K. A., Polyn, S. M., Detre, G. J., and Haxby, J. V. (2006). Beyond mind-reading: multi-voxel pattern analysis of fmri data. *Trends in Cognitive Sciences*, 10(9):424–430.

O'Doherty, J., Hampton, A., and Kim, H. (2007). Model-based fMRI and its application to reward learning and decision making. *Annals of the New York Academy of Sciences*, 1104(1):35–53.

Ogawa, S., Lee, T.-M., Kay, A. R., and Tank, D. W. (1990a). Brain magnetic resonance imaging with contrast dependent on blood oxygenation. *Proceedings of the National Academy of Sciences*, 87(24):9868–9872.

Ogawa, S., Lee, T.-M., Nayak, A. S., and Glynn, P. (1990b). Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields. *Magnetic Resonance in Medicine*, 14(1):68–78.

O'Reilly, R. C. and Munakata, Y. (2000). *Computational explorations in cognitive neuroscience: Understanding the mind by simulating the brain*. Cambridge, MA: MIT press.

Palestro, J. J., Bahg, G., Sederberg, P. B., Lu, Z. L., Steyvers, M., and Turner, B. M. (2018). A tutorial on joint models of neural and behavioral measures of cognition. *Journal of Mathematical Psychology*, 84:20–48.

Paradiso, M. A. (1988). A theory for the use of visual orientation information which exploits the columnar structure of striate cortex. *Biological Cybernetics*, 58(1):35–49.

Pedregosa, F., Eickenberg, M., Ciuciu, P., Thirion, B., and Gramfort, A. (2015). Data-driven HRF estimation for encoding and decoding models. *NeuroImage*, 104:209–220.

Pereira, F., Mitchell, T., and Botvinick, M. (2009). Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, 45(1):S199–S209.

Pouget, A., Dayan, P., and Zemel, R. (2000). Information processing with population codes. *Nature Reviews Neuroscience*, 1(2):125–132.

Pouget, A., Dayan, P., and Zemel, R. S. (2003). Inference and computation with population codes. *Annual Review of Neuroscience*, 26(1):381–410.

Pouget, A., Zhang, K., Deneve, S., and Latham, P. E. (1998). Statistically efficient estimation using population coding. *Neural Computation*, 10(2):373–401. Publisher: MIT Press.

Ren, Z., Li, J., Xue, X., Li, X., Yang, F., Jiao, Z., and Gao, X. (2021). Reconstructing seen image from brain activity by visually-guided cognitive representation and adversarial learning. *NeuroImage*, 228:117602.

Ritchie, J. B. and Carlson, T. A. (2016). Neural decoding and "inner" psychophysics: A distance-to-bound approach for linking mind, brain, and behavior. *Frontiers in Neuroscience*, 10.

Ritchie, J. B., Tovar, D. A., and Carlson, T. A. (2015). Emerging object representations in the visual system predict reaction times for categorization. *PLoS Computational Biology*, 11(6):e1004316.

Sadil, P., Huber, D. E., and Cowell, R. A. (2021). NeuroModulation Modeling (NMM): Inferring the form of neuromodulation from fMRI tuning functions. *bioRxiv*, page 2021.03.04.433362.

Salinas, E. and Abbott, L. F. (1994). Vector reconstruction from firing rates. *Journal of Computational Neuroscience*, 1(1):89–107.

Seeliger, K., Güçlü, U., Ambrogioni, L., Güçlütürk, Y., and van Gerven, M. A. J. (2018). Generative adversarial networks for reconstructing natural images from brain activity. *NeuroImage*, 181:775–785.

Serences, J. T., Saproo, S., Scolari, M., Ho, T., and Muftuler, L. T. (2009). Estimating the influence of attention on population codes in human visual cortex using voxel-based tuning functions. *NeuroImage*, 44(1):223–231.

Series, P., Stocker, A. A., and Simoncelli, E. P. (2009). Is the homunculus "aware" of sensory adaptation? *Neural Computation*, 21(12):3271–3304.

Seung, H. S. and Sompolinsky, H. (1993). Simple models for reading neuronal population codes. *Proceedings of the National Academy of Sciences*, 90(22):10749–10753.

Shen, G., Horikawa, T., Majima, K., and Kamitani, Y. (2019). Deep image reconstruction from human brain activity. *PLoS Computational Biology*, 15(1):e1006633.

Siegel, S., Allan, L. G., Hannah, S. D., and Crump, M. J. C. (2009). Applying signal detection theory to contingency assessment. *Comparative Cognition & Behavior Reviews*, 4:116–134.

Soto, F. A. (2019). Beyond the "Conceptual Nervous System": Can computational cognitive neuroscience transform learning theory? *Behavioural Processes*, 167:103908.

Soto, F. A. and Narasiwodeyar, S. (2021). Improving the validity of neuroimaging decoding tests of invariant and configural neural representation. *bioRxiv*, page 2020.02.27.967505.

Soto, F. A., Stewart, R. A., Hosseini, S., Hays, J. S., and Beevers, C. G. (2021). A computational account of the mechanisms underlying face perception biases in depression. *Journal of Abnormal Psychology*.

Soto, F. A., Vucovich, L. E., and Ashby, F. G. (2018). Linking signal detection theory and encoding models to reveal independent neural representations from neuroimaging data. *PLoS Computational Biology*, 14(10):e1006470.

Sprague, T. C., Adam, K. C. S., Foster, J. J., Rahmati, M., Sutterer, D. W., and Vo, V. A. (2018). Inverted encoding models assay population-level stimulus representations, not single-unit neural tuning. *eNeuro*, 5(3):ENEURO.0098–18.2018.

Sprague, T. C., Boynton, G. M., and Serences, J. T. (2019). The importance of considering model choices when interpreting results in computational neuroimaging. *eNeuro*, 6(6):e0196–19.2019.

Sprague, T. C. and Serences, J. T. (2013). Attention modulates spatial priority maps in the human occipital, parietal and frontal cortices. *Nature Neuroscience*, 16(12):1879–1887.

Tolhurst, D. J., Movshon, J. A., and Dean, A. F. (1983). The statistical reliability of signals in single neurons in cat and monkey visual cortex. *Vision Research*, 23(8):775–785.

Turner, B. M. (2015). Constraining cognitive abstractions through Bayesian modeling. In Forstmann, B. U. and Wagenmakers, E. J., editors, *An introduction to model-based cognitive neuroscience*, pages 199–220. NY: Springer.

Turner, B. M., Forstmann, B. U., Wagenmakers, E. J., Brown, S. D., Sederberg, P. B., and Steyvers, M. (2013). A Bayesian framework for simultaneously modeling neural and behavioral data. *NeuroImage*, 72:193–206.

Turner, B. O., Mumford, J. A., Poldrack, R. A., and Ashby, F. G. (2012). Spatiotemporal activity estimation for multivoxel pattern analysis with rapid event-related designs. *NeuroImage*, 62(3):1429–1438.

Uttal, W. R. (2001). *The new phrenology: The limits of localizing cognitive processes in the brain.* Cambridge, MA: The MIT press.

Van Bergen, R. S., Ma, W. J., Pratte, M. S., and Jehee, J. F. M. (2015). Sensory uncertainty decoded from visual cortex predicts behavior. *Nature Neuroscience*, 18(12):1728–1730.

Van der Vaart, A. W. (2000). *Asymptotic statistics*, volume 3. New York: Cambridge University Press.

van Gerven, M. A. J. (2017). A primer on encoding models in sensory neuroscience. *Journal of Mathematical Psychology*, 76:172–183.

Vazquez, A. L. and Noll, D. C. (1998). Nonlinear aspects of the BOLD response in functional MRI. *NeuroImage*, 7(2):108–118.

Walther, A., Nili, H., Ejaz, N., Alink, A., Kriegeskorte, N., and Diedrichsen, J. (2016). Reliability of dissimilarity measures for multi-voxel pattern analysis. *NeuroImage*, 137:188–200.

Walther, D. B., Caddigan, E., Fei-Fei, L., and Beck, D. M. (2009). Natural scene categories revealed in distributed patterns of activity in the human brain. *Journal of Neuroscience*, 29(34):10573–10581.

Wandell, B. A. and Winawer, J. (2015). Computational neuroimaging and population receptive fields. *Trends in Cognitive Sciences*, 19(6):349–357. Publisher: Elsevier.

Weichwald, S., Meyer, T., Özdenizci, O., Schölkopf, B., Ball, T., and Grosse-Wentrup, M. (2015). Causal interpretation rules for encoding and decoding models in neuroimaging. *NeuroImage*, 110:48–59.

Wixted, J. T. (2007). Dual-process theory and signal-detection theory of recognition memory. *Psychological Review*, 114(1):152–176.

Zemel, R. S., Dayan, P., and Pouget, A. (1998). Probabilistic interpretation of population codes. *Neural Computation*, 10(2):403–430.

Zucchini, W. (2000). An introduction to model selection. *Journal of Mathematical Psychology*, 44(1):41–61.