Bayesian Parametric Receptive-Field Identification from Sparse or Noisy Data

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Giacomo Bassetto

aus Castelfranco Veneto/Italien

Tübingen

2022

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	07.12.2022
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter/-in:	Prof. Dr. Jakob Macke
2. Berichterstatter/-in:	Prof. Dr. Philipp Berens

Abstract

Characterizing the stimulus selectivity of sensory neurons is an important step towards understanding how information about the world is represented in the brain. However, this is a computationally challenging task, in particular due to the probabilistic nature of the relationship between external stimuli and neural responses and the high dimensionality of the space of natural stimuli. State-of-the-art receptive field identification methods based on empirical Bayes scale poorly to high-dimensional settings, and computationally efficient implementations rely on stringent assumptions about the spike generation process. Furthermore, these models fail to provide principled credible intervals for experimentally relevant parameters making it hard to propagate uncertainty for hypothesis testing in regimes of sparse or noisy data.

Here, we present a full Bayesian approach to identify receptive fields in sparse data regimes, which provides also a principled quantification of the estimation uncertainty. We take advantage of the fact that, for many sensory areas, there are canonical models that explain how neurons encode their inputs into firing rates. These models usually rely on few, interpretable parameters and can be used to constrain the space of receptive fields that can explain the data. While such models may not be flexible enough to capture all nuances of a particular receptive field, they can be effective for obtaining a fast characterization of the encoding properties of a neuron.

We perform Bayesian inference directly on these model parameters and we show that we can detect the presence of a receptive field with a few tenths of measured spikes in physiological conditions. Furthermore, we investigate how different amounts of data constrain the model parameters and we illustrate how a full Bayesian approach can be used to test competing hypotheses and characterize a dataset of real, sparsely-sampled neurons. In this work, our focus is directed in modeling neurons in visual cortical areas, but our flexible approach has the potential to be generalized to neurons in other brain areas, with different input-output properties.

Acknowledgments

I would have not been able to accomplish the results presented in this thesis without the many people who supported me over these years. First and foremost I want to thank Prof. Macke for the opportunity to join his research group, for his mentorship and his patience with me. I am grateful for the many opportunities he gave me to participate to exciting conferences and meet other excellent scientists. I want to thank him also for his great achievement of having put together such an amazing group of brilliant people, making the working atmosphere always lively and friendly. My thanks also go to Prof. Kerr for hosting me in his research group during the last phases of our collaboration on the receptive-field mapping project. I particularly wish to thank Dr. Damian Wallace for his advice and our many scientific exchanges. This thesis would have also not been possible without Dr. Carl Holmgren and Dr. Takashi Handa, who collected and provided valuable electrophysiological data. I want to acknowledge my colleagues Pedro Goncalves, Marcel Nonnenmacher, Poornima Ramesh and Alexandre René: from scientific exchanges in the office to random and often lively discussions during our lunch breaks and many other occasions, the time spent with them was always great and never boring. A particular thanks to Marcel Nonnenmacher for being always honest and a good friend. I will also never be able to thank Pedro Goncalves enough for all his support and the positive angle he could always provide, something that kept me motivated throughout some difficult times. A great thanks also to Poornima Ramesh for all the work and the positive energy that made our joint publication possible and for being so often such a great example. A special thanks also to Philippe Fischer, Ivan Vishniakou, Andres Flores and Elhanan Ben Yishay, fellow Phd Students at the Max-Planck Institute for Neurobiology of Behavior and to our coordinator Ezgi Bulca for their feedback, their support and the many lively lunch breaks. I must thank also my amazing neighbors, Rebekka, Rima, Goran and Cai, with a special mention to Mu, who made me feel at home here in Bonn. I have to thank my family for their support and solid trust in me during all these year. Lastly, my gratitude goes wholeheartedly to my wife Mehwish for being such an important cornerstone in my life. She is the best and always so mindful and without her by my side this work would have not been possible.

Contents

1 Introduction **1**

2 Bayesian inference **9**
2.1 Bayes' theorem 9
2.2 Formal definition 10
2.2.1 Model comparison 10
2.2.2 Model averaging 11
2.3 Nested Sampling 12
2.3.1 Mathematical formulation of NS 12
2.3.2 Algorithm 13
2.4 Why Nested Sampling? 15
2.5 Summary 16

3 Receptive fields in V1 **19**
3.1 Neurons in the Primary visual cortex 19
3.2 Receptive-field models 21
3.2.1 Linear-Gaussian Models 22
3.2.2 Linear-Nonlinear (LN) cascades 23
3.2.3 Multi-filter LN cascades 23
3.3 Receptive field identification 25
3.3.1 Moment-based methods 26
3.3.2 Model-based estimators 28
3.4 Summary and discussion 31

4 A generative model for neurons in Primary Visual Cortex **35**
4.1 Generative model 35
4.1.1 Linear filter 36
4.1.2 Static Nonlinearity 40
4.2 Response properties 41
4.2.1 Complex-valued receptive field model 41
4.2.2 Orientation and frequency tuning 42
4.2.3 Direction selectivity 44
4.2.4 Phase dependence and phase invariance 46

4.3 Discretization of the model 49
4.4 Relation to other models 50
4.5 Summary and discussion 51

5 Nested Sampling for GLMs 53
5.1 Motivation 53
5.2 Collapsed Nested Sampling 54
5.3 Application to Generalized Linear Models 56
5.4 Performance analysis 57
5.4.1 Sampler Efficiency 58
5.4.2 Approximation Errors 59
5.4.3 Computational costs 62
5.5 Methods 64
5.5.1 Synthetic data 64
5.5.2 Bayesian inference 65
5.5.3 Estimating the quality of the posterior distribution . . 66
5.5.4 Implementation 68
5.6 Summary 68

6 Receptive field identification on synthetic data 71
6.1 Introduction 71
6.2 Receptive Field detection 71
6.3 Identification of model parameters 74
6.3.1 Characterizing the orientation 74
6.3.2 Quantification of estimation quality 75
6.4 Simulation details & inference 78
6.4.1 Generative model 78
6.4.2 Stimulus and simulation partitioning 78
6.4.3 Noise simulations 79
6.4.4 Sampler settings 79
6.4.5 Residual uncertainty 80
6.4.6 Estimation Error 80
6.5 Data classification 81
6.5.1 Control model 81
6.5.2 Classification based on Bayes factors 81
6.5.3 CVLL classifier 82
6.5.4 Classification performances 82
6.6 Summary and Discussion 83

7 Electrophysiological recordings 85
7.1 Detection and model identification 85
7.1.1 Properties of the dataset 85
7.1.2 Models used for the analysis 87
7.1.3 Receptive Field identification 87

7.1.4 Model identification 89
7.2 Single cell analysis 90
7.2.1 Model comparison 90
7.2.2 Orientation and direction selectivity 92
7.2.3 Role of the nonlinearity 93
7.3 Aggregate results 94
7.3.1 Orientation and direction selectivity 94
7.3.2 Non-linear response properties 95
7.4 Material and Methods 98
7.5 Summary and Discussion 99

8 Conclusions **101**

A Generalized Linear Models **105**
A.1 Learning the model parameters 106
A.2 Generalized Quadratic Models (GQM) 107
A.3 Basis functions 108

B Frequency response **111**
B.1 Spatial filter 111
B.2 Temporal filter 112
B.3 Direction selectivity 113
B.3.1 Power of the response to a moving grating 113
B.3.2 Assymetry of the response 113

C Extensions **115**
C.1 Motion opponency 115
C.2 Non-zero DC gain 116

D SNR for GLMs **119**
D.1 SNR in a linear system with additive Gaussian noise 119
D.2 SNR for a GLM 120
D.3 Partitioning the SNR 121
D.4 Application to the V1 generative model 121

E Electrophysiological recordings **123**

F Supplementary figures and tables **125**

List of Figures

1.1 Illustration of neuroscience experiment 2
1.2 Limitations of a non-parametric approach 5
1.3 RFs sampled from the posterior distribution 6

2.1 Nested Sampling 12

3.1 LN cascade models 22
3.2 Multi-filter LNP 24
3.3 Spike-Triggered Average. 27

4.1 Separable and non-separable filters 37
4.2 Spatial and temporal filters 38
4.3 Stimulus sensitivity 40
4.4 Orientation selectivity 43
4.5 Direction selectivity 45
4.6 Phase selectivity 46
4.7 Phase invariance 48

5.1 Sampling efficiency 58
5.2 Errors: Model evidence 60
5.3 Errors: Posterior distribution 61
5.4 Computational and memory requirements 63

6.1 Simulated data 72
6.2 Detection performance 74
6.3 Classification results 75
6.4 Parameters identification 76

7.1 Description of the data 86
7.2 Receptive field detection 88
7.3 Model comparison 91
7.4 Orientation and direction selectivity 94
7.5 Non-linear response properties 96

B.1 Selectivity the spatial filters 112

F.1 Sampling efficiency on a toy model 126
F.2 Supplement 1 to Fig. 5.3 127
F.3 Supplement 2 to Fig. 5.3 128
F.4 Posterior marginals: false detection 129
F.5 Orientation-frequency joint marginal 130
F.6 Per-parameter RU vs observed spike count 131
F.7 Per-parameter RMSE vs observed spike count 132
F.8 Per-parameter RMSE vs RU 133
F.9 Marginals (1 minute) 134
F.10 Marginals (4 minutes) 135
F.11 Properties of the quadratic term 136

List of Tables

2.1 Scale for Bayes factors 11

4.1 Receptive field parameters 52

6.1 Performance of the Bayesian classifier 73
6.2 RMSE vs RU and spike counts 77
6.3 Variance vs bias 78
6.4 Ground truth and priors 79
6.5 Details on data partitioning 79

7.1 Distribution of spike counts 86
7.2 Model configurations 87
7.3 Average posterior model probabilities 90

A.1 GLM examples 106

F.1 Wilcoxon test 125
F.2 Student's t test 125
F.3 Details on RF detection 126

Chapter 1

Introduction

In order for us and other animals to be able to interact with our environments, the brain must encode, store and process information about its surroundings. In this thesis, we study and apply computationally and data-efficient methods to characterize the encoding properties of neurons in the early stages of the visual processing system.

The neural coding problem

In a typical experiment, a subject is presented with a rich repertoire of visual stimuli while at the same time neural activity from visual areas of the brain is acquired by different means which may consist of electrophysiological recordings or imaging data (see Fig. 1.1A). Neuronal activity consists of trains of stereotypical events called action potentials, or spikes. We can observe that a particular neuron responds particularly vigorously to some features of the visual stimulus, while remaining relatively silent in their absence (see Fig. 1.1, panels B and C). We call the entire set of features that drive the activity of that specific neuron its *receptive field* (RF). We can think of the receptive field as the piece of reality encoded in the activity of each specific neuron. Defined more rigorously, the receptive field is a relatively small dimensional subspace of the entire high-dimensional stimulus in which a neuron computes its response and it is a fundamental component of any model of neural encoding. Since neural activity is intrinsically variable even to repeated presentations of the same stimulus [11, 52, 94, 109], the neural coding problem has to be addressed statistically or probabilistically. In mathematical terms, if we denote a stimulus with s and the associated evoked response with r, solving the neural coding problem implies a full characterization of the conditional probability

$$p(r|s). \tag{1.1}$$

Due to the high-dimensional nature of visual stimuli, as well as to the variability of neural responses, receptive field characterization usually requires

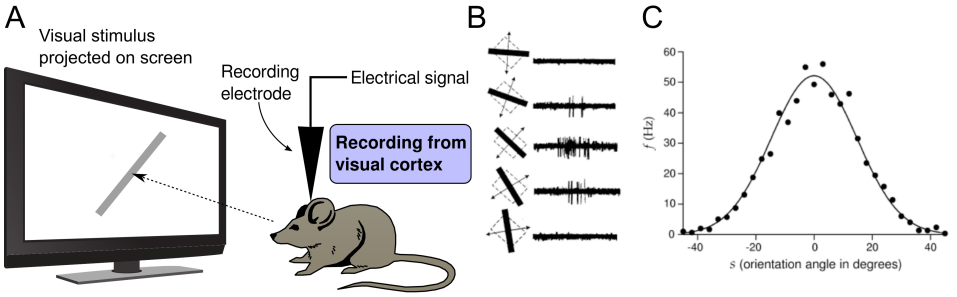

Fig. 1.1: **A typical neuroscience experiment. A)** A repertoire of visual stimuli is presented to a test subject, while electrical activity from one or more neurons is recorded from the visual areas of the brain. Here, the stimulus consists o **B)** Bars with distinct orientations and moving in different directions are presented on the screen, which elicit a neural response that is picked up by the electrode. The high-frequency events observable in some of the traces are known as *spikes*. **C)** Average number of elicited spikes per second (from here on referred to as the *firing frequency* of the neuron) as a function of the orientation of the bar, aligned to the preferred orientation (i.e. the value resulting in the highest observed spike count). The response intensity rapidly falls off when the stimulus orientation departs from the preferred one. This analysis reveals that the orientation of the bar is encoded in the firing frequency of this unit. [2]

long recording sessions to mitigate the effect of neural noise and explore the stimulus space sufficiently. This requirement may not always be compatible with other experimental constraints (e.g. when receptive-field identification is only one of several stages of an extensive experimental protocol) and, to complicate matters, neural selectivity may change on the timescale of an experiment due to adaptation or learning [67]. Similarly, experimental preparations involving awake or behaving animals also place tight technical constraints on the total length of an experiment for reasons concerning the stability of the recording or the minimization of the animal's distress. Consequently, receptive fields may have to be estimated from short segments of data containing only a small number of spikes.

Current approaches and their limitations

A popular approach to the neural coding problem is to identify a low-dimensional linear projection of the stimulus space that preserves the aspects of the stimulus that affect a neuron's probability of spiking. Several dimensionality reduction methods have been developed to address the prob-

[2] Panels B and C from Fig. 1.5 in Dayan and Abbott, "Theoretical Neuroscience" (2005), MIT Press [20].

lem of receptive field characterization. These methods can be categorized into three groups: moment-based [15, 98], model-based [76, 83, 84], and information theoretical [85, 99, 100], although these last two classes are often equivalent in several concrete cases [116]. Since all these methods make use of at least one parameter for each stimulus dimension, we commonly refer to them using the umbrella term "*non-parametric*" models.[3] The basic idea is that a neuron computes its response in a low dimensional subspace, spanned by a small number of stimulus features. Each direction defining a particular subspace can be thought of as a spatiotemporal subfield of the entire receptive field of the neuron. The identification of this subspace, which we denote with K, therefore overlaps with the characterization of the receptive field. We can express the probabilistic relationship between neural response, stimulus and receptive field by means of the conditional probability distribution

$$p(r|s, K). \tag{1.2}$$

For model-based methods, the subspace K can be identified e.g. by maximizing its log-likelihood function $\mathcal{L}(K; r, s) = \log p(r|s, K)$. A common feature to many of these models is their very high number of parameters, which makes them particularly "data-hungry". For example, using a stimulus consisting of a movie with a frame resolution of 100×100 pixels display at 50 Hz, and assuming that a neuron computes its response integrating 500 ms of visual stimulation, the parameter space is 250k-dimensional. Model/likelihood-base estimators therefore require thousands of spikes to converge (see Fig. 1.2C). Given the low firing rates sometimes encountered in in vivo preparations, this translates potentially into hours of data. To address this issue, clever regularization strategies have been developed that take into account known properties of visual receptive fields, like their smoothness [96], spatial and frequency locality [79] or low-dimensional structure [80, 85]. Most of these regularizers can be interpreted in a Bayesian setting as placing a prior distribution over the model parameters encoding our prior knowledge and expectations about the expected properties of a visual receptive field. Priors, in turn, belong also to some parametric family, whose parameters (here called *hyper-parameters*) are usually optimized using a technique know as type II maximum likelihood or empirical Bayes [96]. These algorithms in general scale poorly to high-dimensional settings. State-of-the-art efficient implementations often rely on stringent assumptions, e.g. by restricting $p(r|s, K)$ to be a normal distribution or by limiting the analysis to linear models [5, 79, 96]. Whether or not these constraints are appropriate in a specific context must be assessed on a problem-by-problem

[3] In this context, non-parametric denotes a lack of semantically meaningful parameters, rather than an actual lack of parameters. We maintain that probably the term *unstructured model* would have been a better choice, but non-parametric is already widespread in the literature.

basis,[4]. We must be aware that, should the data not meet the assumptions, the resulting estimates are distorted [68].

Setting aside for the moment any concern regarding computational efficiency, we identify at least two more potential problems associated with non-parametric approaches. The first one is the lack of interpretability of the estimates: since model parameters do not carry any particular meaning other than the intensity of the receptive field in a specific point in space and time (see Fig. 1.2D), quantifying high-level properties of a receptive field, e.g. its frequency or orientation selectivity, requires an additional analysis step in which a parametric model of the receptive field shape is fitted to the non-parametric estimate [89, 93]. The second limitation regards the propagation of estimation uncertainty for hypothesis testing: when the non-parametric RF estimate is not well constrained by the data, there is a need of a method to propagate this uncertainty down the analysis pipeline to derive credible intervals for the inferred RF properties [17]. This problem is particularly aggravated in cases where the data is so scarce that the marginal likelihood of the data is not tightly concentrated around its maximum. In this case, point estimate computed by empirical Bayes used to set the hyper-parameters does not provide an exhaustive representation of the RF structure supported by the data, with the effect of heavy underestimating of the actual amount of posterior uncertainty on the values of the model parameters.

Bayesian Inference on parametric receptive field models

To tackle these issues, we propose to include the receptive field properties of interest directly in the generative model of neural responses. We want also to leverage formerly acquired knowledge to model neural responses to visual stimuli using a small number of semantically meaningful and experimentally relevant parameters. For example, the role of most cells in the early visual areas can be interpreted as edge detectors whose receptive fields can be well approximated by a Gabor wavelet [49]. This model is encoded as the conditional probability distribution $p(r|\boldsymbol{\theta}, s)$ describing the data-generating process, where $\boldsymbol{\theta}$ is a small vector of parameters describing the properties of the receptive field. We will also regularize our estimates by strongly encouraging parameter values that result in *a priori* expected receptive field shape. Concretely, these regularizers will be implemented by means of a prior distribution $p(\boldsymbol{\theta})$ and we will operate within a Bayesian framework, where we will explicitly target the posterior distribution of the model parameters given the data:

$$p(\boldsymbol{\theta}|r, s) = \frac{p(r|\boldsymbol{\theta}, s)p(\boldsymbol{\theta})}{p(r|s)}. \tag{1.3}$$

[4]For example, linearity is definitively not a good assumption when modeling the response of non linear units like visual cortical complex cell.

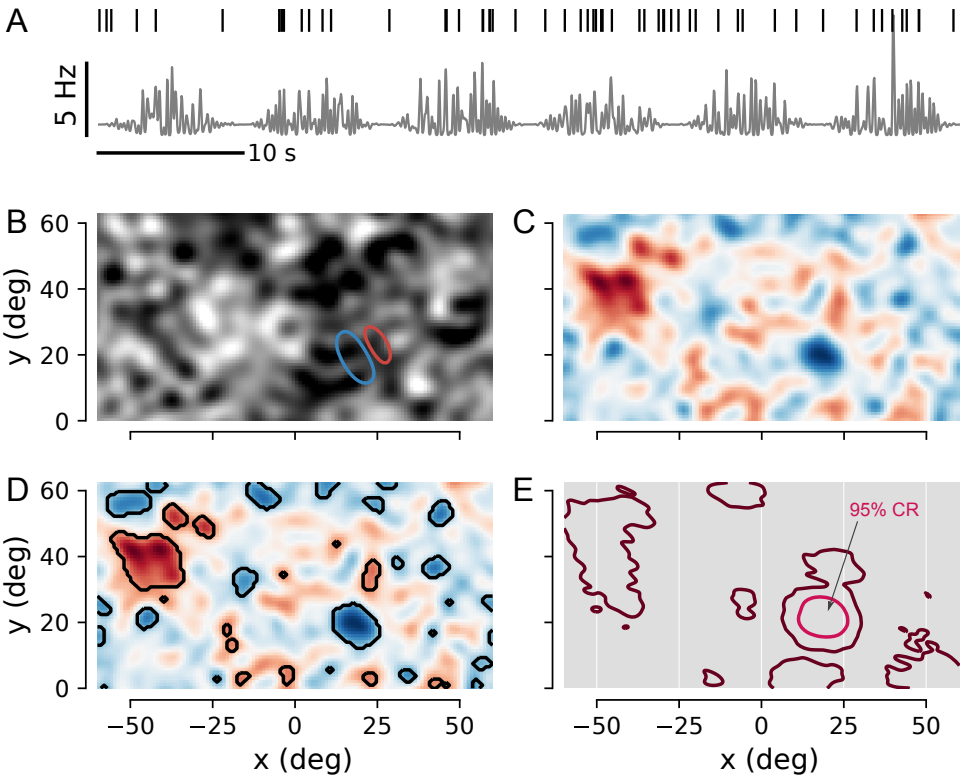

Fig. 1.2: **Limitations of a non-parametric approach. A)** 1 minute of simulated activity. The instantaneous firing rate of the neuron is shown as a gray solid line, while sampled spikes are represented as black ticks above the trace (here, 57 spikes). **B)** A random stimulus frame covering a field of view of approximately 120×60 degrees and the spatial receptive field of the simulated unit (for details about the generative model, see Chapter 4). The stimulus consisting of spatially and temporally correlated Gaussian noise, contrast-modulated at 0.1 Hz. The result is the modulation of the firing rate at the same frequency observable in A. **C)** The spike-triggered average (STA), computed from the spikes in (A) and the corresponding stimuli (for the definition of the STA, see Chapter 3.3). The STA shows what was on average presented on the screen when the unit fired a spike. Due to the presence of spatial correlations in the stimulus, the STA is a biased estimator of the RF. Red indicates excitatory regions of the RF, while inhibitory regions are represented in blue. **D)** Regions of the STA within which values significantly deviate from zero ($p > 99\%$ computed using bootstrap), indicating non-random correlations between the spike train and the stimulus within these regions. However, due to the scarce amount of data used to compute the STA, we cannot tell which regions popped out by chance and which ones instead genuinely suggest the presence of a receptive field. **E)** 95% (pink line) and 99% (maroon line) credible regions for the receptive field position parameters, derived from the posterior distribution given the spikes in (A). This result suggests strong evidence for the presence of a receptive field within the region encircled by the pink solid line.

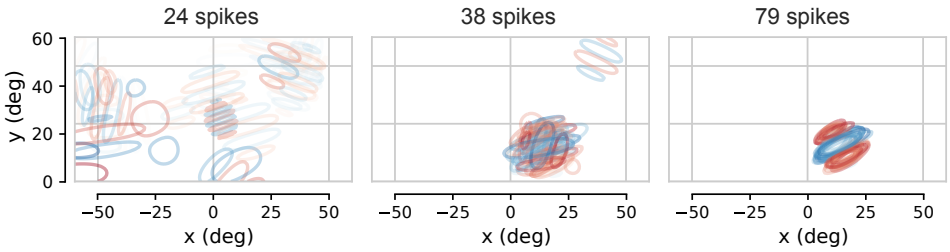

Fig. 1.3: **Receptive fields sampled from the posterior distribution.** Receptive field shapes sampled from the posterior distribution after observing 24 (left), 38 (center) and 79 (right) spikes, corresponding, respectively to 1, 2 and 4 minutes of data. Notice how samples become less scattered as more data is used for inference.

The benefits of this approach are two-fold: first, a low number of parameters allows to estimate receptive fields from a small number of measurements [44]; second, Bayesian inference provides a principled way of deriving credible intervals from the posterior distribution of the model parameters (see Fig. 1.2E). Estimation uncertainty is completely encoded in the posterior distribution, and new evidence is automatically accounted for as it becomes available (see Fig. 1.3). Finally, the marginal likelihood of the data $p(r|s)$, known also as *model evidence*, can be used to compare different competing hypotheses about the spike generation process in a principled, Bayesian way. We will discuss this idea in the next chapter and we will see a concrete application in Chapter 7.

Outline

The rest of this thesis is organized as follows. In the next chapter we will refresh the notions of Bayesian inference and Bayesian model comparison, and we will introduce nested sampling. In Chapter 3 we will revise the properties of neurons in primary visual cortex, the statistical models developed to describe them, and we will have a closer look at the commonly used techniques to identify their receptive fields, which we have already partially introduced here. In Chapter 4 we will present a RF model governed by a small number of parameters, but still flexible enough to capture the salient features of early visual cortical neurons. Chapter 5 will introduce a strategy to perform Bayesian inference efficiently on this type of models. In Chapter 6 we will investigate the convergence properties of our Bayesian approach to receptive field characterization; we will do so by assessing the RF detection and identification performance of our algorithm on simulations spanning a wide range of physiologically realistic firing-rates and noise levels. In Chapter 7 we illustrate the potential of a fully Bayesian approach to the analysis of a large dataset of electrophysiological recordings from rat primary visual cor-

tex; we will show that even small amounts of data can be leveraged to draw meaningful and interesting conclusions. Chapter 8 will provide an overview of the results presented and discussed in previous chapters, and will offer an outlook on possible future research lines.

Chapter 2

Bayesian inference

Bayesian inference (BI) is a method of statistical inference in which Bayes' theorem is used to update the probability for a hypothesis as more evidence or information becomes available. In this chapter we will go briefly through some basic concepts behind a Bayesian inference framework, such as parameter inference Bayesian model comparison (BMC). We will also introduce Nested Sampling, an alternative to Markov-Chain Monte Carlo (MCMC) sampling methods for Bayesian inference combining two inference steps in one single algorithm, and we will explain its advantages over MCMC methods.

2.1 Bayes' theorem

Bayes' theorem provides a principled way to update one's belief regarding some hypothesis X after having observed the outcome of some experiment Y. Any prior knowledge or belief about X is encoded in a *prior* probability distribution $p(X)$. The dependency between X and Y is encoded by a conditional probability $p(Y|X)$. This can be interpreted also as the *likelihood* of X for a fixed Y, because $p(Y|X) = L(X;Y)$. Bayes' theorem states that the updated belief on the value of X is given by the *posterior* probability

$$p(X|Y) = \frac{p(Y|X) \cdot p(X)}{p(Y)}. \tag{2.1}$$

Here, $p(Y) = \int p(Y|X)p(X)\mathrm{d}X$ is sometimes termed the *marginal likelihood*. It is the normalizing factor for the posterior distribution and it plays a crucial role in the context of Bayesian model comparison, discussed in the next section.

2.2 Formal definition

Three levels of inference can be identify in a Bayesian inference framework, which we are going to address in order.

Parameter inference

The first level is parameter inference: here we have a specific model in mind, which we denoted by $\mathcal{M}$, parameterized by a vector of parameters $\boldsymbol{\theta}$, to explain the data generation process $p(\mathcal{D}|\boldsymbol{\theta}, \mathcal{M})$. At this level, we are concerned with identifying the favorite values of the parameters which are consistent with both the observed data $\mathcal{D}$ and some prior knowledge or expectation. This updated belief is encoded in the posterior distribution

$$p(\boldsymbol{\theta}|\mathcal{D}, \mathcal{M}) = \frac{p(\mathcal{D}|\boldsymbol{\theta}, \mathcal{M})p(\boldsymbol{\theta}|\mathcal{M})}{p(\mathcal{D}|\mathcal{M})} \tag{2.2}$$

where $p(\boldsymbol{\theta}|\mathcal{M})$ is our *a priori* belief before we observed any data. Here, the marginal likelihood $p(\mathcal{D}|\mathcal{M})$ is often also termed *model evidence* or *Bayesian evidence* or *model likelihood*, since it quantifies the probability that the data was generated by the model in question after taking into account all the uncertainty about the model parameters. It is obtained by marginalizing the model parameters:

$$p(\mathcal{D}|\mathcal{M}) = \int p(\mathcal{D}|\boldsymbol{\theta}, \mathcal{M})p(\boldsymbol{\theta}|\mathcal{M})\mathrm{d}\boldsymbol{\theta} \tag{2.3}$$

Evaluating the model evidence is arguably computationally demanding and it is a crucial component for Bayesian model comparison and averaging, which we are going to discuss next.

2.2.1 Model comparison

The second level is model comparison. At this stage, we realize that there are several possible competing models $\mathcal{M}_0, \mathcal{M}_1, \ldots$ to explain our data, and we want to know what the relative plausibility of each of them is in light of the data. To this end, we compute the posterior odds

$$\frac{p(\mathcal{M}_1|\mathcal{D})}{p(\mathcal{M}_0|\mathcal{D})}.$$

Each model's posterior is delivered by Bayes' theorem:

$$p(\mathcal{M}_i|\mathcal{D}) = \frac{p(\mathcal{D}|\mathcal{M}_i)p(\mathcal{M}_i)}{p(\mathcal{D})}. \tag{2.4}$$

We do not need necessarily to derive the value of $p(\mathcal{D})$, since this term cancels out when computing the posterior odds:

$$\frac{p(\mathcal{M}_1|\mathcal{D})}{p(\mathcal{M}_0|\mathcal{D})} = \frac{p(\mathcal{D}|\mathcal{M}_1)}{p(\mathcal{D}|\mathcal{M}_0)} \cdot \frac{p(\mathcal{M}_1)}{p(\mathcal{M}_0)}, \tag{2.5}$$

$\lvert \ln B \rvert$	relative odds	best model probability	interpretation
< 1.0	$< 3:1$	< 0.75	not worth mentioning
< 3.0	$< 20:1$	< 0.95	Substantial
< 4.6	$< 100:1$	< 0.99	Strong
> 4.6	$> 100:1$	> 0.99	Decisive

Table 2.1: **BF reference table.** A slightly modified Kass and Raftery's scale to assess the strength of evidence [53].

from which we can learn that the posterior odds equal the *Bayes factor* [50] times the prior odds. The Bayes factor (BF) is the leftmost term on the r.h.s. of eq. (2.5) and can be interpreted as a Bayesian version of the likelihood ratio of two competing hypotheses in classical statistical terms [33]. The BF between $\mathcal{M}_1$ and $\mathcal{M}_0$ is often denoted as B_{10}:

$$B_{01} = \frac{p(\mathcal{D}|\mathcal{M}_0)}{p(\mathcal{D}|\mathcal{M}_1)}.$$

Table 2.1 gives an indication of the scale for the strength of the evidence in favor of either model as a function of the Bayes factor. Bayes factors provide a Bayesian alternative to classical hypothesis testing [34, 35] and in Bayesian statistics they play a similar role to what p-values do in classical statistics [62]. Unlike p-values, however, Bayes factors allow one to compute evidence in favor of a null hypothesis and can be used to compare models that cannot be nested. An other advantage of Bayes factor is that they automatically implement a penalty for including too much model structure, therefore guarding against over-fitting [53].

In *Bayesian model selection* (BMS), the model with the highest posterior probability among a set of competing models is picked to represent the data:

$$\hat{\mathcal{M}}_{BMS} = \arg\max_{\mathcal{M}_i} p(\mathcal{M}_i|\mathcal{D}). \tag{2.6}$$

2.2.2 Model averaging

The third level of inference is represented by *Bayesian Model Averaging* (BMA) [43, 82]. When none of the proposed models is clearly the best in terms of explaining the data, we incorporate the account for model uncertainty within a posterior predictive distribution of the values of the parameters given the data:

$$p(\boldsymbol{\theta}|\mathcal{D}) = \sum_i p(\boldsymbol{\theta}|\mathcal{D}, \mathcal{M}_i) \cdot p(\mathcal{M}_i|\mathcal{D}). \tag{2.7}$$

This strategy is useful when several competing models relying on the same parameters but different mechanics compete as an explanation of the data.

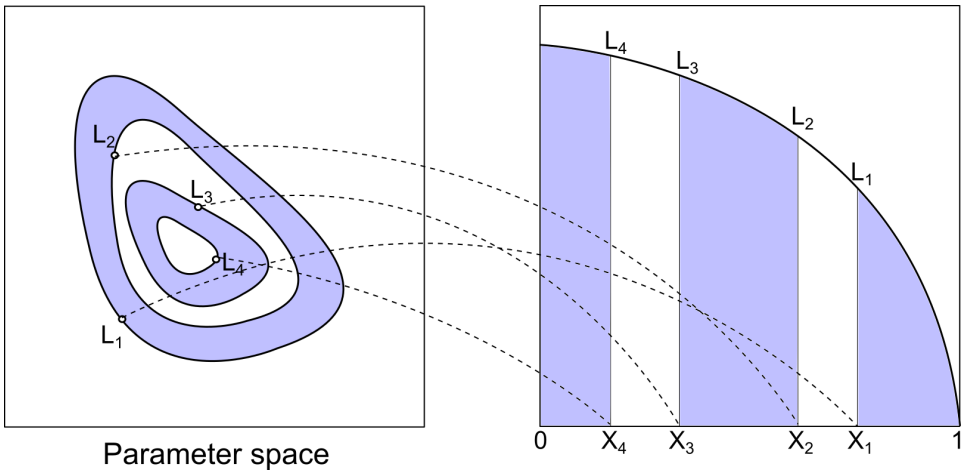

Fig. 2.1: **Nested Sampling.** For a detailed description of the algorithm, see the main text.

2.3 Nested Sampling

As already mentioned, evaluating the model evidence is a computationally demanding task, because it consists in solving a multi-dimensional integral, averaging the likelihood over a possibly much wider prior. *Nested sampling* (NS) [103] is a clever algorithm to compute the model evidence: it transforms the D-dimensional integral into a 1-dimensional integration problem that can easily be solved. As a byproduct, NS also produces posterior samples: it has the unique advantage over MCMC algorithms to tackle model likelihood and parameter inference simultaneously. It can readily be used for most inference problems, within some efficiency limitations that are implementation-dependent.

2.3.1 Mathematical formulation of NS

In this and the following subsections, we will omit the explicit dependency on the model and will use the symbol $\mathcal{L}(\boldsymbol{\theta})$ to denote $p(\mathcal{D}|\boldsymbol{\theta}, \mathcal{M})$, in order to emphasize that we treat the likelihood as a function of the model parameters. Rewriting (2.2) accounting for this new notation,

$$p(\mathcal{D}) = \int \mathcal{L}(\boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta},$$

we emphasize how the evidence is the expected value of the likelihood under the prior. In order to solve this high-dimensional integral, we start by defining $X(\lambda)$ as the prior mass associated with likelihood values above λ:

$$X(\lambda) \triangleq Pr(\mathcal{L}(\boldsymbol{\theta}) > \lambda) = \int_{\mathcal{L}(\boldsymbol{\theta})>\lambda} \pi(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}. \tag{2.8}$$

This is also sometimes termed the *likelihood-restricted prior* (LRP) and the entire nested sampling scheme is built upon this concept. The LRP is a decreasing function of λ, with $X(0) = 1$ and $X(\mathcal{L}_{\max}) = 0$, where $\mathcal{L}_{\max}$ is the maximum possible value of the likelihood. We also additionally define $\mathrm{d}X$ as the prior mass associated with likelihoods $[\lambda, \lambda+\mathrm{d}\lambda]$. An infinitesimal interval $\mathrm{d}X$ contributes an amount $\lambda \mathrm{d}X$ to the evidence, therefore:

$$p(\mathcal{D}) = \int \mathcal{L}(\boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} = \int_0^1 L(X) \mathrm{d}X, \tag{2.9}$$

where $L(X)$ is the inverse of $X(\lambda)$, defined as $L(X) \triangleq \arg\inf_\lambda X(\lambda) \geq X$. Inverting eq. (2.8) has the effect "sorting" the prior volume X according to increasing values of the likelihood function. Assuming that we can evaluate $L_j = L(X_j)$ for a sequence

$$0 < X_M < \cdots < X_2 < X_1 < 1,$$

then the evidence (2.9) can be numerically approximated with arbitrary precision as

$$p(\mathcal{D}) \approx \hat{\mathcal{Z}}_{NS} = \sum_{j=1}^{M} w_j L_j, \tag{2.10}$$

for a suitable set of weights w_j proportional to the change in volume δV_j, e.g. computed using the trapezium rule

$$w_j = \frac{1}{2}(X_{j-1} - X_{j+1}).$$

Since $L(X)$ is a positive, monotonically decreasing function, this sum is well behaved and a lower and an upper bound exist [104].

2.3.2 Algorithm

In general, except for the most trivial distributions, $L(X)$ is unknown. NS overcomes this issue by picking the likelihood levels L_i in a way that the expected value of each X_i is known. Suppose we begin with an initial set of N i.i.d. samples $\boldsymbol{\theta}_i$ drawn from the prior, labeled $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N$: by definition, this set of points is associated to the prior mass $X_0 = 1$ with corresponding likelihood $L_0 = 0$. Without any loss of generality, we also assume that these points are sorted according to their likelihood value, i.e.

$$L_i = \mathcal{L}(\boldsymbol{\theta}_i) < L_j = \mathcal{L}(\boldsymbol{\theta}_j), \text{ for } 0 < i < j \leq N. \tag{2.11}$$

We now remove $\boldsymbol{\theta}_1$, the point with the lowest likelihood, and replace it with another draw from the prior, $\boldsymbol{\theta}_{N+1}$, under the constraint $\mathcal{L}(\boldsymbol{\theta}_{N+1}) > L_1$. According to (2.8), this new set of points occupies a fraction of the prior

mass corresponding to $X(L_1)$, therefore we can associate the prior volume $X_1 \leftarrow X(L_1)$ to θ_1. We relabel the new sequence ensuring all points are sorted according to their likelihood; we now have a set of N points $\boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_{N+1}$. Now $\boldsymbol{\theta}_2$ is the point with the lowest likelihood, L_2, so it must represent the slice $X_2 \leftarrow X(L_2)$. We remove it and we keep iterating the procedure illustrated in this paragraph, accumulating evidence at each iteration following (2.10), until we collect M points and the sum has converged.

The amount prior volume shrinks at each iteration following a Beta distribution [104]:

$$X_i/X_{i-1} \sim \text{Beta(N, 1)}, \tag{2.12}$$

corresponding to an expected shrinkage factor of $e^{-\frac{1}{N}}$. This fact can be exploited to replace $X(L_i)$ with its expected value $\mathbb{E}[X_i] = e^{-\frac{i}{N}}$ when computing the weights in (2.10). The resulting estimator $\hat{\mathcal{Z}}$ is consistent. Alternatively, eq. (2.12) can be used to generate samples from X_i, which can be used to estimate a credible interval for $p(\mathcal{D})$.

Samples from the posterior $p(\boldsymbol{\theta}|\mathcal{D})$ can be extracted as a free by-product of the integration routine by taking the sequence of sampled points $\boldsymbol{\theta}_i$ and weighting each sample j by $p_j = w_j L_j / p(\mathcal{D})$.

Sampling step

The hardest part of nested sampling is to sample uniformly from the prior subject to the hard constraint that the likelihood needs to be above a certain level. This is an active field of research and many specific implementations of this step have been proposed. These include (MCMC-based) local step algorithms, exploiting the live point knowledge, sampling by proximity, and sampling by direction. An excellent review of all these methods is provided in [10]. Some specific implementations worth mentioning because they have been used for some analysis or will be discussed in this thesis are Metropolis nested sampling [104], ellipsoidal sampling with X-means [26], rejection sampling (MultiNest) [27], and slice-sampling (PolyChord) [36].

Termination

The running sum (2.10) will eventually converge for all well behaving likelihood functions, i.e. if $\mathcal{L}(\boldsymbol{\theta}) < \infty$ for all $\boldsymbol{\theta}$): since the amount of prior volume occupied by the pool of live points shrinks exponentially with rate $1/N$, the contribution of the likelihood term L_i to the estimate of the evidence becomes vanishingly small for $i \to \infty$. NS can be terminated when the live point with the highest likelihood $L_{\max}$ would not contribute significantly to $\mathcal{Z}$ if it were removed in the next iteration. This is the approach adopted by MultiNest [27], which terminates if $L_{\max} X_i < \epsilon \cdot p(\mathcal{D})$, where ϵ is a user-defined tolerance. The number of iterations required for convergence scales linearly with the number of live points [105].

2.4 Why Nested Sampling?

Nested sampling was developed to estimate high dimensional integrals often encounter in a Bayesian inference framework when computing e.g. a model's evidence. Very conveniently, NS generates a set of samples from the posterior probability as a byproduct of computing the evidence, solving two problems (parameter inference and computing the evidence) simultaneously. Its convergence properties are a well understood and studied problem and its intrinsic approximation errors are well characterized and have been explicitly modeled [16, 42, 104]. Nested sampling is routinely used as a fool-proof inference black-box in a variety of fields [6].

Multimodal posteriors

One advantage of NS over MCMC algorithms is that NS can deal effortlessly with multi-modal target distributions [26]. When performing Bayesian inference in a scarce-data scenario, multi-modal posterior distributions represent a likely scenario because many diverse explanations may be possible to explain the data. Furthermore, to add additional confusion to an already challenging task, there might be unknown singularities in a model's parameterization causing different parameter sets resulting in the same model behavior: in such cases, the posterior will always be a multi-modal distribution. Although strategies to deal with this problem exist (e.g. simulated annealing [64]), their use requires a considerable amount of fine tuning. When evaluating the target probability distribution is a computationally demanding task, extensive fine tuning sessions for each instance of the problem may be prohibitive. On the other hand, several NS implementations extensions exist to address multi-modal targets, with many of them requiring a minimal amount or not tuning at all [10, 26, 36, 107].

Computational advantages

State-of-the-art MCMC algorithms (e.g. HMC) [64] often require the gradients of the target log-probability density with respect to the model parameters to explore the parameter space efficiently. Computing these gradients may be a computationally expensive operation. Although Hamiltonian trajectories can (in principle) be simulated using approximated gradients computed on mini-batches of data (provided that the acceptance criteria are computed over the whole dataset), this strategy presents a major challenge: smaller batches trade computational time against the efficiency of the sampler, as rougher approximations of the gradients lead to inexact Hamiltonian trajectories, which in turn result in higher rejection rates. Finding the right batch size requires a fair amount of fine tuning. On the other hand, NS relies solely on log-likelihood evaluations and does not need gradients at all.

It is important to mention that, like any other Monte Carlo algorithm, nested sampling also cannot guarantee that all modes of the target distribution are identified. However, the resolution of the algorithm is controlled by the number of live points, which is a free parameter, with higher values of N_{live} resulting in better resolution. This effectively reduces the chances of missing a narrow mode, as more and more live points are used to explore the likelihood landscape. On the other hand, since NS computational cost scales as $O(N_{\text{live}})$, using more live points produces more precise estimates at the cost of longer run times. A compromise between computational costs and accuracy must therefore be assessed on a problem-by-problem basis.

Conclusions

Nested Sampling relies on a minimal number of free parameters which for most do not require tuning (e.g. of a proposal distribution as in conventional MCMC). The number of live points, N_{live}, is the only free parameter of the original NS algorithm, but other implementations may rely on a few extra parameters. MultiNest, for example, requires one additional tolerance parameter for the stopping criterion. Furthermore, it is possible to combine several independent NS runs with a small number of live points into an equivalent extended run with a larger number of live points [104], allowing a user to dynamically increase the resolution of the algorithm if they realize that their initial guess for the N_{live} parameter was not good enough. Since one of the motivations of this work is to provide a flexible and reliable Bayesian inference tool to a community of non-expert users, dealing with a small number of easily interpretable parameters, which require a minimum amount or no tuning at all, is a very desirable property.

We should also emphasize that NS combines parameters and model inference within one single algorithm by simultaneously sampling from the posterior on its way to compute the model evidence. In our opinion, this is one more reason that makes it a powerful alternative to MCMC algorithms, since, for the latter, model evidence must be estimated in some additional post-processing routine after collecting samples from the posterior distribution.

2.5 Summary

In this chapter we formally introduced the most important concepts related to a Bayesian inference framework. After introducing Bayes' theorem, we discussed the three possible levels of Bayesian inference: parameter inference, model inference, and model averaging. We will see concrete applications of all the three types of inference throughout Chapter 6 and 7, where we will discuss receptive field identification and will compare different alternative hypotheses to explain the data. We then introduced nested sampling

and explained we think why it is a valid tool for Bayesian inference, especially for model comparison and averaging. We also mentioned that choosing the right value for the N_{live} parameters is crucial to achieve a good compromise between computational performance and accuracy of the estimates. In Chapter 5 we will present a novel strategy to reduce the number of live points without sacrificing accuracy.

Chapter 3

Modeling and identification of receptive fields in the primary visual cortex

In this chapter, we will discuss the physiological properties of neurons in the primary visual cortex (V1). This will lay the basis to understand the computational model presented in the next chapter. We will also introduce the probabilistic framework that has been developed to model neural activity in V1 and discuss modern inference procedures and their limitations. These considerations provide the motivations for our different approach to modeling neural activity and will serve as a context for the work discussed in this thesis.

3.1 Neurons in the Primary visual cortex

Neurons are specialized cells found in the nervous system. They are electrically excitable cells which form a network and communicate with each other via specialized connections called synapses. The signaling process is partly electrical and partly chemical. Neurons maintain voltage gradients across their membranes. If the voltage abruptly changes over a short interval, the neuron generates an *all-or-nothing*,[1]stereotypical electrical pulse called *action potential*, or *spike*. This potential then rapidly activates synaptic terminals and propagates activity to other neurons. Neural coding is concerned with how sensory and other information are represented in the brain by neurons – in our specific case, in the early stages of the visual (sensory) system. The main goal of studying neural coding is to characterize the relationship between the stimulus and the individual or ensemble neuronal

[1]When we speak about all-or-none responses we mean that, if a neuron responds at all, then it must respond completely. Greater intensity of stimulation does not produce a stronger signal, but can increase firing frequency [51].

responses, and the relationships among the electrical activities of the neurons within the ensemble [9]. The features of the sensory stimulus that drive the response of a specific neuron is known in the literature as that neuron's *receptive field* (RF) [45]. Here, we focus on models of the RF of single visual cortical neurons and how to efficiently identify their parameters.

Primary visual cortex (V1) is the first cortical area dedicated to visual processing. The physiological properties of this area have been intensively studied since the seminal work by Hubel and Wiesel [46], who also introduced the standard classification of neurons in V1 into two main groups: *simple* and *complex cells*. The original classification is based on four characteristic properties of simple cells: 1) the presence of distinct excitatory and inhibitory subregions; 2) spatial summation within each subregion, that is, responses become stronger as stimuli fill more space within a subregion; 3) excitatory and inhibitory subregions are antagonists, i.e. their contributions cancel out; 4) the response to any stimulus can be predicted from the receptive field map. Cells that do not show these characteristics are classified as complex.

Simple cells respond selectively to bars and gratings presented at a specific position, orientation, spatial frequency, and contrast polarity [46, 97]; complex cells also respond to bars or gratings of adequate orientation and spatial frequency, but are insensitive to small translations of the stimulus within the receptive field and respond equally well regardless of the contrast polarity of the stimulus [46, 56]. This constancy in the response to variations of the stimuli is commonly called *phase invariance* [12, 69]. Some simple and complex cells are not only sensitive to the spatial orientation of a visual stimulus, but also to the direction of its motion; these cells are said to be *direction selective* [22, 25].

The discovery of these properties inspired the notion that these neurons operate as edge or line detectors. A simple cell's RF is reminiscent of a linear filter consisting of a single Gabor wavelet [30]: its response depends on the exact alignment of a stimulus bar or grating on the excitatory and inhibitory subfields of the wavelet [49, 87]. The response of a complex cell, instead, resembles the output of an *energy model*, a mechanism combining the response of a quadrature pair of Gabor wavelets[2] to produce a phase invariant response in a similar way that $\cos^2 x + \sin^2 x = 1$ [1, 40, 58, 115].

Multiple alternative and complementary network architectures have been proposed to explain the receptive fields of simple and complex cells, including the effects of recurrent intracortical connections [7, 14], but a detailed discussion of these is beyond the scope of this thesis. The next section will be dedicated to phenomenological, probabilistic models of the activity of cortical neurons in V1.

[2]Two Gabor wavelets with identical envelope function, frequency, and orientation but with a 90-deg phase difference.

3.2 Receptive-field models

Model of RFs can be divided into two categories: *mechanistic models*, conceived to explain how the observed RF properties may arise from the underlying anatomical structure of the system and *phenomenological models*, which abstract all the anatomical or physiological details and focus on the mathematical relationship between sensory stimuli and a neuron's evoked response. All models we are going to introduce rely on two assumptions: 1) sensory neurons compute their responses in a low dimensional subspace, interpreted as the spatiotemporal receptive field(s) of the neuron [101], comprising a small number of stimulus features; 2) spike generation is a probabilistic process, to account for the considerable variability exhibited by neural activity even in response to the repeated presentation of the same stimulus [11, 94, 109].

Stimuli and filters

Visual stimuli are usually presented as discrete time signals (e.g., think of the frame rate of a video signal). The stimulus at each time point t can be considered a high-dimensional vector $s[t]$ containing one value for each pixel in a video frame. We denote the size of the frame (the total number of pixels) as $n_{xy} = n_x \times n_y$. For simplicity, we assume that the response is measured with the same temporal precision as the stimulus and we represent it as a vector $\mathbf{r} \in \mathbb{N}^T$, where T is the length of the experiment (and the number of stimulus frames). Each component r_t is the spike count in time bin t, for $1 \leq t \leq T$. To simplify the notation in this introduction, we restrict our presentation to finite impulsive response (FIR) filters, which can conveniently be expressed in terms of matrix-vector dot products (one can opt for an infinite impulsive response, but the notation would not be as compact). This is equivalent to assuming that all the information required to explain the response at time t is contained in all stimulus frames extending up to n_d time delays in the past, therefore we collect all stimuli preceding each time point t in a single m-dimensional column vector

$$\mathbf{s}_t = \mathrm{vec}([s[t], s[t-1], \ldots, s[t-n_d]]), \tag{3.1}$$

where $m = n_{xy} \times n_d$ and $\mathrm{vec}(\cdot)$ is the vectorization operator stacking the columns of its input on top of one another (stimuli indexed by negative values are assumed to be identically equal to zero).

The output of a discrete-time FIR filter can be expressed as a vector dot-product between the stimulus vector $\mathbf{s}_t$ and a vector of filter weights $\mathbf{k}$ with the same layout as the stimulus:

$$\mathbf{k}^\top \mathbf{s}_t = \sum_{i=1}^{M} k_i s_{t,i},$$

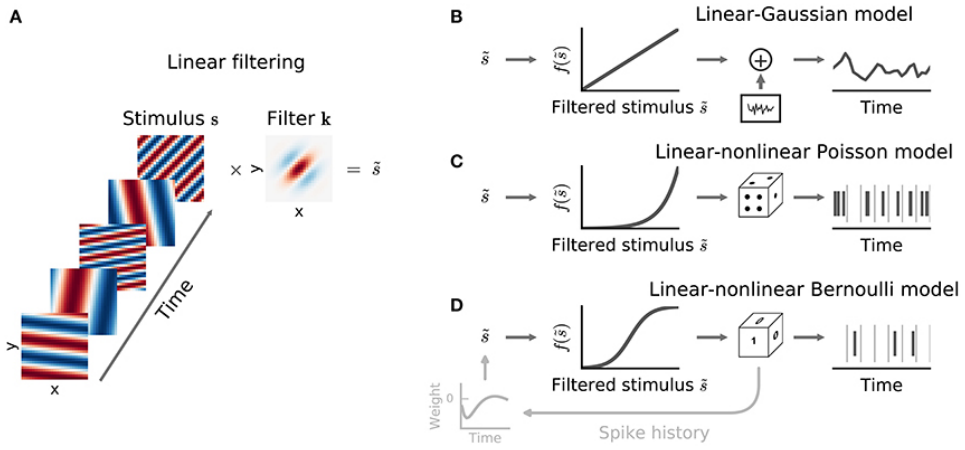

Fig. 3.1: **LN cascade models.**[4] **A)** Filtering of stimulus examples through the linear filter **k**. **B)** (Threshold-)Linear model with Gaussian noise. **C)** Poisson model with exponential nonlinearity. **D** Bernoulli model. All models can be extended using a post-spike filter that indicates dependence of the model's output on the recent response history (light gray).

where $s_{t,i}$ denotes the i'th entry in $\mathbf{s}_t$. Each value in $\mathbf{k}$ indicate the sensitivity of the neuron to inputs at the corresponding point in space and time (Fig. 3.1A).

3.2.1 Linear-Gaussian Models

The simplest model uses the output of one single filter, and a constant bias term (or offset), to model the expected value of the neural activity and generate the neural response from a normal distribution with constant variance and mean firing rate given by the filter output:

$$r_t \sim \mathcal{N}(\hat{r}_t, \sigma^2), \quad \hat{r}_t = \mathbf{k}^\top \mathbf{s}_t. \tag{3.2}$$

The bias term $k^{(0)}$ is absorbed into the vector $\mathbf{k}$ and the stimulus vector $\mathbf{s}_t$ is augmented with an entry equal to 1 at all times, so the offset becomes the coefficient associated to this extra dimension. The normality assumption is not appropriate for spike count data, because it fails to capture the fact that spike counts are non-negative integer numbers. However, it makes analytical derivations very convenient and several regularization techniques, which we will describe in a later section, rely on this assumption for this reason. This model can be extended with a thresholding linear function to filter out negative spike counts predictions (Fig. 3.1B).

[4] From Meyer *et al.* (2017) [68]. This content is licensed under the CC BY 4.0 license. The terms of the license are available at https://creativecommons.org/licenses/by/4.0/.

3.2.2 Linear-Nonlinear (LN) cascades

Linear-nonlinear cascade models [15] provide a useful framework for describing neural responses to high-dimensional stimuli and address the issue of modeling non-negative firing rates. These models define the spiking response in terms of the cascade of linear, nonlinear, and probabilistic stages. They extend the Gaussian model with a rectifying nonlinear stage before the probabilistic spike-generation process:

$$\hat{r}_t \propto f(\mathbf{k}^\top \mathbf{s}_t), \tag{3.3}$$

where $\hat{r}_t$ is the expected spike count in time bin t; $f : \mathbb{R} \to \mathbb{R}^+$ is a static, memory-less function mapping the filter's output to non-negative firing rates, which are assumed to be constant within a bin of size Δ; f accounts for nonlinearities like rectification and response saturation.

The most common instance of the LN cascade model is the Linear Nonlinear Poisson (LNP) model (Fig. 3.1C). In an LNP (), spike times are generated by an inhomogeneous Poisson point process [106] governed by the instantaneous firing rate $\lambda(t)$. The defining feature of a Poisson process is that responses in non-overlapping time bins are conditionally independent given the spike rate. The resulting distribution of spike counts within a bin of size Δ then follows a Poisson distribution:

$$p(r_t|\mathbf{s}_t, \mathbf{k}) = \frac{1}{r_t!}(\lambda_t \Delta)^{r_t} e^{-\lambda_t \Delta}, \quad \lambda_t = f(\mathbf{k}^\top \mathbf{s}_t). \tag{3.4}$$

If f is monotonic and fixed, then the above equation describes an instance of a Generalized Linear Model (GLM, see Appendix A) [66, 72], which extend multi-linear regression to noise models in the exponential family.

Sometimes, the Poisson distribution does not provide an accurate model of the variability observed in the data. In this cases, an alternative noise model can be adopted, like a Bernoulli (giving a Linear Nonlinear Bernoulli cascade) or a negative Binomial distribution.

It is worth mentioning that it is possible to model interactions between spikes in different bins to take into account cellular biophysical processes such as an absolute refractory period or accumulation or spike facilitation (Fig. 3.1D, gray inset) [110].

3.2.3 Multi-filter LN cascades

Multi-filter extension of the LN cascade are also possible. By replacing the single filter $\mathbf{k}$ with a tall matrix $\mathbf{K} = [\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_{n_l}]$, we can generalize a LN cascade model to RFs spanning a n_l-dimensional linear subspace of the stimulus (Fig. 3.2). Each filter represents a spatitemporal receptive field encoding a specific relevant feature of the high-dimensional stimulus. The definition of the instantaneous nonlinearity must of course be adapted to

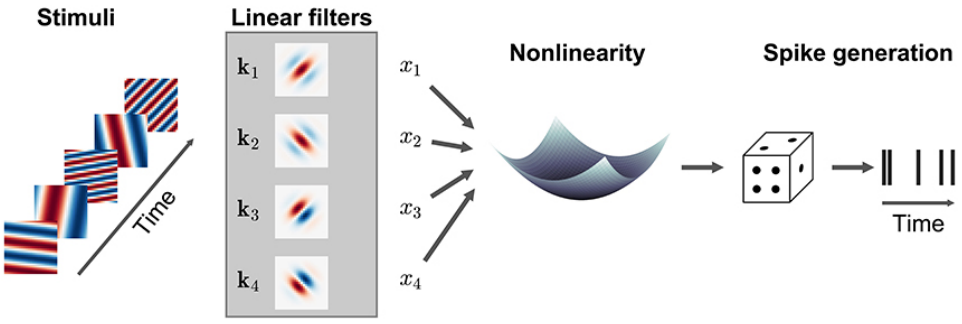

Fig. 3.2: **Multi-filter LNP model.**[6]The input stimulus $\mathbf{s}$ is projected onto a D-dimensional features subspace K representing the linear receptive fields of the neuron. The output of the filters is transformed by an instantaneous nonlinearity f, mapping the filtered stimulus to an instantaneous spike rate λ. Finally, spikes r are generated according to an inhomogeneous Poisson process.

take into account the dimensionality of the stimulus subspace: $f : \mathbb{R}^{n_l} \rightarrow \mathbb{R}^+$. The n_l-dimensional nonlinear behavior of f completely defines the nonlinear response properties of the neuron. In general, multi-filter LN cascade models are not GLMs and therefore, in general, they do not benefit from the convergence properties conferred by the GLM framework. An exception is posed by a specific type of quadratic models discussed below.

Generalized Quadratic Model (GQM)

In order to better understand the type of quadratic model discussed here, we should momentarily depart from the formalism we adopted so far to introduce a generalization of the linear model:

$$\hat{r}_t = k^{(0)} + \sum_{i=1}^{M} k_i^{(1)} s_{t,i} + \sum_{i,j=1}^{M} k_{ij}^{(2)} s_{t,i} s_{t,j} + \dots \tag{3.5}$$

This series expansion is a generalization of function polynomial series expansion to nonlinear filtering operators mapping one time series to another, known as the *Volterra expansion* [114]. The parameters $k^{(n)}$ are known as the Volterra kernels. We can generalize this input-output relationship to non-Gaussian observation noise by introducing a fixed non-linearity as we did for the LN cascade. If we truncate the expansion at the second order term, we obtain a Generalized Quadratic Model (GQM) [78]:

$$\hat{r}_t = f\Big(k^{(0)} + \mathbf{k}^{(1)\top}\mathbf{s}_t + \mathbf{s}_t^\top \mathbf{K}^{(2)}\mathbf{s}_t\Big), \tag{3.6}$$

[6]From Meyer *et al.* (2017) [68]. This content is licensed under the CC BY 4.0 license. The terms of the license are available at https://creativecommons.org/licenses/by/4.0/.

where $\mathbf{K}^{(2)} = \left[k_{ij}^{(2)}\right]$ is a compact matrix representation of the second order Volterra kernel. Although the mapping is nonlinear in the stimulus, it is linear in the kernel parameters, therefore the GQM is a GLM on the space of quadratically-transformed stimuli [32]. This means that the considerations about the geometry of the parameter landscape apply to the GQM as well.

The second Volterra kernel $\mathbf{K}^{(2)}$ of a GQM and the filter matrix $\mathbf{K}$ of a multi-filter LNP are two distinct objects and should not be confused. However, the two are related through a low rank decomposition of $\mathbf{K}^{(2)}$ [78]: consider the low-rank factorization $\mathbf{K}^{(2)} = \sum_{i=1}^{n_l-1} d_{ii}\mathbf{w}_i\mathbf{w}_i^\top = \mathbf{W}\mathbf{D}\mathbf{W}^\top$, where $\mathbf{W}$ is a (tall, skinny) $M \times (n_l-1)$ matrix with columns $\mathbf{w}_i$ and D is a diagonal matrix whose entries $d_{ii} \in \{-1, +1\}$ are constants that control the shape of the non-linearity along each axis in the feature space (-1 for suppressive and $+1$ for excitatory); then, the matrix $\mathbf{K} = [\mathbf{k}^{(1)}, \mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_D]$ encodes the linear subspace of a multi-filter LN cascade where the nonlinearity results from the composition of a quadratic function that maps the n_l-dimensional stimulus to the real line

$$z(\mathbf{s}) = k^{(0)} + \mathbf{k}^{(1)^\top}\mathbf{s} + \sum_{i=1}^{n_l-1} d_{ii}(\mathbf{w}_i^\top \mathbf{s})^2$$

and a 1-D nonlinearity $g(z)$. The full nonlinearity is thus $f(x) = g\big(z(x)\big)$.

3.3 Receptive field identification

After introducing the modeling framework, we will now focus on the estimation of the model parameters. Receptive field estimators can be coarsely subdivided into three distinct classes: moment-based, model-based (or likelihood-based) estimators and information-theoretic estimators. All approaches rely on the dimensionality-reduction assumption we have already introduced when talking about models of the neural activity in the previous section: they all try to find a linear subspace of the stimulus, such that the probability of the response r_t depends uniquely on the linear projection of the stimulus $\mathbf{s}_t$ on said subspace.

Within the commonly adopted LNP framework, information-theoretic estimators have been shown to be equivalent to likelihood-based estimators [116], therefore we will focus on the latter. Since we are concern with Bayesian inference for RF model parameters, model-based estimators provide the necessary context to understand this work. Moment-based estimators are also important, since they are widely adopted for their mathematical simplicity.

3.3.1 Moment-based methods

By characterizing the moments of the spike-triggered stimuli ensemble (STSE) distribution $p(\mathbf{s}|\mathbf{r})$, moment-based estimators seek to identify the kernels of the Volterra expansion (3.5). The spike-triggered average (STA) and the spike-triggered covariance (STC) that we are going to discuss below, estimate the first and the second order terms, respectively.

Spike Triggered Average (STA)

The STA provides an estimate of a neuron's linear receptive field. As the name suggests, the STA is given by the average stimulus in the spike-triggered ensemble (Fig. 3.3) [15, 101]:

$$\boldsymbol{\mu} = \frac{1}{n_{sp}} \sum_{t=1}^{T} t_i \mathbf{s}_t, \tag{3.7}$$

where $n_{sp} = \sum r_i$ is the total number of spikes in the dataset. The STA is an estimate of the cross-correlation between the stimulus and the neural activity. In computing (3.7) we assume that the stimulus has zero mean (i.e. $\mathbb{E}[\mathbf{s}] = 0$) – if not, it can be made so by subtracting the mean from each vector. We can express eq. (3.7) can be more compactly using matrix notation:

$$\boldsymbol{\mu} = \frac{1}{n_{sp}} \mathbf{S}^\top \mathbf{r}, \tag{3.8}$$

where $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_T]^\top$. The STA is the first term in the Volterra kernel series expansion of an LNP neuron's transfer function and can be used to estimate the linear stage of the LNP cascade model [101]. However, it provides an unbiased estimate of a neuron's receptive field only if the stimulus distribution is spherically symmetric (i.e. white noise) [15, 99]. If the stimulus presents non-zero correlations across space or time, the estimated RF is distorted [75]. In this case one can whiten the STA by left-multiplying it with the inverse of the stimulus covariance matrix:

$$\boldsymbol{\mu}_w = \frac{T}{n_{sp}} (\mathbf{S}^\top \mathbf{S})^{-1} \mathbf{S}^\top \mathbf{r} = (\mathbf{S}^\top \mathbf{S})^{-1} \boldsymbol{\mu}. \tag{3.9}$$

This is equivalent to the linear least-squares regression of the stimulus against the spike train, which is the maximum likelihood estimate (MLE) for the model parameters of the linear Gaussian model (3.2). The whitened STA is a consistent estimator (i.e., it converges to the true linear subspace) of the subspace spanned by the linear filter of an LNP if 1) the stimulus distribution is elliptically symmetric and 2) the expected STA is not zero, i.e. the nonlinearity induced a shift in the spike triggered-stimulus distribution. Furthermore, if the stimulus distribution is Gaussian and the the

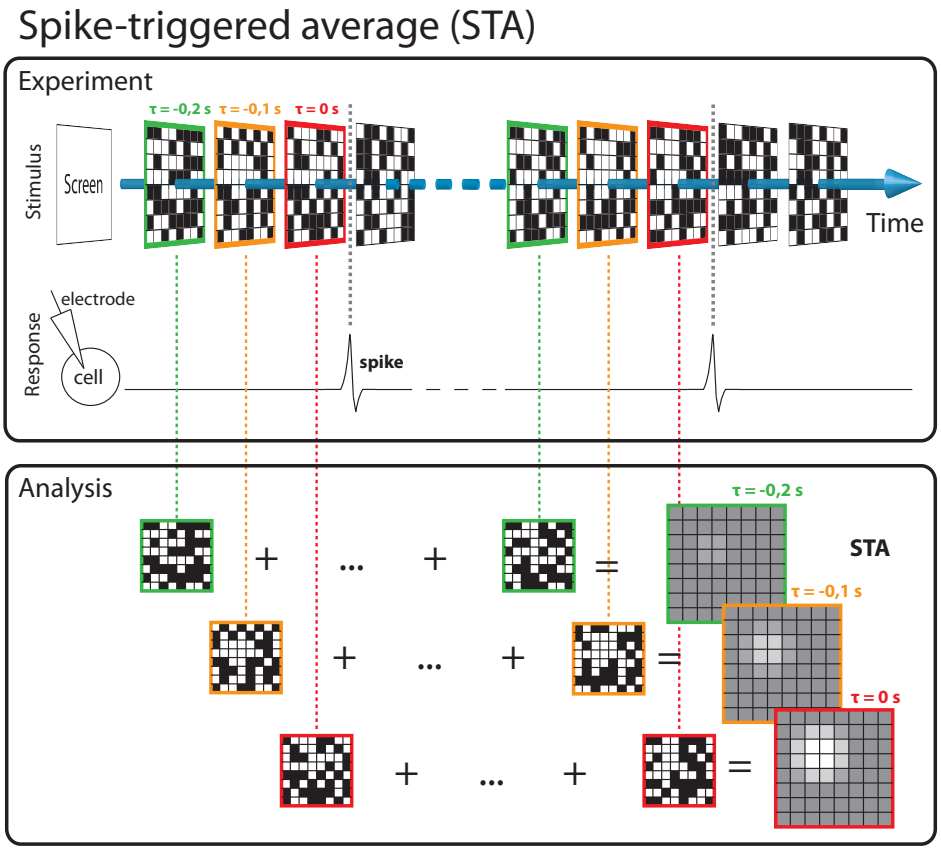

Fig. 3.3: **Spike-Triggered Average.**[8]Spikes are recorded while a stimulus (represented here by the checkerboard patterns) is presented. The stimulus preceding each spike is extracted (here, the 3 lags denoted by the color boxes) and the resulting stimuli subset are averaged. The STA suggests that this neuron is selective for bright spots of light located in the top left corner of the checkerboard.

neuron's nonlinear response function is the exponential, the whitened STA is an asymptotically efficient estimator [75]. For arbitrary stimuli, the STA is in general not consistent nor efficient.

Spike Triggered Covariance

STC analysis provides a complementary tool to STA for estimating linear filters in an LNP cascade model, but, unlike STA, it can be used to identify a multi-dimensional feature space. The STC, as the name suggests, is the co-

[8]Created by StphTphsn and published in Wikipedia at the following link under the CC BY-SA 4.0 license. The terms of the license are available at https://creativecommons.org/licenses/by-sa/4.0/deed.en

variance of the spike-triggered stimulus ensemble. Given the spike-triggered covariance

$$\mathbf{\Lambda} = \frac{1}{n_{sp}-1}\sum_{t=1}^{T} r_t(\mathbf{s}_t - \boldsymbol{\mu})(\mathbf{s}_t - \boldsymbol{\mu})^\top, \tag{3.10}$$

and the raw stimulus covariance (assuming the stimulus has zero mean)

$$\mathbf{C} = \frac{1}{T-1}\sum_{t=1}^{T} \mathbf{s}_t\mathbf{s}_t^\top,$$

STC analysis identifies the stimulus features affecting a neuron's response via an eigenvector decomposition of $(\mathbf{\Lambda} - \mathbf{C})$ [98]. Eigenvectors with eigenvalues significantly positive or negative correspond to stimulus axes along which the neural response is enhanced or suppressed.

The requirements imposed on the stimulus for the STC to provide a consistent estimator of the relevant stimulus subspace, are stricter than for the STA. The STC estimate is unbiased provided the overall stimulus distribution is spherically or elliptically symmetric (as was the case for the STA estimator of a single-filter model) and the stimulus dimensions are independent or can be linearly transformed to be independent of each other [75, 98]. These conditions are met only by a Gaussian stimulus distribution, and in other cases the bias can be very significant [75].

3.3.2 Model-based estimators

Maximum likelihood

The consistency of model-based depends on specific assumptions about the nature of the stimulus and the type of transformation mapping the input stimulus to the observed output response. Maximum likelihood estimators (MLEs), on the other hand provide consistent estimates of the RF's filters, independently of the choice of the stimulus distribution. This approach rely on explicit generative models of neural activity, such as those introduced in the previous section. The relationship between the stimulus and the corresponding evoked response is encoded in the conditional probability distribution $p(\mathbf{r}|\mathbf{S}, \boldsymbol{\theta})$, where $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_T]^\top$ is the stimulus matrix. For simplicity, we lumped together all model parameters (the filters and any parameter of the nonlinearity) in one single vector parameter $\boldsymbol{\theta}$. This distribution acts as likelihood function of the model parameters given the observed data $\mathcal{D} = \{\mathbf{S}, \mathbf{r}\}$, therefore the value of $\boldsymbol{\theta}$ that can best explain the data is the one maximizing the likelihood:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_{\mathrm{ML}} &= \arg\max_{\boldsymbol{\theta}} p(\mathbf{r}|\mathbf{S}, \boldsymbol{\theta}) \\ &= \arg\max_{\boldsymbol{\theta}} \big[\log p(\mathbf{r}|\mathbf{S}, \boldsymbol{\theta})\big]\end{aligned} \tag{3.11}$$

Introducing the logarithm operation does not change the position of the maximum, but makes inference easier by making the cost function more numerically stable. For many common choices of f, the log-likelihood of an LNP is a concave function [76], therefore a single global optimum exists and it can be found by standard optimization techniques such as gradient ascent or Newton's method.

MLEs are prone to suffer from noise in the data, which they tend to overfit when the number of parameters per data point ratio is high. For example, we already mentioned that the whitened STA is the MLE for a linear, Gaussian model. Despite providing an unbiased estimate, the whitening procedure has the undesired effect of amplifying noise at high spatial frequencies, therefore increasing the variance of the estimator and its data requirements. For a model with of tenths of thousands of parameters like those used to characterize a visual receptive field[9], the amount of data required for an accurate estimate is experimentally unfeasible. With limited data, optimizers tend to fit also random fluctuations, leading to poor estimates of RF parameters. This problem is exacerbated when experimental time budgets are very tight or the population of neuron being analyzed is intrinsically low-firing, resulting on very sparse data available for inference.

Maximum *a posteriori* (MAP)

The necessity of preventing overfit led to the development of several regularized estimators that penalize implausible values of the model parameters. Similar considerations apply also to STA and STC analysis, for which regularized estimators have been developed [77, 78, 85]. Regularizers are often contextualized within a Bayesian framework, where a prior distribution $p(\boldsymbol{\theta}, \boldsymbol{\alpha})$ explicitly encodes prior knowledge about the system, where $\boldsymbol{\alpha}$ is a vector of hyper-parameters. For visual RFs, the prior may encode, for example, smoothness an locality. The regularized estimate is obtained by maximizing the (unnormalized) posterior belief $p(\boldsymbol{\theta}|\mathbf{r}, \mathbf{s})$, which is known as maximum a posteriori (MAP) estimator:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_{\mathrm{MAP}} &= \arg\max_{\boldsymbol{\theta}} p(\mathbf{r}|\mathbf{S}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{\alpha}) \\ &= \arg\max_{\boldsymbol{\theta}} \big[\log p(\mathbf{r}|\mathbf{S}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}, \boldsymbol{\alpha}) \big]. \end{aligned} \tag{3.12}$$

Again, taking the log does not change the location of the maxima, but it makes the optimization computationally easier. Since multiple observations

[9] A multi-filter LN cascade has $n_d n_{xy} n_l$ RF parameters. For a stimulus of 100×100 pixels, 50 temporal delays (spanning 1s at 50Hz) and a 10-dimensional subspace, this amounts $5 \cdot 10^6$M RF parameters. A GQM (with full-rank $\mathbf{K}^{(2)}$) has $n_d^2 n_{xy}^2 + n_d n_{xy}$ model parameters, which, for the same stimulus as for the previous example, amounts to more than $50 \cdot 10^9$ parameters.

are usually considered conditionally independent given the predicted instantaneous firing rate (which in turn is a function of past stimuli and model parameters), the likelihood factorizes into a series of independent terms,

$$p(\mathbf{r}|\mathbf{S},\boldsymbol{\theta}) = \prod_{t=1}^{T} p(r_t|\mathbf{s}_t,\boldsymbol{\theta}), \tag{3.13}$$

where T is the number of observed data points. The log-likelihood term in (3.12) therefore scales linearly with T, whereas the magnitude of log prior is constant in the number of data points. For a small number of data points, the regularization effect of prior is not negligible and it pushes the estimates towards *a priori* expected solutions. This reduces the variance of the MAP estimator, while at the same time it helps preventing overfitting the noise. When T is large, the contribution of the prior becomes vanishingly small in comparison to the likelihood, and the MAP estimate converges to the ML estimate. This property makes the MAP estimate asymptotically unbiased and consistent.

The functional form of the prior determines the a priori expected structure of an RF. Assuming a Gaussian prior with covariance matrix $\boldsymbol{\Sigma} = \alpha\mathbf{I}$, strong weights are penalized – this is also known as *ridge regression*, or L_2 regularization. Using the same covariance matrix, but using a Laplace prior instead, is equivalent to performing *Lasso regression*, or L_1 regularization, which results in sparse estimate. Again using a Gaussian prior, but placing a separate penalty on each parameter, i.e. we assume a diagonal covariance such that $\sigma_{ii}^2 = \alpha_i^{-1}$, we get *Automatic Relevance Determination (ARD)* [108], which promotes sparse estimates as well; an ARD prior can also be imposed also on a per-filter basis to automatically selection the appropriate dimensionality of the feature space in a multi-filter LNP [78]. With *Automatic Smoothness Determination (ASD)* [96] the covariance matrix of a Gaussian prior is parameterized using a squared exponential kernel [92], in a way that the correlation between filter coefficients falls off as a function of their spatial distance; ASD is used to encourages spatial and temporal smoothness. Finally, with *Automatic Locality Determination (ALD)* [79] we can encourage spatiotemporally- and frequency-localized estimates. ALD contains ASD and ridge regression as special cases.

Selecting the hyper-parameters

In eq. (3.12), $\boldsymbol{\alpha}$ is assumed fixed. However, $\boldsymbol{\alpha}$ can be also made part of the inference procedure: its value can be chosen using a procedure known as Empirical Bayes (EB) or type II maximum likelihood [96], which consists of maximizing the evidence (also known as marginal likelihood)

$$p(\mathbf{r}|\mathbf{S},\boldsymbol{\alpha}) = \int p(\mathbf{r}|\mathbf{S},\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\alpha})\mathrm{d}\boldsymbol{\theta}. \tag{3.14}$$

The model parameters are then fitted using the MAP, according to:

$$\begin{aligned}\boldsymbol{\alpha}^* &= \arg\max_{\boldsymbol{\alpha}} p(\mathbf{r}|\mathbf{S},\boldsymbol{\alpha}) \\ \hat{\boldsymbol{\theta}}_{\mathrm{MAP}} &= \arg\max_{\boldsymbol{\theta}} p(\mathbf{r}|\mathbf{S},\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\alpha}^*).\end{aligned} \tag{3.15}$$

In ARD, ASD and ALD, this is how the optimal value of the hyper-parameters are automatically adjusted.

Alternatively, suitable hyper-parameters values can be set by cross-validating the model predicting performance on some held-out data [44, 63, 67].

Low-rank approximations

Introducing a regularization term is not the only way to facilitate the inference procedure. The amount of required data can be reduced by reducing the number of model parameters to be learned. This can be done by imposing a low-rank structure to the RF filters [80, 85]:

$$\mathbf{k}_l = \sum_{i=1}^{n_h}\sum_{j=1}^{n_g} a_{ijl}\big(\mathbf{h}_i \otimes \mathbf{g}_j\big), \tag{3.16}$$

where $\mathbf{h}_i$ and $\mathbf{g}_j$ denote temporal and spatial filters, respectively, and "$\otimes$" is the Kronecker product. Each $\mathbf{h}_i \otimes \mathbf{g}_j$ is a rank-1, space-time separable filter. This low-rank approximation of the linear filters reduces the number of receptive-field parameters from $n_d n_{xy} n_l$ to $n_d n_h + n_{xy} n_g + n_h n_g n_l$.

3.4 Summary and discussion

In this chapter we introduced the properties of visual cortical neurons, the target of our later analyses, and we described several probabilistic generative models for neural activity in response to external stimuli. We discussed moment-based and model-based estimators and defined the instances in which the former are equivalent to the latter. We discussed how the amount of data required by these estimators can be reduced thanks to properly chosen regularizers, which can be formalized within a Bayesian inference framework: for sparse or noisy data, these reduce the estimation variance by biasing the estimates towards *a priori* more likely receptive field shapes. Empirical Bayes methods provide state-of-the-art solutions to the receptive-field inference problem. However, despite their mathematical elegance, we identify three major drawbacks of this class of algorithms, which we will now discuss in detail.

Computational requirements

The first drawback, is represented by their high computational requirements: evidence optimization scales cubically with the number of model parameters and it is a serious computational bottleneck for these algorithms. Efficient implementations mostly rely on strong modeling assumptions, which considerably limit their application to a restricted number of scenarios. Available implementations are essentially limited to single-filter linear models with additive Gaussian observation noise [5, 79, 96], for which the marginal likelihood has a closed-form analytical solution. In all other cases, efficient evidence optimization is still an open matter of research. When not met by the data, these assumptions may lead to considerably biased estimates [68].

Recent work [44] showed that this limitation can be mitigated to a great extent by reducing the number of model parameters by modeling each receptive field filter as a linear combination of fixed, well-defined basis functions. Their choice fell on natural cubic splines, which rely on one single hyper-parameter – namely, the number of splines. The necessity of evaluating the marginal likelihood of the data was eliminated altogether, favoring a model-selection strategy based on cross-validation. This resulted in computationally efficient algorithms, which the authors showed to outperform the non-spline versions on a variety of applications.

Quantification of posterior uncertainty

The second limitation we can identify concerns the quantification of the estimation uncertainty encoded by the posterior distribution. Empirical Bayes relies on the assumption that the marginal likelihood of the data is a narrow peak around the optimal value of the hyper-parameters, that is $p(\boldsymbol{\alpha}|\mathbf{r},\mathbf{S}) \propto p(\mathbf{r}|\mathbf{S},\boldsymbol{\alpha}) \approx \delta(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)$, where $\delta(\cdot)$ is a Dirac's delta. Therefore all estimation uncertainty is correctly captured by the posterior. In mathematical terms,

$$p(\theta|\mathbf{r},\mathbf{S},\boldsymbol{\alpha}^*) \approx p(\theta|\mathbf{r},\mathbf{S}) = \int p(\theta|\mathbf{r},\mathbf{S},\boldsymbol{\alpha})p(\boldsymbol{\alpha}|\mathbf{r},\mathbf{S})\mathrm{d}\boldsymbol{\alpha}. \tag{3.17}$$

For very sparse and noisy data, the model evidence may not meet this assumption, and the posterior distribution $p(\theta|\mathbf{r},\mathbf{S},\boldsymbol{\alpha}^*)$ would strongly underestimate the actual amount of estimation uncertainty. To overcome this problem, a fully Bayesian treatment of the problem targeting $p(\boldsymbol{\theta},\boldsymbol{\alpha}|\mathbf{r},\mathbf{S})$ would be required. This is notoriously a computationally demanding problem, especially in very high-dimensional parameter spaces.

Propagation of uncertainty

With one or more parameters per stimulus dimension, non-parametric models are only informative of whether or not a specific stimulus dimension

is relevant for the computation performed by the neuron. Any high-level feature ϕ, e.g. the RF's location or spatial orientation, is not explicitly represented. This is actually not an exclusive problem of non-parametric models: even when the receptive field is modeled as a liner combination of basis functions, be them raised cosines [84] or splines [44], the issue still persists.

Usually, a functional receptive field model, which we represent here as a function $f(\phi)$, is fitted to point estimates such as $\mathbf{k}_{\text{MAP}}$ by minimizing the square loss $\|f(\phi) - \mathbf{k}_{\text{MAP}}\|^2$ [89, 93]. This is in general a non-linear optimization problem and, especially when $\mathbf{k}_{\text{MAP}}$ is noisy because the data itself was sparse or noisy, multiple optima may exist. Additionally, we should also take into account any uncertainty encoded by $p(\mathbf{k}|\mathbf{r}, \mathbf{S})$, which with sparse of noisy data would not be negligible, and propagate it to quantify $\text{Var}[\phi]$. This is an open problem, since we have already established that the mapping $\mathbf{k} \to \phi$ is not necessarily injective.

Conclusions

This third issue is strongly entangled with the second one. We propose to address both in an organic way, by modeling the RF features of interest explicitly and adopt a fully Bayesian approach. The benefits of this strategy are twofold: first, we drastically reduce the number of model parameters, which will be beneficial to help in the inference procedure [44]; second, we eliminate any intermediate or post-processing step, effectively getting rid of the uncertainty propagation problem at the root.

Chapter 4

A generative model for neurons in Primary Visual Cortex

In this chapter we will present a generative model for neural activity in primary visual cortex. Our aim is to develop a model that encompasses the most common stereotypical responses observed in primary visual cortex (V1) using a compact set of parameters. Idealized simple and complex cells can be described by spatially oriented Gabor wavelets [1, 49, 87]. We will generalize this notion and develop a low-rank receptive field model able to reproduce the most salient properties of neural responses in V1.

This chapter is structured as follows. After introducing a high-level overview of the model, we describe in detail the structure of the receptive field model. We will then study how this model responds to particular classes of stimuli often used in neuroscience. We will then describe an efficient discrete implementation of the model. Finally, we will discuss its relation with other models found in the literature.

4.1 Generative model

Our generative model is an instance of a Linear Nonlinear Poisson (LNP) cascade, [1] and is defined by the following equations:

$$r_t \sim \text{Poisson}(\lambda_t \Delta), \quad \lambda_t = \eta\big(a + \mathbf{b}^\top \tilde{\mathbf{s}}_t + \tilde{\mathbf{s}}_t^\top \mathbf{C}\, \tilde{\mathbf{s}}_t\big), \quad \tilde{\mathbf{s}}_t = F\{\mathbf{s}_t\}. \tag{4.1}$$

As in Chapter 3, $\mathbf{s}_t$ and r_t denote, respectively, the stimulus and the response of the neuron, here modeled in terms of spike counts within a bin of size Δ. The raw stimulus $\mathbf{s}_t$ is mapped to a 2-dimensional feature vector $\tilde{\mathbf{s}}_t$ by a linear operator $F\{\,\cdot\,\}$, which we will discuss next. The instantaneous firing rate of

[1] For details about LNP models, see Chapter 3.

the neuron λ_t is related to $\tilde{\mathbf{s}}_t$ by a nonlinear mapping $\eta : \mathbb{R}^2 \to \mathbb{R}^+$, after the quadratic projection defined by the parameters a, $\mathbf{b}$ and $\mathbf{C}$, respectively a scalar, a vector and a matrix.

4.1.1 Linear filter

The actual receptive field is represented by the linear operator $F\{\cdot\}$ in eq. (4.1). Here, we will provide its definition in continuous space and time, as the investigation of the filter's response properties results more amenable to an analytical treatment in this domain. The actual implementation, however, must be discretized in space and time to deal with high-dimensional, unstructured visual stimuli such as sequences of images or movies. The discussion of the discretization procedure is reserved to a later section.

The linear operator $F\{\cdot\}$ consists of a pair of space-time oriented, parametric, linear filters f_1 and f_2. This particular structure allows to tune the response of the model to visual stimuli with arbitrary orientation and spatial phase, using only a minimal set of model parameters. [2] Each filter computes a weighted sum of the stimulus intensities, over recently past time and local space, and mathematically it is modeled as the time-sliding dot product

$$\tilde{s}_i(t) = F_i\{s\}(t) = \int_{\mathbb{R}^2} \int_0^t f_i(\mathbf{x}, t' - t)\, s(\mathbf{x}, t')\, \mathrm{d}\mathbf{x}\, \mathrm{d}t', \tag{4.2}$$

where $s(\mathbf{x}, t)$ is a spatio-temporal signal $s : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$, denoting the stimulus in continuous space and time, assumed identically equal to zero for $t < 0$. The two kernels are modeled as follows:

$$f_1(\mathbf{x}, t) = h_5(-t) g_c(\mathbf{x} - \mathbf{x}_o) + k_{\mathrm{dir}} h_3(-t) g_s(\mathbf{x} - \mathbf{x}_o) \tag{4.3a}$$

$$f_2(\mathbf{x}, t) = h_5(-t) g_s(\mathbf{x} - \mathbf{x}_o) - k_{\mathrm{dir}} h_3(-t) g_c(\mathbf{x} - \mathbf{x}_o) \tag{4.3b}$$

The core of the model consists of four space-time separable units obtained by combining two spatial kernels $g_c(\mathbf{x})$ and $g_s(\mathbf{x})$ with two temporal kernels $h_3(t)$ and $h_5(t)$ (Fig. 4.1, top row). The spatial kernels g_c and g_s form a quadrature pair, i.e. their Fourier transforms are identical up to a 90° phase shift, whereas h_3 and h_5 have different temporal dynamics (their exact functional form is given below). Alone, these units provide orientation-selective responses with different temporal onset and spatial phase selectivity. In order to model direction selectivity, these units must be combined to give a non-separable spatiotemporal profile [1] (Fig. 4.1, bottom row). The strength of the effect is controlled by the scalar parameter k_{dir}, which is assumed to be less than 1 in magnitude. For $k_{\mathrm{dir}} = 0$, we recover the slow space-time separable units in the top row of Fig. 4.1, resulting in a receptive field which is insensitive to the motion direction of the stimulus.

[2] As mentioned in Chapter 3, simple and complex cells in primary visual cortex can be roughly classified into orientation and direction selective [22, 47].

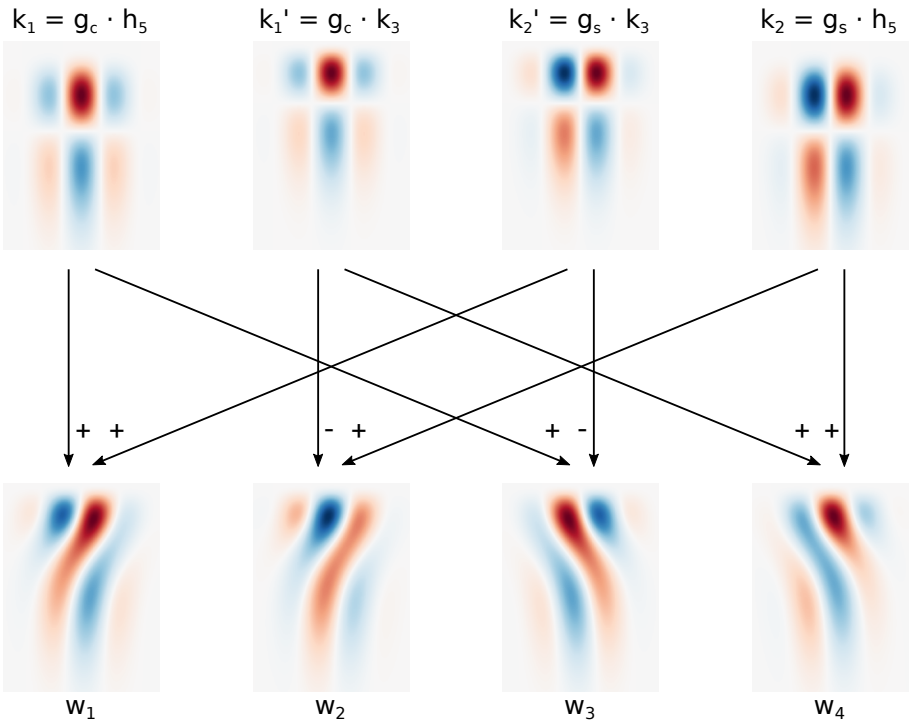

Fig. 4.1: **Separable and non-separable filters.** The top row displays a xt-section of the four space-time separable units of the model. Appropriate linear combinations of these units result in non-separable filters used for motion detection (bottom row).

For $k_{\text{dir}} = 1$ we recover the w_1 and w_2 non-separable kernels in the bottom row of Fig. 4.1; conversely, for $k_{\text{dir}} = -1$, we recover w_3 and w_4, which are selective for the opposite direction. In order to avoid a singular parameterization which will introduce artificial modes in the posterior distribution – by changing the sign of k_{dir} and rotating the spatial filters by 180° we obtain the same filters – for all practical purposes we constrain k_{dir} in the range $[0, 1]$. Finally, the 2-dimensional parameter $\mathbf{x}_o$ determines the spatial location of the receptive field within the visual field.

Spatial filters

The spatial filters $g_c(\mathbf{x})$ and $g_s(\mathbf{x})$ are parameterized, respectively, as a cosine and a sine Gabor functions (see Fig. 4.2A), which are related to a complex-valued Gabor wavelet $g(\mathbf{x})$ according to $g(\mathbf{x}) = g_c(\mathbf{x}) - jg_s(\mathbf{x})$. The Gabor wavelet is parameterized as follows [30]:

$$g(\mathbf{x}) \stackrel{\text{def}}{=} \frac{|\boldsymbol{\Sigma}|^{-\frac{1}{2}}}{2\pi} \exp\left(-\frac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x}\right) \cdot \exp(\mathbf{k}^\top \mathbf{x} - \varphi) \tag{4.4}$$

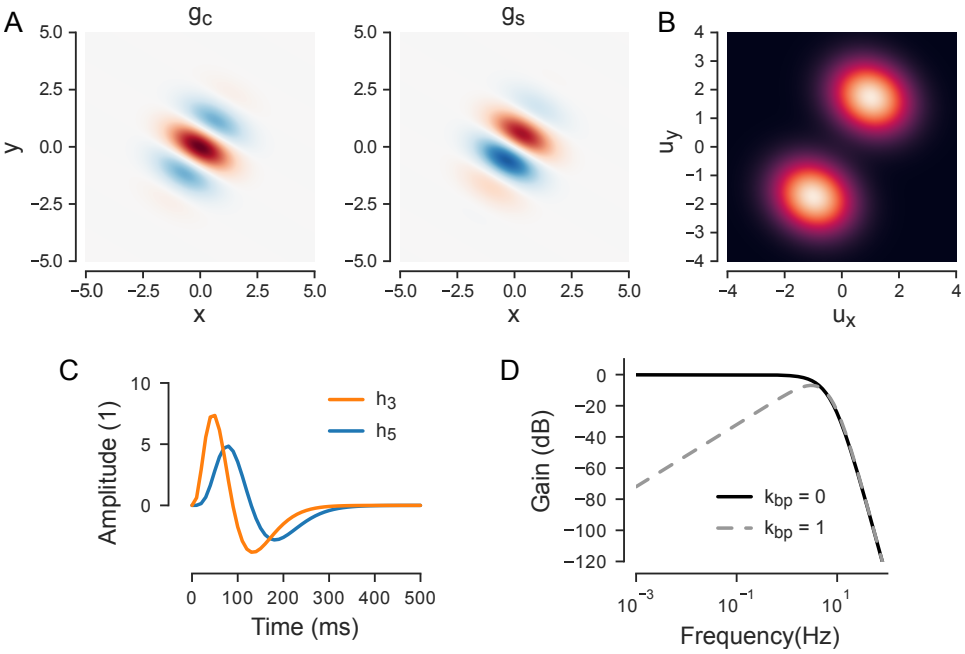

Fig. 4.2: **Spatial and temporal filters A)** Spatial kernels; cosine component (left) and sine component (right). **B)** Power spectrum of the spatial kernels; value computed according to eq. (B.3). **C)** Temporal kernels, for $k_{bp} = 1$; h_3 has a faster course than h_5. **D)** Power spectrum of h_5 for distinct values of k_{bp}; k_{bp} determines the type of frequency response (low-pass or band-pass).

where $\mathbf{k}_0 \in \mathbb{R}^2$, $\varphi \in [0, 2\pi]$ and $\mathbf{\Sigma} = \mathbf{R}^\top \mathbf{S} \mathbf{R}$ is a positive-definite, symmetric matrix with

$$\mathbf{S} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{pmatrix}.$$

Each parameter has a particular effect on the response properties of the spatial filter: $\mathbf{k}$ determines the preferred spatial frequency and orientation; φ_{RF} determines the preferred spatial phase; σ_x, σ_y and ψ determine the frequency and orientation bandwidth of the filter (Fig. B.1A and B). When $\psi = \angle\mathbf{k}$, σ_x and σ_y determine also the horizontal and vertical size of the filter in the local coordinate system. To keep the model general, the spatial frequency $\mathbf{k}$ and the covariance matrix $\mathbf{\Sigma}$ are introduced as two independent parameters [19, 49]. However, it is not uncommon to align one of the principal axes of $\mathbf{\Sigma}$ with the preferred spatial frequency, i.e. by constraining $\psi = \angle\mathbf{k}$. This choice reduces the number of spatial filter parameters from 7 to 6, but we lose the ability of modeling asymmetrical tuning curves. This parameterization is usually followed in computer vision literature for defining a bank of Gabor filters as feature extractors [29, 91], but also in computational neuroscience [20]. We will also follow the convention $\psi = \angle\mathbf{k}$ in later chapters, although it is important to mention that the model can be more expressive.

Temporal filters

The temporal subunits h_3 and h_5 are parameterized as the difference of two gamma function with the same rate, but different shape parameters (Fig. 4.2C):

$$h_n(t) \triangleq \alpha \frac{(\alpha t)^n}{n!} \left(1 - k_{\text{bp}} \frac{(\alpha t)^2}{(n+2)!}\right) e^{-\alpha t} \cdot \Theta(t), \tag{4.5}$$

where $\Theta(t)$ is the Heaviside step function, $\alpha > 0$ is a rate parameter that determines the frequency cutoff of the filter and $k_{bp} \in [0, 1]$ interpolates between a low-pass ($k_{bp} = 0$) and band-pass ($k_{bp} = 1$) response (Fig. 4.2D) [115]. The free parameters of the model are α and k_{bp}, while n is fixed. All parameters being equal, The two kernels have different temporal profiles, with h_5 being slower than h_3. These functions can model the stereotypical biphasic temporal responses observed in V1, with the amplitude of the second through controlled by the parameter k_{bp}. Triphasic temporal profiles, which are seldom encountered in V1, could be modeled using a more complex impulsive response by including of additional parameters. Since this type of response is quite rare [20], we sacrificed some extra flexibility in order to minimize the number of model parameters to facilitate inference in very small datasets for the majority of cells.

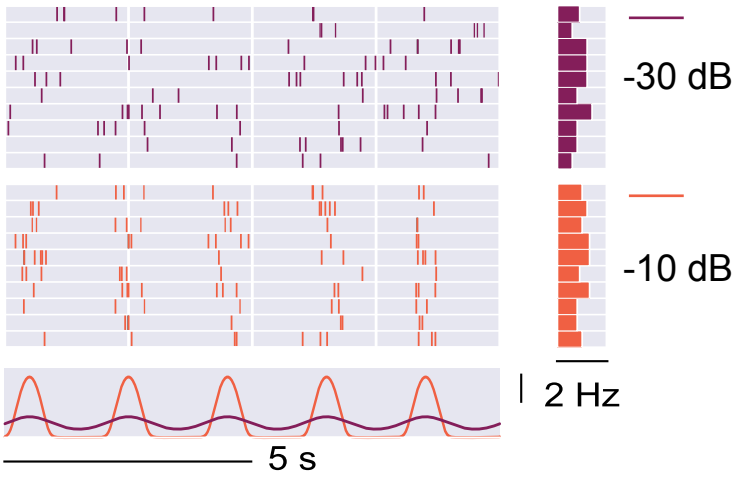

Fig. 4.3: **Stimulus sensitivity.** By changing the relative magnitude of the nonlinearity parameters a, $\mathbf{b}$ and $\mathbf{C}$ we can model how tightly the firing rate of the neuron is coupled to the visual stimulus. Here, we simulated the response of this model to a filtered stimulus consisting of a sinusoidal signal to different values of the nonlinearity parameters (raster plots), resulting in a poorly responsive (top) and well responsive (bottom) unit. The values of the parameters were chosen to result in the same average firing rate (1 Hz). The corresponding instantaneous firing rates are reported in the bottom panel: notice how the fluctuations of the purple line are less pronounced than those of the orange line.

4.1.2 Static Nonlinearity

While simple cells are tuned to different phases of the stimulus and their response properties can be captured by a linear model [70], complex cells exhibit some degree of phase invariance [45] and are responsive to motion energy in the stimulus [1, 112, 115]. This latter set of properties can be captured by a so-called *energy mechanism*[3], the response of which is not sensitive to the relative position of visual features [1, 88]. The quadratic function

$$z(t) = a + \mathbf{b}^\top \tilde{\mathbf{s}}(t) + \tilde{\mathbf{s}}(t)^\top \mathbf{C}\, \tilde{\mathbf{s}}(t), \tag{4.6}$$

which is part of the static nonlinearity of the LNP model defined in eq. (4.1), is a parsimonious way to implement both linear and non-linear behaviors. It is equivalent to a second-order Volterra expansion [3] of the actual nonlinear operation performed by a neuron, and contains both the linear model and the energy model as special cases: the first by imposing $\mathbf{C} = \mathbf{0}$, while the latter with $\mathbf{b} = \mathbf{0}$ and $\mathbf{C} = c \cdot \mathbf{I}$, for some positive real-valued scalar c. By changing the relative magnitude of a and the stimulus-related parameters $\mathbf{b}$ and $\mathbf{C}$,

[3] An energy model is a mechanism summing the squared outputs of a quadrature pair.

we model different levels of sensitivity to the visual stimulus (Fig. 4.3).

4.2 Response properties

In this section, we are going to investigate the response properties of the receptive field model defined above. We will analytically derive its response to a family of signals routinely used to assess the properties of visual receptive fields, namely contrast-modulated and moving sinusoidal gratings. We will also study the role played by the non linearity. As we shall see in the next paragraph, this analytical treatment is easier if we operate in the complex-valued domain: we will define a complex valued operator W mapping a visual stimulus $s(\mathbf{x}, t)$ to a complex-valued time series $\tilde{s}(t)$ such that $\mathrm{Re}[\tilde{s}(t)] \equiv s_1(t)$ and $\mathrm{Im}[\tilde{s}(t)] \equiv s_2(t)$, where $s_1(t)$ and $s_2(t)$ are the outputs of the linear subunits as defined by eq. (4.2).

A central role in the analysis that follow is played by the notion of *power* of a complex-valued periodic signal $z(t)$, which is defined as the average value of its squared magnitude within one period of length T:

$$P_z = \frac{1}{T} \int_{-T/2}^{+T/2} |z(t)|^2 \mathrm{d}t. \tag{4.7}$$

From the power, we can derive the another quantity of interest, the root-mean-squared (RMS) amplitude of the signal,

$$z_{RMS} \triangleq \sqrt{P_z}, \tag{4.8}$$

which is the amplitude a constant signal should have to deliver the same power of the periodic one within the same amount of time. **NB:** The power of a signal is a constant. It should not be confused with the power spectrum of the signal, $P_z(\omega) = |\hat{z}(\omega)|^2$, i.e. the squared magnitude of its Fourier transform, which is instead a function measuring the amount of energy delivered at each frequency.

4.2.1 Complex-valued receptive field model

As mentioned above, our analysis will be greatly facilitated by operating in the complex domain. We define a new complex-valued, space-space time separable operator

$$W\{s\}(\mathbf{x}, t) = \int_0^t \int_{\mathbb{R}^2} \overline{g(\mathbf{x}' - \mathbf{x})} h(t - t') s(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \, \mathrm{d}t', \tag{4.9}$$

where $g(\mathbf{x})$ is the complex-valued Gabor wavelet (4.4) and

$$h(t) = h_5(t) - j \cdot k_{\mathrm{dir}} h_3(t), \tag{4.10}$$

with $h_n(t)$ defined as in eq (4.5). With some derivations, we can show that

$$\tilde{s}(t) = W\{s\}(\mathbf{x}_o, t) = F_1\{s\}(t) + j \cdot F_2\{s\}(t), \tag{4.11}$$

where F_1 and F_2 are the real-valued linear operators associated to the real-valued kernels (4.3a) and (4.3b). Thanks to the complex notation, the interactions of four real-valued, space-time separable subunits are compactly represented by one single complex-valued, space-time separable operator, which greatly reduces the complexity of our analysis. This new linear operator is the cascade of a cross-correlation with kernel $g(\mathbf{x}')$ and a convolution with kernel $h(t')$. The complex signal $\tilde{s}(t)$ can be found by evaluating the operator's output at $(\mathbf{x}_o, t)$, for all values of t, which is a one-dimensional, complex-valued temporal signal. From the linearity and the separability of the operator, we can find that its response to a complex-valued harmonic signal

$$\xi(\mathbf{x}, t) = A \cdot e^{j\mathbf{k}_0^\top \mathbf{x}} e^{j\omega_0 t} \tag{4.12}$$

with amplitude $A \in \mathbb{C}$, spatial frequency $\mathbf{k}_0$ and temporal frequency ω_0 has the following simple analytical expression:

$$W\{\xi\}(\mathbf{x}, t) = A\, \hat{h}(\omega_0)\overline{\hat{g}(\mathbf{k}_0)}\, e^{j\mathbf{k}_0^\top \mathbf{x}} e^{j\omega_0 t} = \hat{h}(\omega_0)\overline{\hat{g}(\mathbf{k}_0)} \cdot \xi(\mathbf{x}, t), \tag{4.13}$$

where $\hat{g}(\cdot)$ and $\hat{h}(\cdot)$ are the Fourier transforms of the spatial and temporal kernels, respectively, given in eq. (B.1) and eq. (B.12). This means that $\xi(\mathbf{x}, t)$ is an eigenfunction for W. From this, it follows that the receptive field's response to $\xi(\mathbf{x}, t)$ is a temporal harmonic signal $\tilde{\xi}(t) = \dot{A}\, e^{j\omega_0 t}$, where the complex amplitude is $\dot{A} = A\, \hat{h}(\omega_0)\overline{\hat{g}(\mathbf{k}_0)}\, e^{j\mathbf{k}_0^\top \mathbf{x}_o}$. Since harmonic signals are a building block for more sophisticated stimuli, if follows that we can decompose the receptive field's response to such stimuli in terms of simpler, analytically available signals.

4.2.2 Orientation and frequency tuning

The orientation-selectivity properties of the model can be investigated by studying the linear filter's response to a counterphase sinusoidal grating, which is a class of stimuli commonly used to characterize the responsive properties of cortical visual neurons. This type of signal is a static (i.e. non moving), contrast-modulated sinusoidal grating; the contrast is modulated by a sinusoidal temporal signal. Mathematically:

$$s(\mathbf{x}, t) = A \cdot \cos(\mathbf{k}_0^\top \mathbf{x} - \varphi_s) \cos(\omega_0 t - \varphi_t) = A \cdot s_x(\mathbf{x}) \cdot s_t(t), \tag{4.14}$$

where $\mathbf{k}_0 \in \mathbb{R}^2$, $\varphi_s \in [0, 2\pi]$ and $A \in \mathbb{R}^+$, are the spatial frequency, phase and amplitude of the grating, while $\omega_0 \in \mathbb{R}$ and $\varphi_t \in [0, 2\pi]$ are the angular frequency and phase of the contrast-modulating signal. The response of the linear filter factorizes into a spatial and a temporal sub-components:

$$W\{s\}(t) = \frac{1}{4} A \cdot \tilde{s}_x(\mathbf{x}_o) \cdot \tilde{s}_t(t),$$

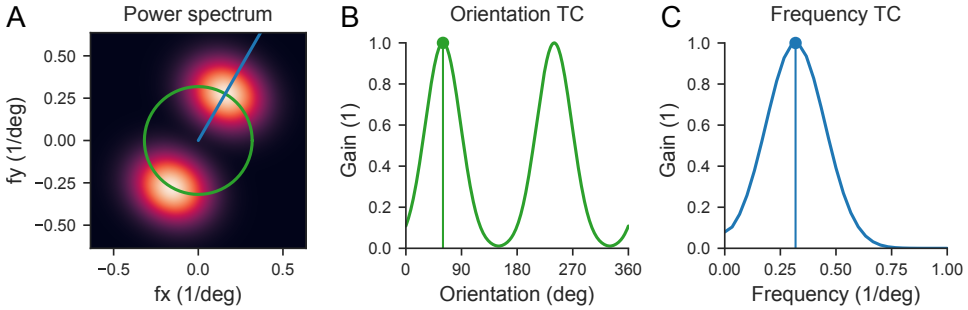

Fig. 4.4: **Orientation selectivity. A)** Power spectrum of the Gabor wavelet corresponding to the spatial filters in Fig. 4.2A. The green and the blue lines correspond to the slices used to evaluate the spatial orientation and spatial frequency responses. **B)** Spatial orientation selectivity. **C)** Spatial frequency selectivity.

where $\tilde{s}_x$ and $\tilde{s}_t$ are the spatial and the temporal component of the response, respectively. In this context, $\tilde{s}_x(\mathbf{x}_o)$ is a constant modulating the amplitude (and changing the phase) of the temporal signal $\tilde{s}_t(t)$. The value of its squared magnitude defines the selectivity of the receptive field to spatial properties of the stimulus. It is implicitly a function of the grating's parameters $\mathbf{k}_0$ and φ_s, which we denote with $\mathrm{tc}(\mathbf{k}_0, \varphi_s) \triangleq |\tilde{s}_x(\mathbf{x}_o)|^2$ and we refer to it as the tuning curve (TC) of the filter. For sufficiently large values of $|\mathbf{k}_0|$, the dependency on $\mathbf{x}_o$ and φ_s becomes negligible:

$$\mathrm{tc}(\mathbf{k}_0, \varphi_s) \approx |\hat{g}(\mathbf{k}_0)|^2 + |\hat{g}(-\mathbf{k}_0)|^2. \tag{4.15}$$

This is a mixture of two 2D Gaussians centered at $\pm\mathbf{k}_{RF}$ (From here on, we will use the subscript "RF" to denote those filter parameters having the same name of some of the stimulus parameters). For sufficiently large values of the product $|\mathbf{k}_{RF}|\sigma_x$, the two lobes are well separated and the function has two local maxima at $\mathbf{k}_0^* \approx \pm\mathbf{k}_{RF}$ (see Fig. 4.4A), corresponding to a preferred orientation $\theta_0^* = \angle\mathbf{k}_{RF} + n\pi$ (for $n \in \mathbb{Z}$) and a preferred spatial frequency $\kappa_0^* = |\mathbf{k}_{RF}|$ (where κ_0 and $theta_0$ denote the polar coordinates of $\mathbf{k}_0$, such that $\mathbf{k}_0 = \kappa_0 \angle\theta_0$). By evaluating the slice in the spatial frequency plane with constant magnitude $|\mathbf{k}_0| = \kappa_0^*$ we can recover the *orientation tuning curve* of the filter (Fig. 4.4B)). Similarly, by fixing the orientation $\angle\mathbf{k}_0 = \theta_0^*$ and varying the magnitude of the spatial frequency, we obtain its *frequency tuning curve* (Fig. 4.4C). In general, the orientation and frequency tuning curves do not have a compact analytical representation, which may limit the understanding of the effect that each model parameter has on the selectivity properties of the filter. However, they can be readily evaluated for any given value of the stimulus and of the model parameters, just by following their definition. This means that they can be fitted to data, if one so desires.

4.2.3 Direction selectivity

The direction selectivity properties of the filter can be investigated by studying its response to a drifting grating, i.e. a sinusoidal plane wave:

$$s(\mathbf{x}, t) = A \cdot \cos(2\pi\nu(\mathbf{n}^\top \mathbf{x} - ct) - \varphi_0). \tag{4.16}$$

The spatial frequency ν is the number of cycles per unit length along the direction $\mathbf{n}$, c is the drifting velocity and φ_0 is the initial phase shift when $t = 0$. For $c > 0$, the drifting direction is $\mathbf{n}$, whereas for $c < 0$ it is $-\mathbf{n}$. When $c = 0$, there is no drift and the grating is static. By rearranging the terms in eq. (4.16), we can parameterize the grating in terms of the 2D angular spatial frequency $\mathbf{k}_0 = 2\pi\nu\,\mathbf{n}$ and the angular temporal frequency $\omega_0 = 2\pi\nu c$. Using the trigonometric identity $2\cos x = e^{jx} + e^{-jx}$, we can express the drifting grating as the sum of two complex harmonic signals:

$$s(\mathbf{x}, t) = \xi(\mathbf{x}, t) + \overline{\xi(\mathbf{x}, t)}, \tag{4.17}$$

where $\xi(\mathbf{x}, t) = A' e^{j\mathbf{k}_0^\top \mathbf{x}} e^{-j\omega_0 t} e^{-j\varphi_0}$ and $A' = A/2$. Unlike the case of a counterphase grating, we cannot factorize the filter response into the product of a spatial and of a temporal component, because the input signal is not space-time separable. We must instead compute the responses to ξ and $\overline{\xi}$ individually. Thanks to eq. (4.13), however, we can derive the response

$$\tilde{s}(t) = \hat{h}(-\omega_0)\overline{\hat{g}(\mathbf{k}_0)}\xi(\mathbf{x}_o, t) + \hat{h}(\omega_0)\overline{\hat{g}(-\mathbf{k}_0)\xi(\mathbf{x}_o, t)} \tag{4.18}$$

From here, deriving the expression for the power is just a matter of a few steps (see Appendix B.3 for details), from which we obtain

$$P_{\tilde{s}} = \big[P_h(-\omega_0)P_g(\mathbf{k}_0) + P_h(\omega_0)P_g(-\mathbf{k}_0)\big] \cdot P_\xi, \tag{4.19}$$

where $P_g(\mathbf{u}) = |\hat{g}(\mathbf{u})|^2$ and $P_h(\omega) = |\hat{h}(\omega)|^2$ denote, respectively, the power spectra of the spatial and the temporal filter, derived from the Fourier transforms (FT) of the respective kernels given in eq. (B.1) and (B.12). How exactly each model parameters shapes $P_{\tilde{s}}$ is not straightforward,[4] but we can nevertheless get a high-level qualitative understanding of this relationship. If the model is space-time separable, i.e. $k_{\text{dir}} = 0$, $h(t)$ is a real-valued function, therefore $\hat{h}(t)$ is Hermitian, from which follows that $P_h(\omega)$ is an even function; in this case, $P_{\tilde{s}}$ has the following simpler expression:

$$P_{\tilde{s}} = P_h(\omega_0)\big(P_g(\mathbf{k}_0) + P_g(-\mathbf{k}_0)\big) \cdot P_\xi. \tag{4.20}$$

Since $P_h(\omega_0)$ is an even function (because it is the power spectrum of a real valued signal) and $P_g(\mathbf{k}_0) + P_g(-\mathbf{k}_0)$ is even by construction, the filter

[4] We can compute $P_{\tilde{s}}$ for any choice of model and grating parameters using indeed eq. (4.19).

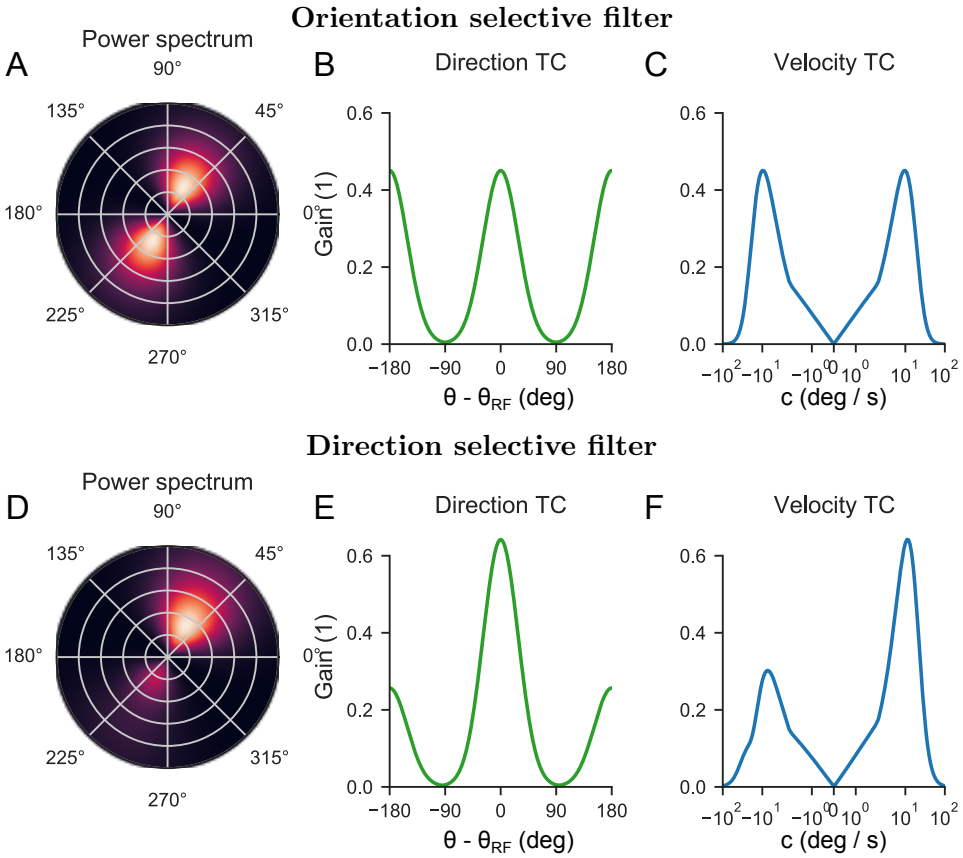

Fig. 4.5: **Direction selectivity. A)** Velocity power spectrum of the RF for $k_{\text{dir}} = 0$. The angular dimension is the direction of motion of the grating and the radial dimension is the speed in (deg/s); radial grid lines are spaced in steps of 10 deg/s. **B)** Direction selectivity: using $c = c^*$, the preferred speed, we vary the orientation of the grating. **C)** Speed selectivity: using a grating with $\angle \mathbf{x}_0 = \angle \mathbf{k}_{RF}$, we evaluated the strength of the response for different values of the drifting speed. **D, E, F)** As A, B and C, but for a direction selective filter obtained by setting $k_{\text{dir}} = 1$.

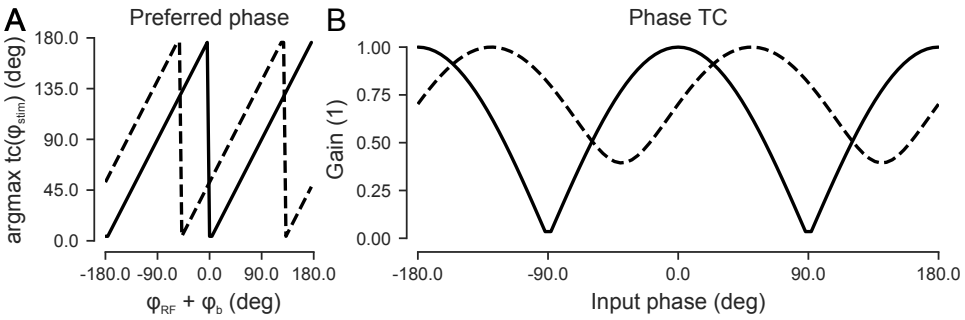

Fig. 4.6: **Phase selectivity. A)** Preferred spatial phase as a function of the sum $\varphi_{RF} + \angle\dot{B}$ for $k_{\text{dir}} = 0$ (solid line) and $k_{\text{dir}} = 1$ (dashed line). **B)** Strength of the response of $\dot{B}\,W\{s\}(t)$ for a counterphase grating input $s(t)$, as a function of the phase of the grating, for $k_{\text{dir}} = 0$ (solid line) and $k_{\text{dir}} = 1$ (dashed line). We measured the output RMS and normalized for the maximum observed value in each configuration.

response is equally strong up to a 180° flip of the drifting velocity and the resulting model is not direction selective (see Fig. 4.5A, B and C). When instead $k_{\text{dir}} \neq 0$, opposite directions will elicit different responses. More precisely, if $k_{\text{dir}} > 0$ and $\angle\mathbf{k}_0 = \angle\mathbf{k}_{RF}$, the response will be stronger for $c > 0$ than for $c < 0$ (see Fig. 4.5D, E and F). Conversely, the opposite relationship is true if $k_{\text{dir}} < 0$.

4.2.4 Phase dependence and phase invariance

After investigating the selectivity properties of the linear filter, we are now going to study the role of the quadratic nonlinearity in shaping the selectivity of the model. To start, denoting as usual the complex-valued output as $\tilde{s}(t)$ and the corresponding real-valued feature vector as $\tilde{stim}(t)$, we note that

$$\mathbf{b}^\top \tilde{\mathbf{s}}(t) + \tilde{\mathbf{s}}(t)^\top \mathbf{C}\, \tilde{\mathbf{s}}(t) = \text{Re}\Big[\dot{B}\, \tilde{s}(t) + \dot{C}\, \tilde{s}(t)^2\Big] + D\, |\tilde{s}(t)|^2 \tag{4.21}$$

for some constants $\dot{B}, \dot{C} \in \mathbb{C}$ and $D \in \mathbb{R}$. We will now study separately the contribute of the linear and of the quadratic terms.

Preferred spatial phase

The net effect of the linear term $\mathbf{b}^\top \tilde{\mathbf{s}}(t)$ is to adjust the phase selectivity of the linear part of the response. In other words, the parameter $\mathbf{b}$ determines the relative position of excitatory and inhibitory subregions of the spatial filter. Mathematically, this can be expressed as

$$\mathbf{b}^\top \tilde{\mathbf{s}}(t) = b_1 \cdot F_1\{s\}(t) + b_2 \cdot F_2\{s\}(t) = \text{Re}\big[\dot{B}\, \tilde{s}(t)\big] \tag{4.22}$$

for $\dot{B} = b_1 + j \cdot b_2$. If we expand the term $\dot{B}\,\tilde{s}(t)$, we see that the effect of multiplying by a complex-valued constant is to offset the spatial phase of the receptive field by $\angle\dot{B}$. In other words, this is equivalent to a new spatial filter $g'(\mathbf{x})$ with spatial phase $\varphi'_{RF} = \varphi_{RF} + \angle\dot{B}$, such that $\dot{B}\,\tilde{s}(t) = |\dot{B}|\,\tilde{s}'(t)$, where $\tilde{s}'(t)$ is the response of this new receptive field. The preferred spatial phase φ^* of the filter for different values of $\varphi_{RF} + \angle\dot{B}$ is reported in Fig. 4.6A: for $k_{\text{dir}} = 0$, $\varphi^* = \varphi_{RF} + \angle\dot{B}$ (solid line), up to a 180° shift for negative values; for $k_{\text{dir}} = 1$ we observe the same pattern, but with a displacement of 45° (dashed line); in both cases the relation is periodic with period 180°. This plot shows that only the sum $\varphi_{RF} + \angle\dot{B}$ is identifiable, making the spatial phase parameter φ_{RF} redundant, since only the combined effect of $\varphi_{RF} + \angle\dot{B}$ can be observed in the output. We can therefore set $\mathbf{k}_{RF}$ to some constant arbitrary value, and let the phase response properties of the RF be determined uniquely by the parameter $\mathbf{b}$; in our implementation we used $\varphi_{RF} = 45°$, since for this value $g_c(\mathbf{x})$ and $g_s(\mathbf{x})$ have the same energy. The strength of the response of a linear filter with preferred spatial phase $\varphi^* = 0$ to a counterphase grating with spatial frequency $\mathbf{k}_0 = \mathbf{k}_{RF}$ as a function of the grating's phase φ_0 is illustrated in Fig. 4.6B: while a relative phase $\varphi_0 = \pm 90°$ completely kills the output $\dot{B}\tilde{s}(t)$ for $k_{\text{dir}} = 0$ (solid line) because the contributions from excitatory and inhibitory subregions of the RF completely cancel out, this is not the case for $k_{\text{dir}} = 1$ (dashed line), which leads to a model with a less pronounced sensitivity to the spatial phase of the stimulus.

Non-linear response and phase invariance

We now focus on the quadratic term

$$z_{sqr}(t) = \tilde{\mathbf{s}}(t)^\top \mathbf{C}\,\tilde{\mathbf{s}}(t) = \text{Re}\big[\dot{C}\,\tilde{s}(t)^2\big] + D\,|\tilde{s}(t)|^2. \tag{4.23}$$

After expanding and rearranging the term $\tilde{\mathbf{s}}(t)^\top \mathbf{C}\,\tilde{\mathbf{s}}(t)$, we can show that $|\dot{C}| = \sqrt{2\,\text{tr}(\mathbf{C}^2) - \text{tr}(\mathbf{C})^2}$, $\angle\dot{C} = -\arctan(2c_{12}, c_{11} - c_{22})$, and $D = \text{tr}(\mathbf{C})$, where "tr(·)" denotes the trace operator. The term $D\,|\tilde{s}(t)|^2$ is actually proportional to the power of the linear filter's output, and it does not depend on the specific phase of the input, as confirmed also by simulating this signal for different values of the spatial phase of a counterphase sinusoidal grating stimulus (Fig. 4.7A). This term has a net excitatory or suppressive effect depending of the sign of tr($\mathbf{C}$) (positive, excitatory; negative, suppressive). The other term, instead, contains higher-order harmonics of the input. Expanding this terms gives:

$$\text{Re}\big[\dot{C}\tilde{s}(t)^2\big] = |\dot{C}||\tilde{s}(t)|^2 \cos\big(2\angle\tilde{s}(t) + \angle\dot{C}\big).$$

For an input with a sinusoidal temporal component oscillating with frequency ω_0, this is a sinusoid with temporal frequency $2\omega_0$ (Fig. 4.7B).

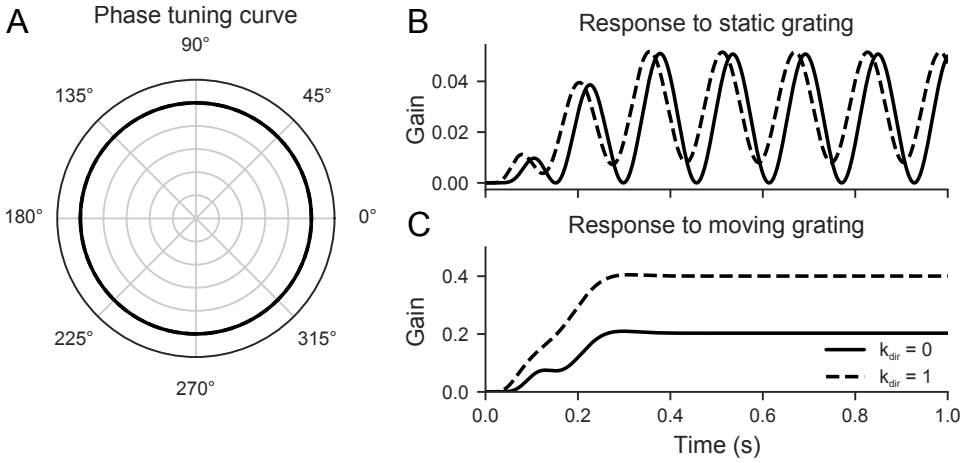

Fig. 4.7: **Phase invariance. A)** Normalized output power of the response to a counterphase sinusoidal input with spatial frequency $\mathbf{k}_0 = \mathbf{k}_{RF}$; the angular dimension is the phase of the stimulus while the radial dimension is the normalized gain; grid lines are spaced in steps of 0.2 units. Notice how the gain is constant and equal to 1 as function of the stimulus phase. **B** Response to a static grating for $k_{\text{dir}} = 0$ (solid line) and for $k_{\text{dir}} = 1$ (dashed line); the amplitude of the response follows the modulation of the contrast. **C)** Response to a rightward moving grating for a receptive field with preferred direction $\theta^* = 0°$ for $k_{\text{dir}} = 0$ (solid line) and for $k_{\text{dir}} = 1$ (dashed line); in this case, after an initial transient, the strength of the response is more or less constant over time.

This effect goes under the name of *frequency doubling* and it is observed in complex-like cells in V1 [20]. The response to a drifting grating, instead, after an initial transient is a sustained signal (Fig. 4.7C).

4.3 Discretization of the model

So far we adopted a continuous space-time formulation of the receptive field model, as this facilitate the analytical investigation of its properties. In concrete applications, however, visual stimuli often consist of discrete signals: for example, a movie is a discrete series of frames, which in turn are a matrix of pixels. These can be thought as piece-wise constant signals defined on a finite spatiotemporal grid, and can be represented using 3-dimensional tensors.

The most straightforward way to discretize the linear filters is to take advantage of the discrete nature of the stimulus and covert all integrals into sums. For any space-time separable subunits like those forming the kernels (4.3), this operation would yield:

$$\tilde{s}_n = \sum_{k=0}^{n} h_{n-k} \sum_{i,j} g_{ij} s_{ijk}. \tag{4.24}$$

The coefficients h_k and g_{ij} are obtained by integrating the continuous kernels on the grid defining the piece-wise constant visual stimulus. If we truncate the temporal convolution to the first M terms, eq. (4.24) is equivalent to a vector dot product between the receptive field filter $\mathbf{h} \otimes \mathbf{g}$ and the stimulus vector $\mathbf{s}_n$, defined as in eq. (3.1) (here, "$\otimes$" denotes the Kronecker's product). Assuming $\mathbf{g} \in \mathbb{R}^{N_{xy}}$ and taking into account the separable structure, the time required to filter a stimulus sequence of length N_t therefore scales as $O(N_t(M + N_{xy}))$. Although conceptually simple, this approach has one main disadvantage: the value of M at which we truncate the sum grows, in general, inversely with the sampling step size Δ and the decaying rate of the filter.[5]

A more efficient efficient strategy relies on converting the continuous time filter $h(t)$ to a discrete time one $h_d[n]$. This can be done by transforming the transfer function of the continuous-time system, $H(s) = \mathcal{L}\{h\}(s)$, into the transfer function of the discrete-time system, $H_d(z) = \mathcal{Z}\{h_d\}(z)$,[6]through a mapping of the continuous frequency plane S to the discrete frequency plane Z [74]:

$$s \leftarrow \Delta^{-1} \ln z \approx \frac{2}{T} \frac{z-1}{z+1}. \tag{4.25}$$

[5]This fact is better illustrated through an example. Consider an exponentially decaying kernel $h(t) = \alpha \exp(-\alpha t)$. The corresponding discrete kernel is $h[k] = (1-a)a^k$, with $a = \exp(-\alpha\Delta)$. If we consider only the first M samples, we discard $\epsilon = a^M$ mass of the kernel. For a desired level of precision ϵ, $M = \lceil -(\alpha\Delta)^{-1} \ln \epsilon \rceil$, where $\lceil \cdot \rceil$ denote the smallest integer larger than its argument.

Operating this mapping is equivalent to approximating the value of the impulsive response $\tilde{h}_d[n]$ using the trapezoidal rule [74], i.e.

$$h_d[n] = \int_{(n-1)\Delta}^{n\Delta} h(\tau)\mathrm{d}\tau \approx \frac{\Delta}{2}\big[h((n-1)\Delta) + h(n\Delta)\big],$$

which gives an approximation error that scales as $o(\Delta^2)$ [113]. Treating the convolution as a discrete-time linear time-invariant system, however, has an additional advantage: the discrete-time system can be expressed by the linear recursive difference equation

$$y[n] = -\sum_{i=1}^{P} a_i \cdot y[n-i] + \sum_{j=0}^{P} b_j \cdot u[n-j], \tag{4.26}$$

where u is the input and y is the output of the filter. The computational cost of this implementation scales as $O(N_t(P + N_{xy}))$, where P is the order of the filter, which depends only on the functional family of the kernel $h(t)$. In other words, the order of the system is completely independent of the discretization time step or the value of any of the kernel's parameters (for example, the slowest temporal filter used in our model, h_5, has order $P = 8$ and does not depend on the value of α or k_{bp}).

4.4 Relation to other models

The low-rank structure of our receptive field model shares many similarities with other models of motion perception in V1 [1, 39, 40, 115]. The organization of the receptive field in linear subunits and their interactions generalizes the structure of the energy models developed by Adelson & Bergen [1] and Watson & Ahumada [115] as an explanation motion perception in complex cells. By parameterizing the interactions between the output of the two linear subunits, instead of assuming a fixed quadratic relationship, the resulting structure can be used to model the linear behavior characteristic of simple cells as well. In more recent year, low-rank receptive fields models have been adopted to reduce the number of model parameters in single-filter LNP [80], and as a basis for an efficient implementation of information-theoretic spike-triggered average and covariance analysis [85].

This generative model can be interpreted as a generalized quadratic model (see eq. (3.6)):

$$k^{(0)} + a, \quad \mathbf{k}^{(1)} = \mathbf{W}\mathbf{b}, \quad \mathbf{K}^{(2)} = \mathbf{W}^\top \mathbf{C}\mathbf{W},$$

where $\mathbf{W} = [\mathbf{f}_1, \mathbf{f}_2]$ and $\mathbf{f}_1$, $\mathbf{f}_2$ are column vectors representing the linear subspace spanned by the receptive field. This formulation is reminiscent of

[6] $\mathcal{L}$ and $\mathcal{Z}$ denote the Laplace- and the Z-transform [74], respectively.

the elliptical-LNP model introduced by Park and Pillow [78]. This model can also be treated as a GQM on the projected stimulus $\tilde{\mathbf{s}}$, where the linear filter plays the role of a family of parametric basis functions extracting relevant features from the stimulus. As all GQMs, this also can be considered a GLM on the space of quadratically transformed stimuli [32]. The corresponding design matrix and GLM coefficients are:

$$\boldsymbol{\beta} = [a,\, b_1,\, b_2,\, c_{11},\, c_{22},\, c_{12}]^\top, \tag{4.27}$$

$$\mathbf{x} = [1,\, \tilde{s}_1,\, \tilde{s}_2,\, \tilde{s}_1^2,\, \tilde{s}_2^2,\, 2 \cdot \tilde{s}_1 \tilde{s}_2]. \tag{4.28}$$

This equivalence will be fundamental to enable the application of the efficient nested sampling implementation introduced in the next chapter.

4.5 Summary and discussion

In this chapter we introduced a linear-nonlinear cascade model capable of reproducing many of the stereotypical responses observed in V1, while relying on a small number parameters. The RF is modeled in terms of a cascade of two linear subunits (filters) and a static quadratic non-linearity. Spatial filters were modeled as Gabor functions because provide a good empirical description of receptive field shapes typically observed in V1 [49]. As a fully quadratic generalization of the energy model [1, 115], this model can reproduce orientation and direction selectivity, linear responses and phase invariance. Nevertheless, we realize that this is not an exhaustive model of all properties of V1 receptive fields and we discuss some possible extensions in appendix C.

By restricting the receptive field to a very specific parametric family, we can drastically reduce the number of parameters in comparison the other models discussed in Chapter 3 (see Table 4.1), even below the already small parameter space represented by a family of splines [44]. By reducing the number of parameters, we seek to make inference easier when data is scarce. Inference will be performed on semantically meaningful parameters without any intermediate post-processing stage, making inference results readily interpretable in terms of experimentally relevant receptive field properties. Still, this model can be treated within the GLM framework, a property that is crucial to speed up inference, as we will see in Chapter 5.

Parameter	Dim.	Constraint	Description
$\mathbf{x}_o$	2	-	Spatial location
$\mathbf{k}$	2	-	Preferred spatial frequency
σ_x	1	$\sigma_x > 0$	Horizontal scale
σ_y	1	$\sigma_y > 0$	Vertical scale
ψ	1	$[0, 2\pi]$	Affects the orientation tuning
α	1	$\alpha > 0$	Decay rate of temp. filter
k_{bp}	1	$[0, 1]$	Frequency response type (BP/LP)
k_{dir}	1	$[0, 1]$	Direction selectivity

Table 4.1: **Receptive field parameters.** The receptive field is entirely modeled by the 8 parameters reported in this table, for a total of 10 degrees of freedom (DoF), or 9, when enforcing $\psi = \angle\mathbf{k}$.

Chapter 5

Nested Sampling for GLMs with parametric basis functions

In this chapter, we will present a variation of Nested Sampling (NS) developed to target the posterior distribution over model parameters of Generalized Linear Models (GLMs) where the design matrix is specified by means of parametric basis functions. We named this new algorithm "Collapsed Nested Sampling" (CNS), after the strategy used to compute the evidence: CNS improves sampling efficiency by marginalizing out some of the model parameters, therefore reducing the effective dimensionality of the parameter space explored by the sampler. In order to make the distinction between the two algorithms clear, from here on we refer to conventional NS as "ordinary nested sampling" (ONS). We will first explain the motivations for developing CNS, followed by a mathematical characterization of its structure. We will test CNS and ONS on synthetic data generated using the receptive field model presented in chapter 4. We will compare CNS' performance against ONS on a variety of metrics, to characterize its convergence properties.

5.1 Motivation

Nested sampling is a popular alternative to MCMC algorithms to simultaneously estimate a model's evidence and sample from the posterior distribution of its parameters (for an overview of the NS algorithm, see Chapter 2.3). Relying on a very small number of free parameters, nested sampling can as a black-box Bayesian inference framework. The exact number of parameters vary across each specific implementation of the algorithm, but one particular parameter is common to all of them: this is the number of live points, N_{live}, i.e. the number of particles NS uses to explore the posterior landscape. The choice of N_{live} affects both the precision of the estimated model evidence

and of posterior distribution – which we here denote respectively by $\hat{\mathcal{Z}}$ and $\hat{p}(\boldsymbol{\theta}|\mathcal{D})$ to distinguish them from the respective true values. Higher values produce more precise estimates at the expense of additional computational cost. As mentioned already in Chapter 2.3, the computational complexity of NS scales as $O(N_{\text{live}})$ [103]. Choosing an appropriate value for N_{live} is of paramount importance: with too small values we sacrifice resolution, concretely increasing the chances of missing some narrow posterior modes (and as already mentioned in Chapter 2, this is not an only an issue of NS, but a problem affecting all sampling-based strategies to approximate a probability distribution); too large values, on the other hand, result in longer running times for negligible improvements.[1] Theoretically, N_{live} should scale at least as $O(D^2)$ to produce good estimates [105], where D is the number of model parameters; this, however, is just a suggested lower bound. In some practical implementations, having the number of live points scaling as $O(D^3)$ gives better performance [36]. This means that the computational complexity of NS scales supra-linearly with the number of model parameters. If it were possible to reduce the dimensionality of the parameters space, while targeting the same, identical posterior distribution, we could expect a reduction in NS' running time.

5.2 Collapsed Nested Sampling

Nested sampling is a Monte Carlo (MC) algorithm to compute the values of an integral like

$$\mathcal{Z} = p(\mathcal{D}|\boldsymbol{\theta}) = \int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} = \int p(\mathcal{D},\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}, \tag{5.1}$$

which, in the context of Bayesian inference, corresponds to the evidence in favor of a data-generating model $p(\mathcal{D}|\boldsymbol{\theta})$, governed by parameters $\boldsymbol{\theta}$ and producing data $\mathcal{D}$. Former knowledge or prior belief about $\boldsymbol{\theta}$ in encoded in the prior probability distribution $p(\boldsymbol{\theta})$. This equation is equivalent to (2.3), with the omission of the explicit dependence on the model. The integrand is the joint distribution of parameters and data, $p(\mathcal{D},\boldsymbol{\theta})$, which is proportional to the posterior distribution of parameters given the data:

$$p(\boldsymbol{\theta}|\mathcal{D}) \propto p(\mathcal{D},\boldsymbol{\theta}) = p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}).$$

Without any loss of generality, we can split the vector of model parameters $\boldsymbol{\theta}$ into two disjoint subsets, $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, and factorize the aforementioned joint distribution accordingly. Mathematically,

$$p(\mathcal{D},\boldsymbol{\theta}) = p(\mathcal{D},\boldsymbol{\theta}_1,\boldsymbol{\theta}_2) = p(\mathcal{D}|\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1)p(\boldsymbol{\theta}_1), \tag{5.2}$$

[1] For example, we are interested only on the value of some summary statistics of the posterior distribution, like its mean or variance, up to 1 part in 1000 and improving the precision to 1 part in 1 million would not bring any further practical advantage.

where

$$\boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \subset \boldsymbol{\theta} : \boldsymbol{\theta}_1 \cup \boldsymbol{\theta}_2 = \boldsymbol{\theta} \ \wedge \ \boldsymbol{\theta}_1 \cap \boldsymbol{\theta}_2 = \varnothing. \tag{5.3}$$

The posterior distribution allows a similar factorization:

$$p(\boldsymbol{\theta}|\mathcal{D}) = p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1, \mathcal{D})p(\boldsymbol{\theta}_1|\mathcal{D}) \propto p(\mathcal{D}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \tag{5.4}$$

If we substitute eq. (5.2) in (5.1), the expression for the model evidence becomes:

$$\mathcal{Z} = \int p(\boldsymbol{\theta}_1) \Big(\int p(\mathcal{D}|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1) \mathrm{d}\boldsymbol{\theta}_2 \Big) \mathrm{d}\boldsymbol{\theta}_1, \tag{5.5}$$

where the innermost integral is the marginal likelihood of the partial model $p(\mathcal{D}|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1)$ conditioned on the value of $\boldsymbol{\theta}_1$:

$$p(\mathcal{D}|\boldsymbol{\theta}_1) = \int (\mathcal{D}|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1) \mathrm{d}\boldsymbol{\theta}_2 \tag{5.6}$$

As such, it is the normalizing constant of the conditional posterior distribution $p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1, \mathcal{D})$ appearing in (5.4). Assuming $p(\mathcal{D}|\boldsymbol{\theta}_1)$ is known, we can recast the original problem expressed by (5.1) on a lower-dimensional parameter space, represented by the new integral

$$\mathcal{Z} = \int p(\mathcal{D}|\boldsymbol{\theta}_1) p(\boldsymbol{\theta}_1) \mathrm{d}\boldsymbol{\theta}_1. \tag{5.7}$$

Reformulating the problem in this way has two main advantages:

1. Since the dimensionality of $\boldsymbol{\theta}_1$, D_1 is smaller than that of the original set of model parameters, D, a smaller number of live points is required to explore the posterior landscape. Since N_{live} scales in general as some monotonically increasing, supra-linear function of the number of model parameters and the computational cost of NS scales as $O(N_{\text{live}})$, we can theoretically expect asymptotically shorter run times. This is true as long as there is no additional computational overhead for evaluating $p(\mathcal{D}|\boldsymbol{\theta}_1)$ instead of $p(\mathcal{D}|\boldsymbol{\theta}_2, \boldsymbol{\theta}_1)$: this may occur as additional computational costs for computing $p(\mathcal{D}|\theta_1)$ itself or as additional computational overhead from some internal step of the NS implementation.

2. We can expect $p(\mathcal{D}|\boldsymbol{\theta}_1)$ to be more efficient to explore. Resulting from the marginalization of some of the parameters of a higher dimensional model, $p(\mathcal{D}|\boldsymbol{\theta}_1)$ is smoother than the original $p(\mathcal{D}|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, i.e. it has comparatively broader and shallower peaks. This property makes it less likely for the nested sampler to miss a narrow mode or get stuck around a local maximum on its way to accumulate evidence across the posterior distribution.

We will show in the results that there is some evidence backing up the second point. Any consideration about the first point is instead a more delicate matter, since a series of factors determines the actual computational cost of NS beyond purely theoretical considerations and are potentially implementation-dependent – this point will be discussed in detail in the results section.

Evaluating $p(\mathcal{D}|\boldsymbol{\theta}_1)$

So far we have assumed that the value of $p(\mathcal{D}|\boldsymbol{\theta}_1)$ is readily available, but this is not generally the case: an analytical marginalization is not always possible. Nevertheless, since $p(\mathcal{D}|\boldsymbol{\theta}_1)$ is the normalization factor of $p(\boldsymbol{\theta}_2|\mathcal{D}, \boldsymbol{\theta}_1)$, its value may be approximated, e.g., with variational methods [8] or the Laplace's method[2][21], provided that $p(\boldsymbol{\theta}_2|\mathcal{D}, \boldsymbol{\theta}_1)$ meets the criteria for these approximations to be meaningful.

Sampling the marginalized parameters

If needed, samples from $p(\boldsymbol{\theta}_2|\mathcal{D})$ can be generated as follows: for each $\boldsymbol{\theta}_1^{(k)} \sim p(\boldsymbol{\theta}_1|\mathcal{D})$ generated by NS, sample $\boldsymbol{\theta}_2^{(k)} \sim p(\boldsymbol{\theta}_2|\mathcal{D}, \boldsymbol{\theta}_1^{(k)})$. This approach is similar to how marginalized parameters are sampled in a partially collapsed Gibbs sampler [111]. How exactly to sample from $p(\boldsymbol{\theta}_2|\mathcal{D}, \boldsymbol{\theta}_1^{(k)})$ depends on the concrete instance of the problem one is working on. If directly sampling the conditional distribution is not straightforward, samples could be generated using MCMC [31] or any other feasible sampling strategy, e.g., rejection sampling [8, 13], slice sampling [71] or by importance sampling, to approximate the posterior distribution $p(\boldsymbol{\theta}_2|\mathcal{D}, \boldsymbol{\theta}_1)$.

5.3 Application to Generalized Linear Models

Generalized linear models (GLMs) provide a convenient framework to extend multilinear regression to observation noise models in the exponential family [66, 72]. We have already mentioned them in Chapters 3 and 4, and Appendix A discusses them in detail.

In this chapter we are concerned in particular with GLMs which relate the dependent variable y on some vector of covariates $\mathbf{x}$ by means of a linear mixture of non-linear functions of $\mathbf{x}$:

$$g\big(\mathbb{E}[y]\big) = \beta_0 + \sum_{j=1}^{p} \beta h_j(\mathbf{x}), \tag{5.8}$$

where g is the link function (see Appendix A), $\beta_0, \beta_1, \ldots, \beta_p$ are linear mixing coefficients and $h_1(\cdot), \ldots, h_p(\cdot)$ are a set of (non-linear) basis functions [24, 32, 98], whose collective behavior is governed by a set of parameters, $\boldsymbol{\psi}$.

The likelihood of a data point $(\mathbf{x}_i, y_i) \in \mathcal{D}$ is therefore a function of the dependent variables $\mathbf{x}_i$, of the coefficients $\boldsymbol{\beta} = [\beta_0, \beta_1, \ldots, \beta_p]^\top$, and of the parameters $\boldsymbol{\psi}$. Considering a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$ consisting of n pairs of independent and dependent variables, the likelihood of the entire set of observations $\mathbf{y} = [y_1, y_2, \ldots,]^\top$ factorizes as the product of conditionally independent terms:

$$p(\mathbf{y}|\boldsymbol{\beta}, \boldsymbol{\psi}, \mathbf{X}) = \prod_{i=1}^{n} p(y_i|\boldsymbol{\beta}, \boldsymbol{\psi}, \mathbf{x}_i),$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top$. This likelihood being conditionally dependent on the disjoint set of model parameters $\boldsymbol{\theta} = \{\boldsymbol{\psi}, \boldsymbol{\beta}\}$ suggests that the posterior distribution $p(\boldsymbol{\beta}, \boldsymbol{\psi}|\mathbf{y}, \mathbf{X}) \propto p(\mathbf{y}|\boldsymbol{\beta}, \boldsymbol{\psi}, \mathbf{X})p(\boldsymbol{\psi}, \boldsymbol{\beta})$ can be explored using the strategy outlined in (5.5):

$$\mathcal{Z} = \int p(\boldsymbol{\psi}) \Big(\int p(\mathbf{y}|\boldsymbol{\beta}, \boldsymbol{\psi}, \mathbf{X}) p(\boldsymbol{\beta}) \mathrm{d}\boldsymbol{\beta} \Big) \mathrm{d}\boldsymbol{\psi}. \tag{5.9}$$

Here, we also assumed that the $\boldsymbol{\beta}$ and $\boldsymbol{\psi}$ are *a priori* independent, therefore the prior factorizes as $p(\boldsymbol{\psi}, \boldsymbol{\beta}) = p(\boldsymbol{\psi}) \cdot p(\boldsymbol{\beta})$, because there is no reason to believe that the linear mixing coefficients depends on the actual shape of the basis functions, or the other way around. Although we could have decided to marginalize $\boldsymbol{\psi}$, marginalizing out the coefficients $\boldsymbol{\beta}$ has a clear advantage: the structural properties granted by the GLM framework ensure that $p(\mathbf{y}, \boldsymbol{\beta}|p(\mathbf{y}|\boldsymbol{\beta}, \boldsymbol{\psi}, \mathbf{X})p(\boldsymbol{\beta})$ is log-concave in $\boldsymbol{\beta}$ provided that $p(\boldsymbol{\beta})$ is as well, therefore the innermost integral can be approximated using Laplace's method [21] when an analytical solution is not available:

$$p(\mathbf{y}|\boldsymbol{\psi}, \mathbf{X}) \approx \frac{(2\pi)^{\frac{m}{2}} e^{f(\hat{\boldsymbol{\beta}})}}{|H(\hat{\boldsymbol{\beta}})|^{\frac{1}{2}}},$$

where m is the dimensionality of $\boldsymbol{\beta}$, $f(\boldsymbol{\beta}) = \log p(\mathbf{y}, \boldsymbol{\beta})$, $\hat{\boldsymbol{\beta}} = \arg\max f(\boldsymbol{\beta})$ and H is the Hessian of f. For a Poisson GLM, log-concavity is preserved even when a non-canonical link function is used, provided it satisfies some mild regularity conditions [76].

5.4 Performance analysis

Both ordinary NS and CNS were used to infer the posterior distribution over model parameters on synthetic data generated by the receptive field model described in the previous chapter. Both NS and CNS were run different values of N_{live} and, for each configuration, each algorithm was run multiple times using different random seeds (for details regarding the data generation and the sampler configuration, see the methods section below). The

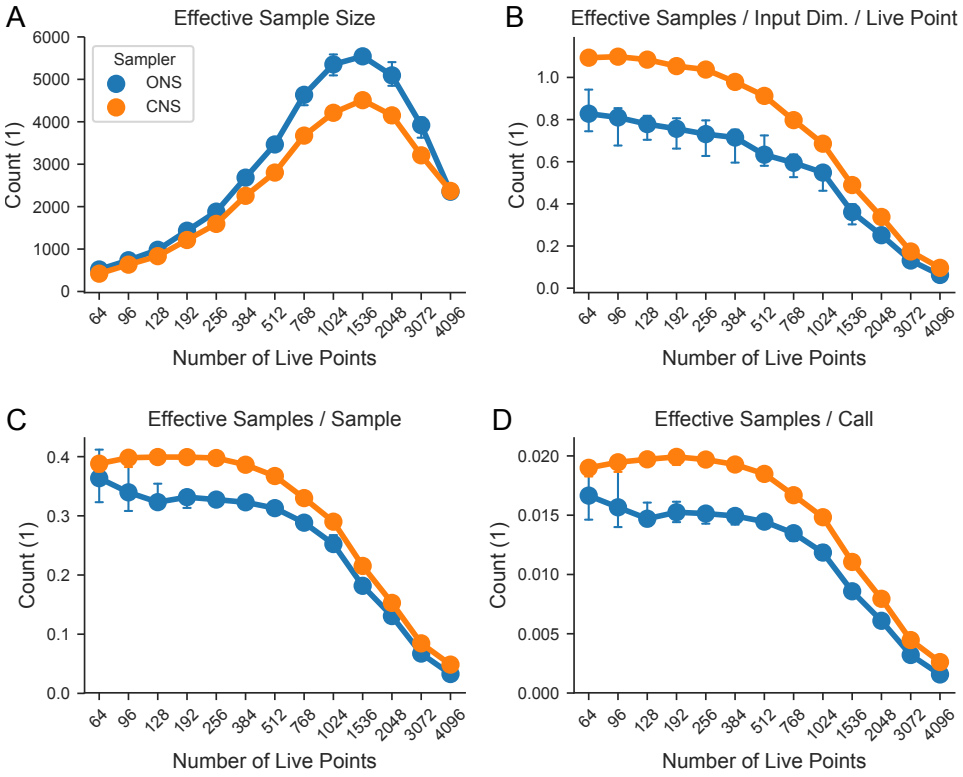

Fig. 5.1: **Sampling efficiency.** Comparing sampling efficiency, measured according to several metrics (blue, ordinary NS; orange, collapsed NS). **A)** Effective sample size as a function of N_{live}. **B)** Effective samples per live point per input dimension. **C)** Average number of effective samples generated by each actual sample. **D)** Average number of effective samples generated for each evaluation of the log-likelihood function. CNS performs better than ONS on all fronts, except A.

performance of both samplers were evaluated according to a set of different criteria: 1) sampler efficiency, 2) approximation error, 3) computational and memory costs. These will be addressed one by one in the following subsections.

5.4.1 Sampler Efficiency

As mentioned earlier, the number of iterations of a nested sampler, which equals the number of generated samples from the posterior distribution, scales linearly with N_{live}. Each sample $\boldsymbol{\theta}^{(i)}$ is associated with a weight, w_i, representing the amount of posterior mass associated with the corresponding likelihood shell. To every set of n weighted samples is associated an Effective Sample Size (ESS), n_{eff}, which gives the equivalent number of independent

samples in the set [55], computed as follows:

$$n_{\text{eff}} = \frac{(\sum_{i=1}^{n} w_i)^2}{\sum_{i=1}^{n} w_i^2}. \tag{5.10}$$

ONS generates consistently more effective samples than CNS and for both samplers, n_{eff} steadily increases with N_{live} up to $N_{\text{live}} = 1536$, and then it decreases (Fig. 5.1A). Since $N_{\text{live}} = 1536$ live points is the configuration delivering the highest number of effective samples, we will treat the corresponding NS/CNS outputs as a "surrogate" ground-truth posterior distribution for the rest of our analysis. Despite yielding a smaller number of effective samples, CNS is more efficient than NS in terms of number of effective samples per live point per input dimension (Fig. 5.1B), number of effective samples per generated sample (Fig. 5.1C), and number of effective samples per log-likelihood call (Fig. 5.1D. All three efficiency metrics considered decay for increasingly large values of N_{live}. This seems to be an intrinsic behavior of a nested sampler when targeting a narrow likelihood (Fig. F.1; for details see methods below). Note that an efficiency of 0.02 effective samples per log-likelihood call is half the maximum achievable efficiency, which is 0.04, corresponding 25 log-likelihood evaluations per sample. Samples from the likelihood-constrained prior are generated by evolving a random point for 25 random walk steps (the likelihood condition is checked at each step). If this process were 100% efficient, we would be generating 1 sample per 25 log-likelihood evaluations. Both algorithms struggle with sampling new independent points from the likelihood-restricted prior (LRP) for very large values of N_{live}. The reason could be an intrinsic limitation of the random-walk strategy used to generate new proposals. Alternative strategies are available (see Chapter 2), but our initial experiments revealed that these require quite some tuning on an instance-by-instance basis, therefore making them not suitable for the automatic analysis of large datasets. Although for the specific analysis presented in this chapter we might have fine-tuned an other strategy for the sake of efficiency, we chose to adopt random-walk generated proposals to remain consistent with the sampler settings used in chapters 6 and 7. All considered, the random-walk proposal strategy was the most efficient on average and without fine tuning.

5.4.2 Approximation Errors

Model evidence

Our experiments suggest that the approximations adopted by CNS do not substantially affect the final quality of the estimated model evidence: for $N_{\text{live}} = 1536$, CNS estimates are slightly negatively-biased compared to NS estimates.[3] The bias is significant for both GLMs considered (Fig. 5.2A;

[3] As a matter of fact, Laplace's method, adopted by CNS to compute $p(\mathcal{D}|\boldsymbol{\psi})$, may underestimate the amount of probability mass in the tails of the distribution, resulting in

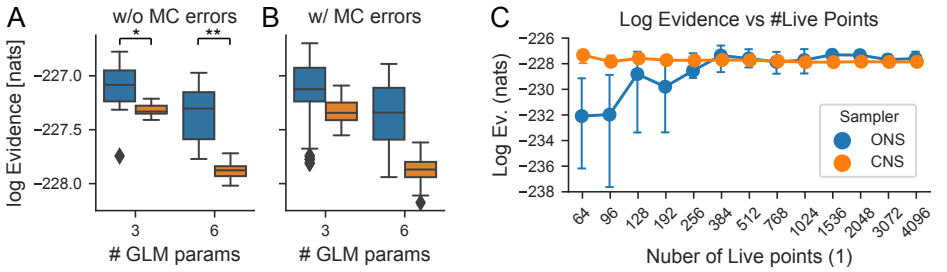

Fig. 5.2: **Errors: Model evidence. A)** Estimated log-evidence of the two GLM models for $N_{\text{live}} = 1536$;**B)** As panel A, but taking into account the MC error intrinsic in NS/CNS integration. **C)** Estimated log-evidence of the quadratic model vs number of live points.

*: $p = 2.70 \cdot 10^{-2}$; **: $p = 6.58 \cdot 10^{-4}$; independent sample t-test with unequal variance, alternative hypothesis $\mathbb{E}[\hat{\mathcal{Z}}_{NS}] > \mathbb{E}[\hat{\mathcal{Z}}_{CNS}]$), but small (less than 1% in both cases), even more so compared to the intrinsic variability of NS estimates: once MC integration errors are taken into account $\mathbb{E}[\mathcal{Z}_{CNS}]$ lies within the 95% confidence intervals of $\mathcal{Z}_{NS}$ (Fig. 5.2B). [4] CNS estimates are also significantly more precise (i.e. have a smaller variance) than their basic NS counterparts (Fig. 5.2A, B; *: $p = 3.31 \cdot 10^{-2}$; **: $p = 1.78 \cdot 10^{-3}$; Levene test for the equality of variances). This is not true only for $N_{\text{live}} = 1536$, but across the whole range of N_{live} values considered: Fig. 5.2C illustrates the variability of the estimated evidence of the quadratic model across all values of N_{live}.

Posterior distribution

The lower variability of the CNS is observed also in the estimated posterior densities. The dissimilarity between estimated posterior distributions was measured in terms of the accuracy of a binary classifier in discriminating between samples belonging to two different estimated posteriors (for details, see description in methods). On average, posteriors obtained through CNS are closer to the reference group than their conventional-NS counterparts across all values of N_{live} (Fig. 5.3A). This effect is invariably observed also when the reference group is composed of CNS estimates(Fig. 5.3A), and it is indeed stronger in this second case. The difference is significant even when measuring the dissimilarity among posterior estimates belonging to the same reference group (Fig. 5.3B). The same result holds also when dissimilarity is measured in terms of Kullback-Leibler divergence (for details, see description in methods). To get a concrete idea of this effect, we shall compare estimated posterior marginals generated by the two methods con-

slightly negatively-biased estimates.

[4] MC errors are estimated using the simulated procedure described by Skilling [104].

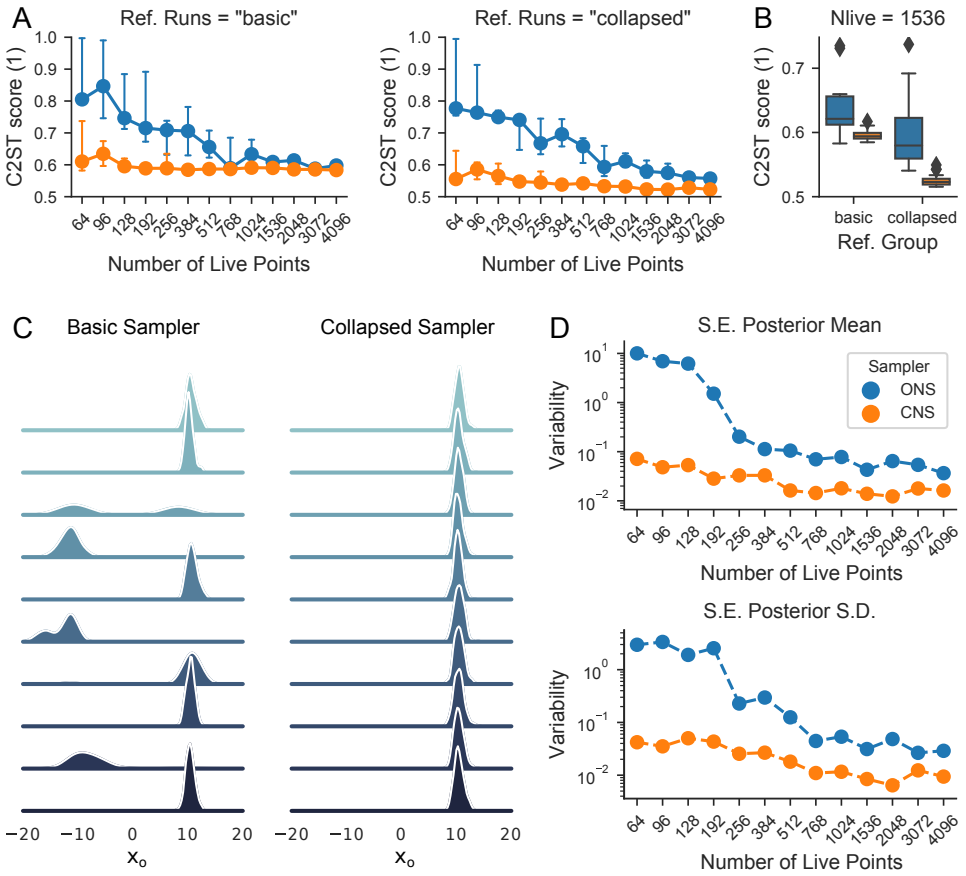

Fig. 5.3: **Errors: Posterior distribution.** **A)** Dissimilarity, measured as the accuracy of a binary classifier (for details see text), between reference posterior distributions and posterior distribution estimated by the basic sampler (blue) or the collapsed sampler (orange) using different values of N_{live}; the reference group consists of posterior distributions estimated using ONS (left) or the CNS (right) for $N_{\text{live}} = 1536$. **B)** Dissimilarity between the reference groups and the output of ONS (blue) and of CNS (orange) for $N_{\text{live}} = 1536$. **C)** Posterior density of the horizontal RF location parameter x_o for 10 NS runs with different random seeds and $N_{\text{live}} = 64$; output of the basic sampler (left columns) and of the collapsed sampler (right column); **D)** Variability of the estimated posterior mean (top) and standard deviation (bottom) as a function of the number of live points used to explore the posterior, measured as the SD of the posterior mean across the different NS runs.

sidered. Fig. 5.3C shows 10 estimated marginals for $p(x_o|\mathcal{D})$ under the linear model, obtained using different random seeds, for $N_{\text{live}} = 64$. Posterior estimates provided by the basic sampler (left) show a large degree of variability, missing the principal posterior mode 50% of the time. Conversely, estimates generated by the collapsed sampler are much more consistent with each other and show considerably smaller intra-variability. To quantify this finding, we summarize each estimated posterior distribution using a finite number of statistics f; we then evaluate the sample in-group standard deviation[5]. This is a measure of the precision (i.e. intrinsic variability) of the estimated statistic. The precision of the inferred posterior mean and standard deviation of the parameter x_o for different values of N_{live} are reported in Fig. 5.3D, top and bottom panel, respectively. A condensed summary of the variability of posterior mean and standard deviation of all RF parameters, for both the linear and the quadratic model, are reported in Fig. F.2. The precision of CNS estimates is significantly higher than the one of the basic NS estimates (Wilcoxon rank-sum and Student's t-test for paired samples; all p-values are reported in Table F.1 and Table F.2, respectively).

5.4.3 Computational costs

The asymptotic computational complexity of CNS scales as $O\big(f(D_1)\big)$, while for ONS it scales as $O\big(f(D)\big)$, for $D_1 < D$ and some monotonically increasing function f. Theoretically, this implies smaller computational costs for CNS than for NS, but provided that computing $p(\mathcal{D}|\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ or $p(\mathcal{D}|\boldsymbol{\theta}_1)$ takes approximately the same amount of time and that there is no different computational overhead (of any kind) in the two scenarios. Results prove that this assumption is not met in this specific case. The amount of time required on average for one log-likelihood evaluation in CNS is approximately 5 times slower than regular NS (Fig. 5.4A, B, C): after all, CNS requires to solve an optimization problem with multiple subroutine calls to evaluate $p(\mathcal{D}|\boldsymbol{\psi})$, while evaluating $p(\mathcal{D}|\boldsymbol{\beta}, \boldsymbol{\psi})$ amount to just one function call. Nevertheless, a word of caution is due here: these performances are computed by dividing the total running time of each NS run by the number of log-likelihood evaluations performed by the nested sampler and not by profiling the two functions in an isolated environment. Although the latter method would be more precise to assess the actual cost of one single log-likelihood call, evaluating the log-likelihood is just one of many steps of the NS algorithm. Other factors are at play, which may contribute a considerable amount of overhead, one above many being the maintenance of an internal compact representations of the pool of live points, so that new proposals can be effi-

[5]Posterior estimates are grouped according to the generating sampler and number of live points used to estimated them; within each group, estimated posteriors were generated using the same number of live points and sampler, but different random seeds; the standard deviation is taken across different random seeds.

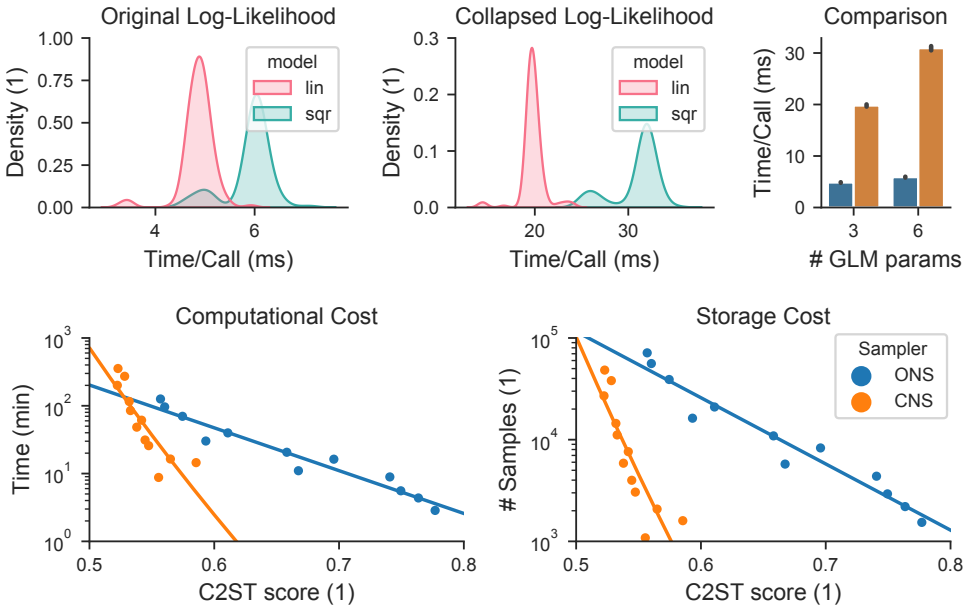

Fig. 5.4: **Computational and memory requirements. A, B)** time required for one single log-likelihood evaluation in ONS (A, $\log p(\mathcal{D}|\boldsymbol{\beta}, \boldsymbol{\psi})$) and in CNS (B, $\log p(\mathcal{D}|\boldsymbol{\psi})$); red, time for the linear model, with 3 nonlinearity parameters; green, time for the full quadratic model, with 6 nonlinearity parameters; computing the CNS log-likelihood is approximately 5 times slower. **C)** Average time per log-likelihood evaluation in ONS (blue) and CNS (orange). **D, E)** Run-time (D) and number of samples to store (E) versus the precision of the posterior estimates.

ciently generated from the likelihood-constrained prior. These subroutines affect the *de facto* running time of a nested sampler implementation beyond the computational costs of evaluating one likelihood function instead of the other.

On the other hand, run times alone may be not an ideal metric to compare the performance of CNS and ONS. A better comparison would be asking how much time is required for the estimated posterior distribution to converge within a desired level of precision. Therefore, although CNS is slower than ONS for any fixed value of N_{live} in our tests, CNS converges faster than ONS to a good approximation of the posterior distribution (Fig. 5.4D).

Furthermore, good approximations of the posterior distribution is represented more compactly by CNS than by ONS, as illustrated in Fig. 5.4E. Here, the average number of generated samples are plotted against the average quality of the resulting posterior distribution, for any given value of N_{live} considered, resulting in smaller sizes of the corresponding result files. This fact may not be a critical factor when analyzing a small dataset consisting only of few cells, but it may become an issue for larger datasets sizes (hundreds of cells) and when multiple models need to be compared for each cell.[6]

This statement is true for also if we measures dissimilarity in terms of KL-divergence, following the procedure described in the methods section below.

5.5 Methods

5.5.1 Synthetic data

The dataset on which all analyses described in this chapter were carried out consists of the simulated response of a visual cortical neuron to a spatio-temporal visual stimulus. The stimulus is a movie consisting of 1800 of spatio-temporally correlated Gaussian noise, sampled at 30Hz, for an equivalent duration of 1 minute. Frame size is 40 by 40 pixels. Neural activity was binned in bins aligned with frame presentations and generated using a slight variation of the generative model described in Chapter 4: the temporal filter was omitted and the principal axes of the 2D Gaussian envelope were align to the grating (i.e., $\psi = \angle\mathbf{k}$). A softplus rectifying non-linearity was used.

[6] As an example to get a rough idea of the memory requirements, the amount of space required to store a NS run on the quadratic model consisting of about 10^3 samples ($N_{\text{live}} = 64$) amounts to approximately 200KB, whereas to store 10^5 samples ($N_{\text{live}} = 4096$) it is approximately 20MB.

5.5.2 Bayesian inference

The model used to generate the synthetic data, and more generally the generative model of visual cortical neurons activity presented in chapter 4, can be used within the framework described in this chapter: it is a GQM where the predictor is the output of a linear filter, here represented in matrix notation:

$$\tilde{\mathbf{S}} = \mathbf{S}\,\mathbf{W}(\boldsymbol{\psi}),$$

where, $\mathbf{S} \in \mathbb{R}$ is the entire stimulus, and $\mathbf{W}(\boldsymbol{\psi})$, is a two-column matrix representing the RF filters; $\boldsymbol{\psi} = \{x_o, y_o, \theta, \kappa, \sigma_x, \sigma_y\}$ is the vector of receptive field parameters. In this context, the basis functions are represented by the columns $\mathbf{W}(\boldsymbol{\psi})$ and the parameters of the non-linearity correspond to the GLM coefficients: as already mentioned, a GQM is linear in its parameters and it is equivalent to GLM on quadratically-transformed inputs. The design matrix is

$$\mathbf{X} = [\mathbf{1},\, \tilde{\mathbf{s}}_1,\, \tilde{\mathbf{s}}_2,\, \tilde{\mathbf{s}}_1 \circ \tilde{\mathbf{s}}_1,\, \tilde{\mathbf{s}}_2 \circ \tilde{\mathbf{s}}_2,\, 2 \cdot (\tilde{\mathbf{s}}_1 \circ \tilde{\mathbf{s}}_2)]^\top, \tag{5.11}$$

where $\mathbf{1}$ is a columns vector of ones, $\tilde{\mathbf{s}}_1$ and $\tilde{\mathbf{s}}_2$ are, respectively, the first and the second columns of $\mathbf{S}$ and "$\circ$" is the Hadamard product, also known as element-wise product. The GLM parameters are $\boldsymbol{\beta} = \{a, b_1, b_2, c_{11}, c_{22}, c_{12}\}$. The ground-truth parameters $\boldsymbol{\psi}^*$ and $\boldsymbol{\beta}^*$ used to generate the data are[7]

$$\boldsymbol{\psi}^* = [10, -10, 0.7, 0.05, 2.5, 2.5] \quad \boldsymbol{\beta} = [0.2, 1.77, 1.77, 0, 0, 0]$$

The values chosen for $\boldsymbol{\beta}^*$ resulted in an average firing rate of about 1Hz and a signal-to-noise ratio of -18dB. An improper prior $p(\boldsymbol{\beta}) = 1$ was used for $\boldsymbol{\beta}$, making the posterior distribution $p(\boldsymbol{\beta}|\boldsymbol{\Theta}, \mathcal{D})$ identical to the likelihood $p(\mathcal{D}|\boldsymbol{\beta}, \boldsymbol{\Theta})$ up to a normalizing constant. Uniform priors were used for all remaining RF parameters, specifically $x_o, y_o \sim \mathcal{U}(-20, 20)$, $\sigma_x, \sigma_y \sim \mathcal{U}(0.5, 5)$, $\kappa = |\mathbf{k}| \sim \mathcal{U}(0, 0.5)$ and $\theta = \angle\mathbf{k} \sim \mathcal{U}(0, 2\pi)$. These priors allocate a considerable amount of mass to physiologically very unlikely RF shapes (e.g. large RF with very large spatial frequency), and may not be a good choice when, e.g., inference is used to detect a RF in sparse and noisy data; however, the focus of the analysis in this chapter is to study how the two alternative implementations of the sampler differ, not how well we can recover ground-truth model parameters.[8] We purposely used wide priors to make it challenging for both samplers to localize the bulk of the posterior distribution.

In addition to the quadratic model, we fitted also a linear depending only on the linear terms, i.e. $\mathbf{X} = [\mathbf{1}, \tilde{\mathbf{s}}_1, \tilde{\mathbf{s}}_2]$ and $\boldsymbol{\beta} = [a, \mathbf{b}]^\top$. We did this to

[7] The units for the elements of $\boldsymbol{\psi}$ are pixels, pixels, radians, cycles/pixels, pixels and pixels, respectively.

[8] In other words, we want to know how well and efficiently we can learn the posterior distribution, not how close the GT parameters are to the posterior MAP or mean; this will be the focus of next chapter and, there, a different prior will indeed be used to encourage the identification of physiologically realistic RFs when the evidence in the data is low.

investigate if and how the dimensionality of the marginalized space affects quality of the estimates.

Task schedule

Multiple values of N_{live} were used to assess the dependence of the algorithm performance on this parameter. The tested values were

$$N_{\text{live}} = [64, 96, 128, 192, 256, 384, 512, 768, 1024, 1536, 2048, 3072, 4096].$$

For each value of N_{live}, ONS and CNS were run 10 times with different random seeds. Tasks were grouped according to their seed in 10 different scripts. All tasks corresponding to one specific seed were executed sequentially by their corresponding script. All scripts were executed simultaneously on a cluster with the following hardware specifications: 2× Intel® Xeon® Scalable Processor "Skylake" 2.40 GHz (for a total of 40 cores), 768GB DDR4 RAM and 8× NVIDIA® GeForce RTX 2080 Ti. This resulted in exactly 10 tasks running simultaneously at all times. We restricted to 4 the number of CPU cores available to each task; tasks were hosted by 2 GPUs; each GPU then hosted at most 5 tasks simultaneously at each point in time.

Data for Fig. F.1

To generate Fig. F.1 we used ONS to fit a 6-dimensional problem. For the prior we used a multivariate Gaussian distribution $p(\theta) = \mathcal{N}(\theta; \mathbf{0}, \mathbf{I})$, where $\mathbf{I}$ is the 6-dimensional identity matrix. The likelihood is a multivariate Gaussian distribution $\mathcal{N}(\mathbf{1}, \mathbf{\Sigma})$, where the mean is a vector of ones; diagonal elements of the covariance matrix are set to 0.01, off diagonal ones to 0.008. This correspond to a standard deviation of 0.1 in each direction and highly-correlated parameters, making the posterior approximately 10 times tighter than the prior, therefore challenging to find. ONS was run using the same exact setting of every other analysis performed in this chapter.

5.5.3 Estimating the quality of the posterior distribution

The quality of each estimated posterior distribution $(\boldsymbol{\theta}|\mathcal{D})$ (s represents the sampler, n the number of live points and i the random seed) is measured by computing its average dissimilarity from a group of posterior estimates posing as reference.[9]Dissimilarity is measured according to some measure $d(\hat{p}^s_{n,i}, p)$ (where p is the target). We take the reference across a set of reference targets to mitigate the effects of random factors. Mathematically, the quality of $\hat{p}^s_{n,i}$ is the following function:

$$Q(\hat{p}^s_{n,i}) = \frac{1}{M^s_{n,i}} \sum_{\hat{p}^*_j \neq \hat{p}^s_{n,i}} d(\hat{p}^s_{n,i}, \hat{p}^{s^*}_{n^*,j}), \tag{5.12}$$

where $\hat{p}^{s^*}_{n^*,j}$ is a reference distribution as , s^* is the reference sampler and $n^* = 1536$; $M^s_{n,i}$ is the number of reference distributions that are not identical to $\hat{p}^s_{n,i}$ (this is done to compute the quality of estimated belonging to the reference group itself, without biasing the result by comparing a reference to itself).

Classifier Two-Samples Test (C2ST)

The first dissimilarity measure is based on how accurately a binary classifier can discriminate samples generated by two different distributions P and Q. This technique goes under the name of Classifier Two-Sample Test (C2ST) [61]. A binary classifier is trained to discriminate between two populations of samples, $x_i \sim P$ and $y_j \sim Q$, then its accuracy on some held-out data is used as a proxy to the dissimilarity between the two generating distributions. [10] The binary classifier consists of a Multi-Layer Perceptron (MLP) [38, 95] with two hidden layers of 60 units each, using a ReLU activation function. I used the `MLPClassifier` implementation provided in the `scikit-learn` Python package [81], version 0.23.2. The classifier was trained using the Adam optimizer [54] for a maximum of 10^4 iterations on 75% of the available data. The remaining 25% is kept for validation. The performance of the trained classifier on the held-out data gives the dissimilarity between the two distributions.

Kullback Liebler Divergence (DKL)

The second dissimilarity measure is based on the symmetric KL divergence [48]

$$d(P, Q) = \frac{1}{2} D_{KL}(P||Q) + \frac{1}{2} D_{KL}(Q||P), \tag{5.13}$$

between the tested probability distribution P and the reference distribution Q, where $D_{KL}(x||y)$ is the asymmetrical KL divergence between the two probability distributions x and y [59, 60]. In order to evaluate each KL divergence, each set of samples was first approximated using a multivariate normal distribution with mean and covariance matching the relative sample

[9] Ideally, we would want to compare each $\hat{p}^s_{n,i}(\boldsymbol{\theta}|\mathcal{D})$ to the true posterior distribution $p(\boldsymbol{\theta}|\mathcal{D})$ by evaluating some dissimilarity measure. However, since ground-truth is not available, we resort to use what in principle should be a high-quality estimate of the posterior distribution. These are the posterior estimates generated with $N_{\text{live}} = 1536$ – as already mentioned, this value of N_{live} corresponds to the largest ESS for both the basic and the collapsed sampler, therefore resulting in the highest resolution estimates of the posterior pdf among the alternatives.

[10] Intuitively, if samples for the two classes are generated by the same underlying posterior distribution (P and Q are identical), the accuracy of the trained classifier on unseen data should be close to chance. Conversely, if the $P \neq Q$, the classifier would learn, in principle, how to discriminate between the two and we would observe a higher classification accuracy.

statistics and then the KL divergence was evaluated analytically. This is admittedly a crude approximation, as the posterior distribution may not necessarily be unimodal and this may be a poor approximation. Since the posterior distribution of the orientation parameter is bimodal by construction, this parameter was ignored and only the marginal posterior distribution of the remaining RF parameters was considered for this analysis.

5.5.4 Implementation

The code for this analysis uses the Nested Sampler implementation provided in the Python package `Dynesty` [107]. The likelihood-conditioned prior distribution is approximated using multiple bounding ellipsoids[26]. New proposal points are generated, conditioned on the multi-elliptical bounds, by evolving a randomly-picked live point for 25 random walk steps [26].

5.6 Summary

In this chapter we have presented a novel strategy to optimize nested sampling runs when some parameters of a model can be marginalize out efficiently. This strategy resembles a technique known as partially-collapsed Gibbs sampling used to improve the convergence and efficiency in a Gibbs sampler [111]. This approach makes it possible to apply NS to an equivalent, lower-dimensional problem, resulting in better convergence properties. We named this algorithm Collapsed Nested Sampling. We showed that CNS can be used to sample from the receptive field model presented in Chapter 4, treating the RF parameters as basis functions parameters and marginalizing the parameters of the non-linearity. We benchmarked both samplers on synthetic data, and showed than CNS is a more efficient sampler than ONS on this particular problem: not only estimates of the model evidence returned by CNS have lower variance, but also the quality of the posterior distributions is superior. We showed that, for each configuration studied, CNS has a smaller Monte-Carlo error and smaller storage requirements than ONS for any desired level of accuracy, making it the sampler of choice for computationally-efficient inference.

We observed that both algorithms struggle when sampling new independent points from the likelihood-restricted prior (LRP) for very large values of N_{live}, and the reason could be an intrinsic limitation of the random-walk strategy used to generate new proposals. As mentioned in Chapter 2.3, several strategies have been developed to sample from the LRP. One alternative would be rejection sampling from a multi-ellipsoid approximation of the current set of live points [26]. This strategy requires an additional algorithm parameter to control how tightly the ellipsoids cover the set of live points. After some initial experiments in this directions, we realized that this additional parameter needs quite some careful fine tuning on each separate

instance of the problem (i.e. for each simulation or actual neural recording). While we could have undergone the tuning procedure for the analysis presented in this chapter, doing so on a recording-by-recording basis on a large dataset would be unpractical. Furthermore, we observed that the random-walk strategy was on average a more efficient alternative on large datasets, and it required no tuning. For thes reasons, in order to make the results presented here relatable to work presented in following chapters we adopted here the same LRP-sampling strategy.

CNS can be applied to any GLM using parametric basis functions. For example, it can sample the non-linearity and scale parameters of a family of raised cosine basis function used to model the temporal profile of a history filter [84]. In principle, it can handle also integer or discrete parameters, such as the number of basis functions itself, for example when modeling a receptive field as a liner combination of splines [44]. As a matter of fact, for NS the continuous or discrete nature of the parameter space does not matter, as long as a prior distribution can be defined and new live points can be sampled from the likelihood-restricted prior. This fact makes NS, and by extension CNS, a good sampling strategy in problems where a traditional, potentially gradient-based MCMC approach could not be applied, opening to the possibility of sampling the posterior distributions over parameter classes that are usually not handled in a Bayesian framework.

Chapter 6

Receptive field identification on synthetic data

6.1 Introduction

In Chapter 4 and Chapter 5 we presented a compact generative model and an efficient algorithm to perform Bayesian inference on its parameters. In this chapter we will investigate the how reliably we can expect to detect a receptive field and identify its parameters. We will test the detection and identification tasks on synthetically generated data covering a wide range of firing rates and noise levels compatible with the physiological levels observed in real neural population (Fig. 6.1). More specifically, we will investigate the reliability of a Bayesian classifier to detect the presence of a receptive field as a function of the amount of available data, here measured as number of observed spikes, and the noisiness of the neuron. We will also quantify how these same quantities affect the identification of model parameters.

6.2 Receptive Field detection

To detect the presence of a RF in the data, we built a Bayesian classifier comparing the evidence of the RF model and the evidence of a null model M_0 predicting random Poisson spikes at constant rate. The Bayesian classifier performs its classification based on the Bayes factor

$$BF_{RF,0} = \frac{p(\mathcal{D}|\mathrm{RF})}{p(\mathcal{D}|M_0)},$$

where $\mathcal{D}$ is the data and $p(\mathcal{D}|\mathrm{RF})$ and $p(\mathcal{D}|M_0)$ are, respectively, the marginal likelihoods of the data under the RF model and under the null model (for details, see sec. 6.5.2 and 2). We are specifically interested in the ability of the classifier to detect an existing receptive field (i.e. the *sensitivity*), in the probability of missing an existing receptive field (i.e. the *miss rate*), and

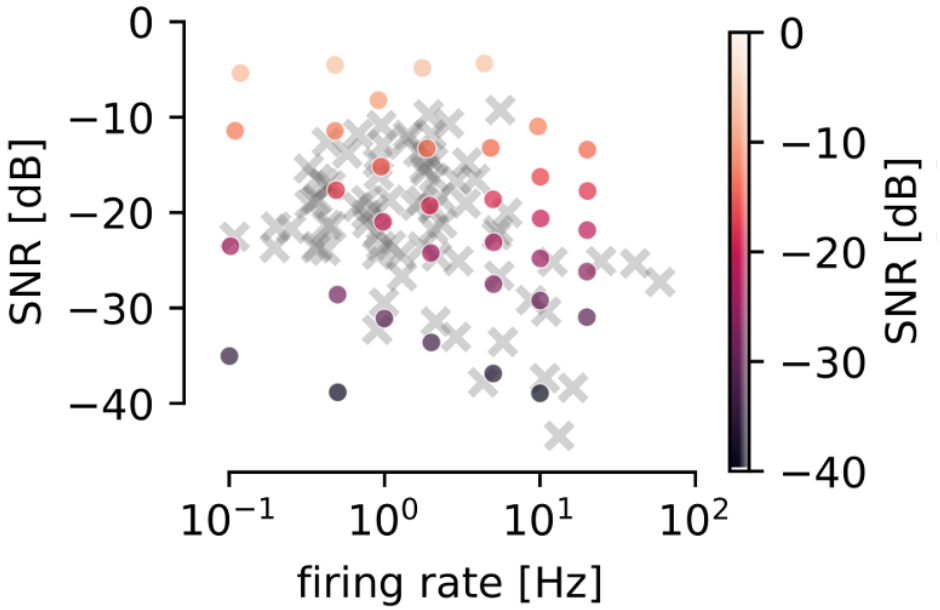

Fig. 6.1: **Simulated data.** Each colored point represents a simulation in our synthetic dataset. Crosses represent actual neurons in the electrophysiological dataset.

	Test result		
Rec. Field	Detected	Rejected	Undefined
No	2.6%	0.0%	97.4%
Yes	71.2%	19.5%	9.4%

Table 6.1: **Summary performance of the Bayesian classifier.** Highlighted are sensitivity (green), miss rate (blue) and fall-out (red). For details about how these indexes are computed, see text.

in the chances of wrongly detecting the presence of a receptive field in pure noise (i.e. the *fall-out*). For definition of these indexes, see section 6.5.4.

Overall, the Bayesian classifier correctly detects 71.2% of the receptive fields in the dataset. Chances of dismissing a receptive field as noise are instead 19.5%; the probability of detecting presence of a RF in pure noise in 2.6% (the detailed performances of this classifier are reported in Table 6.1). The high miss rate can be explained by the relative abundance of trials with very few spikes in this dataset: 4 out of 7 trials are short, 60s trials (see Table 6.5 and the methods section for details). If we correct for the over representation of short trials with almost no spikes and we consider only segments of data with at least 10 spikes, the situation significantly improves: the sensitivity is now 84.7% and the miss rate is 8.6%. What we just reported are marginal indexes. We take can also into account the amount of data available to the classifier, measured in terms of spike counts, and the intrinsic noisiness of the neuron (its SNR; see Appendix D): the detection performance steadily increases for higher spike counts and higher SNR, which signal a strong stimulus drive (Fig.6.2A). Similarly, the probability of missing an existing receptive field decreases accordingly (Fig.6.2B).

Wrongly detecting a receptive field in noise is an unlikely event. However, it is interesting that the Bayesian classifier is seemingly unable to definitively rule out the presence of a RF in a pure noise: the vast majority of these simulations are left unclassified, with different degrees of mild evidence in favor or against the receptive field model (Fig. 6.3A; see also Table 6.1). One would instead expect that all the additional parameters of a RF model would not find enough support in random data, when instead a more parsimonious explanation should be preferred. Inspecting the actual posterior distributions is very insightful, revealing that for all parameters except the spatial frequency κ, the marginal posterior distributions are indistinguishable from the priors; the posterior marginal for κ, instead, is concentrated at a-priori-very-unlikely high values (see supplementary Fig. F.4). A receptive fields with these properties would integrate a large patch of the the stimulus, weighting nearby locations with fast-alternating sign. Since the stimulus is spatially correlated, such an integration effectively cancels out

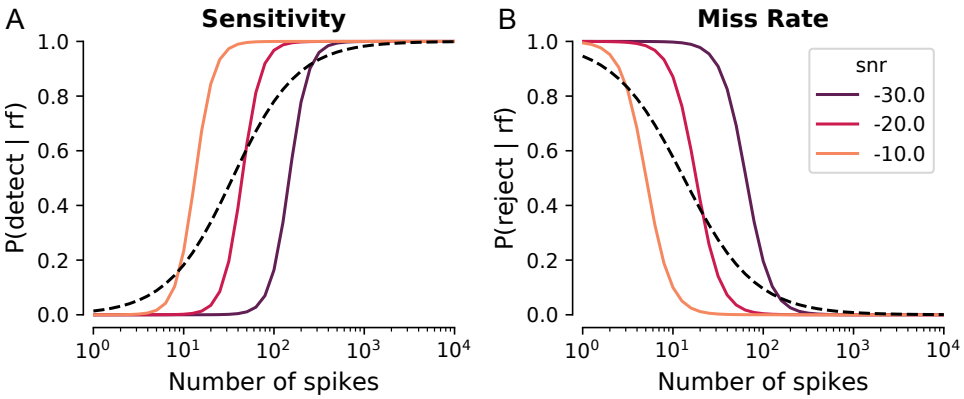

Fig. 6.2: **Detection perfomance.** Sensitivity **(A)** and Miss Rate **(B)** of the Bayesian classifier. Dashed lines show how the index changes as a function of number of spikes only. Solid lines take into account also the SNR of the neuron. **N.B.** Since the Bayesian classifier is not a binary classifier, the equivalence "sensitivity" $= 1 -$ "miss rate" does not hold. The two indexes are independent, therefore reporting both is not redundant.

the stimulus contribution. This outcome is apparently independent of the number of spikes in the simulation.

We trained also a binary classifier based on predictive performance on held-out data, quantified in terms of cross-validated log-likelihood (CVLL; for details, see the method section). The CVLL classifier does not suffer from this problem: indeed, for all the tested scenarios, no receptive field was ever wrongly detected in noise trials, suggesting a virtually optimal performance (Fig. 6.3B).

6.3 Identification of model parameters

We will now focus on the issue concerning the identification of the receptive field parameters, a problem that can be summarized by the question: how does our uncertainty around the model parameters decrease as a function of available data?

6.3.1 Characterizing the orientation

Let us start with an example: suppose that we want to identify the orientation of the receptive field within a level of uncertainty that we deem acceptable (Fig. 6.4A). How much data would we need for this task? We considered the posterior distribution of model parameters for each simulation in the dataset and every subset thereof (for details about data partitioning, see the methods section). For each fit, we computed the posterior root-mean-squared error (RMSE) from ground-truth according to eq. (6.4),

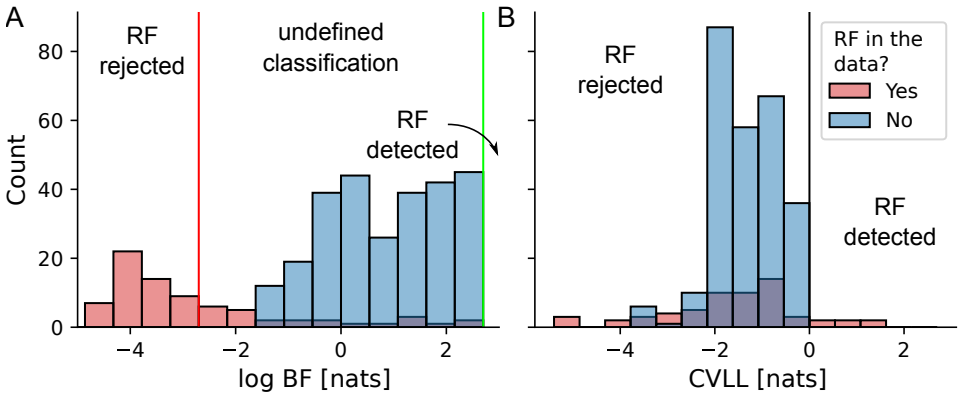

Fig. 6.3: **Classification results.** Test scores for simulations and segments where a receptive field was not detected by the Bayesian classifier. Noise trials are shown in blue, while actual RF trials are in red. **A)** Bayesian classifier. The vertical red line is the threshold for rejection, based on substantial evidence in favor of the null model. The green line is the threshold for detection. **B)** CVLL classifier. The vertical line is the threshold for the binary classification.

summarizes both bias and variance of the estimate (details in the next session). The RMSE steadily decrease as a function of observed spike counts, and is as low as 10° already with a few tens of spikes. This parameter is typically assessed by probing the cell with static or moving gratings with different orientations. Due to a trade-off between accuracy and experimental time, 8 to 12 equally-spaced different orientations are typically tested, giving a resolution between 30 to 45 degrees, represented here by the shaded area. With less than 100 spikes (blue shaded line), we can identify the RF orientation with a precision approximately 10 times higher than in a typical study.

6.3.2 Quantification of estimation quality

Intuitively, the more the available data, the more precise the estimates should be. *Residual uncertainty* (RU), computed as the ratio of the entropy of the posterior to the entropy of the prior (see section 6.4.5 for details), provides a quantitative measure of this trend. RU steadily decreases for higher spike counts (Fig. 6.4B). Furthermore, comparably less spikes are needed to achieve a given level of precision for reliably-firing cells (higher SNR) than for more noisy ones. RU can also be computed on a per-parameter basis, making it possible to investigate which parameters are more or less constrained by some given amount of data (Fig. 6.4C; for further details, see supplementary Fig. F.6). Some parameters, like e.g. position and 2D spatial frequency ($\mathbf{x}_0$ and $\mathbf{k}$), can be learned more efficiently than others,

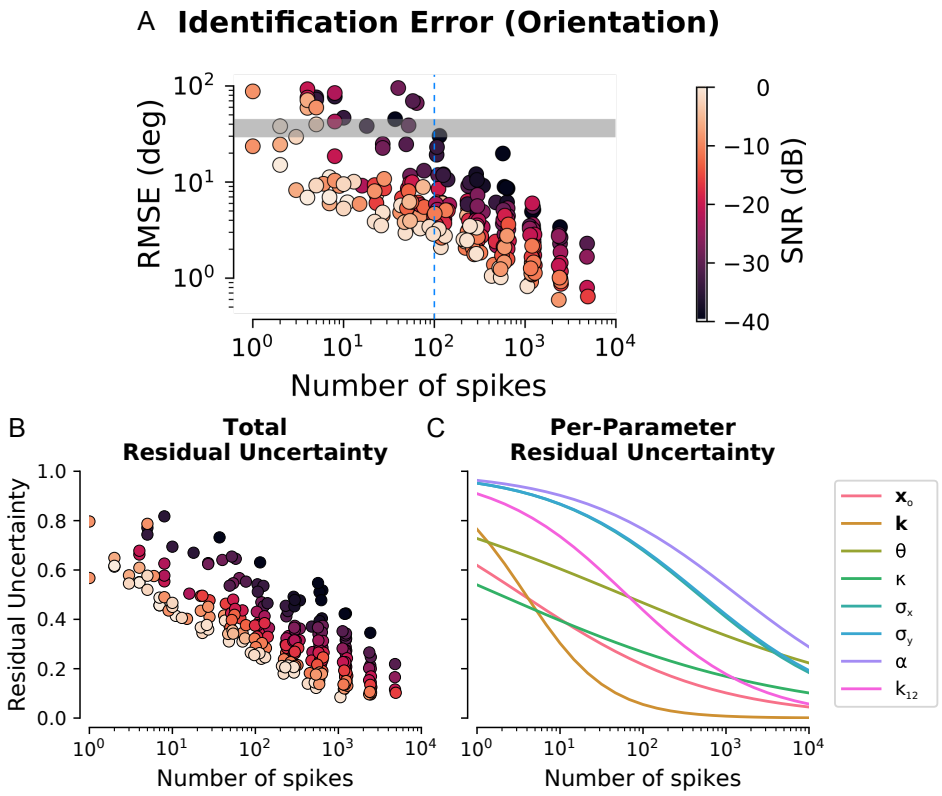

Fig. 6.4: **Identification of the model parameters. A)** Overall identification error for the orientation parameter, measured in terms of posterior RMSE, as a function of spike counts and SNR. The shaded area represents the typical resolution for the orientation parameters (30° – 45°). The blue line denotes 100 spikes. **B)** Total RU as a function of observed spike counts. **C)** Per-parameter RU as a function of observed spike counts.

such as the size or the scale parameter of the temporal filter (σ_x, σ_y and α, respectively). For the 2D spatial frequency $\mathbf{k}$, RU decreases faster than for either the orientation parameter θ or the polar frequency κ. Although at first surprising, this result is well explainable in terms of basic properties of the differential entropy: in general, the joint entropy of two random variables X and Y is bounded above by the sum of the entropy of the respective marginals:

$$H[X, Y] \leq H[X] + H[Y],$$

with equality if and only if X and Y are independent. In this case, θ and κ are clearly not independent, as revealed by their joint posterior distribution (see supplementary Fig. F.5).

RU provides a useful but at the same time a very abstract quantitative insight about the how the quality of the estimates is shaped by the amount

	Residual Uncertainty		Spike Count	
	ρ^2	p-value	ρ^2	p-value
param.				
$\mathbf{x}_o$	0.93	4.85e-142	0.51	2.14e-39
$\mathbf{k}$	0.90	1.60e-124	0.49	1.14e-37
σ_x	0.89	2.33e-117	0.47	7.48e-36
σ_y	0.87	2.79e-108	0.47	4.28e-35
α	0.60	2.14e-50	0.32	1.83e-22
k_{12}	0.84	8.12e-98	0.44	1.10e-32

Table 6.2: **RMSE vs RU and spike counts.** Correlations between RMSE and residual uncertainty (left) and between RMSE and spike count (right) are measured in terms of Spearman's ρ correlation coefficient. As we are interested only in the amount of correlation and not on its sign, we report here its square ρ^2.

of data: it does not tell us directly how tightly around GT the posterior distribution is concentrated. Thanks to our knowledge of ground-truth model parameters, we can investigate address this question by actually inspecting the posterior distributions. We measure estimation errors as RMSE from ground truth (definition in sec. 6.4.6). This error metric has the advantage of being expressed in the same units of the parameters, therefore it provides an easily interpretable quantification of the actual magnitude of the errors. The same qualitatively dependency on spike count and SNR is also found for the RMSE (see supllementary Fig. F.7): RMSE decreases with increasing spike counts and is smaller for higher SNR. We find that, overall, RMSE is better predicted by RU than it is by spike count (see supplementary Fig. F.8; Table 6.3.2, Kendall's tau). AS RU can easily be evaluated after the posterior distribution has been learned via nested sampling, we can use its value as a reliable proxy for the underlying RMSE. This information can be used to assess the amount of data collected was sufficient to achieve a desired error level.

We can further decompose the estimation error into bias and variance according to

$$\text{MSE} = \text{BIAS} + \text{VARIANCE},$$

in order to empirically quantify the *accuracy* and the *precision*[1] of the estimate. Here, MSE indicates the *mean square error*, which is of course the square of the RMSE. We found that for all parameters except the temporal filter's scale parameter α, the variance of the posterior distribution contributed significantly more than bias to the total estimation error (see

[1] Accuracy refers how close the average estimate is to ground truth; precision refers to how close measurements are to each other.

param.	average	s.e.m.	z-score	p-value
$\mathbf{x}_o$	0.620	0.014	8.73	$1.26 \cdot 10^{-18}$
$\mathbf{k}$	0.652	0.016	9.64	$2.59 \cdot 10^{-22}$
κ	0.665	0.016	10.23	$7.48 \cdot 10^{-25}$
σ_x	0.689	0.014	13.11	$1.48 \cdot 10^{-39}$
σ_y	0.685	0.014	12.84	$5.05 \cdot 10^{-38}$
α	0.493	0.019	0.36	$3.60 \cdot 10^{-01}$
k_{12}	0.586	0.016	5.30	$5.78 \cdot 10^{-08}$

Table 6.3: **Variance vs bias.** Fractional amount of the Mean Squared Error accounted for by variance (s.e.m.: Standard Error of the Mean).

Table 6.3); for α the difference was not significant. This is a desirable result: since in a real-world application ground-truth knowledge would not be available, it is reassuring to know that bias is not a main source of error in our estimates.

6.4 Simulation details & inference

6.4.1 Generative model

Neural activity was simulated using the generative model presented in Chapter 4 using the receptive-field model parameters reported in Table 6.4. We simulated a purely linear, direction-selective cell ($k_{\text{dir}} = 1$, $\mathbf{C} = \mathbf{0}$) for different values of the parameters a and $\mathbf{b}$ (its magnitude), resulting is the distribution of average frequency and SNR reported in Fig. 6.1 (for details on how to compute the SNR of a Poisson spiking neuron, see Appendix D).

6.4.2 Stimulus and simulation partitioning

In order to minimize potential discrepancies between the simulations and a target real-world application presented in the next chapter, we used the the same stimulus movie that was used in the electro-physiological recordings discussed in Appendix E. This resulted in 5 minutes of simulated activity for each tested condition. The last minute of the simulation was held-out for cross-validation. The remaining 4 minutes were split in chunks of 1 and 2 minutes, resulting in 7 data segments for each simulated recording (partitioning details are reported in Table 6.5).

Param.	Units	Value	Prior
x_o	deg	20	$\mathcal{U}(x_{\min}, x_{\max})$
y_o	deg	20	$\mathcal{U}(y_{\min}, y_{\max})$
θ	deg	30	$\mathcal{U}(0, 2\pi)$
κ	cycles/deg	0.05	$\kappa/\kappa_{\max} \sim \text{Beta}(1.5, 8)$
σ_x	deg	3.75	$4\sigma_x \sim \mathcal{LN}(2.9, 0.28)$
σ_y	deg	3.75	$4\sigma_y \sim \mathcal{LN}(2.9, 0.28)$
α	s^{-1}	60.0	$(\alpha - 40)/80 \sim \text{Beta}(3, 6)$
k_{12}	–	0.8	$\mathcal{U}(0, 1)$
a	–	varying	$\mathcal{N}(0, 25)$
$\mathbf{b}$	–	varying	$\mathcal{N}(\mathbf{0}, 25 \cdot \mathbf{I}_2)$

Table 6.4: **Ground truth and priors.**. Values used to generate the simulated data. $\mathcal{U}(\cdot,\cdot)$, $\mathcal{N}(\cdot,\cdot)$ and $\mathcal{LN}(\cdot,\cdot)$ are the Uniform, the Normal and the Log-Normal distributions, respectively. $\mathbf{I}_2$ is the 2×2 identity matrix.

Block	Duration	t_{start} [s]	t_{end} [s]
1.1	1 min	0	60
1.2	1 min	60	120
1.3	1 min	120	180
1.4	1 min	180	240
2.1	2 min	0	120
2.2	2 min	120	240
4.1	4 min	0	240

Table 6.5: **Details on data partitioning.** Beginning and end of each simulation segment.

6.4.3 Noise simulations

Noise trials were constructed by randomly shuffling the spikes of the receptive-field simulations and then partitioning the result spike train according to Table 6.5.

6.4.4 Sampler settings

The posterior distribution were sampled using to the algorithm discussed in Chapter 5. We used a random-walk proposal with 25 steps and we set $N_{\text{live}} = 512$.

6.4.5 Residual uncertainty

The dimensionless *residual uncertainty* RU of (a subset of) model parameters $\boldsymbol{\Theta}'$ is computed as

$$\mathrm{RU} = 1 - \frac{H\big[p(\boldsymbol{\Theta}'|\mathcal{D})\big]}{H\big[p(\boldsymbol{\Theta}')\big]}, \tag{6.1}$$

where $H[p]$ is the entropy of the distribution p, and $\boldsymbol{\theta}' \subseteq \boldsymbol{\theta}$. RU = 1 means that the posterior is indistinguishable from the prior. Conversely, RU = 0 means that the posterior has collapsed onto a delta peak, i.e. we assign with absolute confidence a certain value to the model parameters. Each of the 1D and 2D marginals considered must first be discretized into a probability histogram, and then its entropy is computed according to:

$$H[\psi] = -\sum_{n=1}^{N} p_n \log_2 p_n, \tag{6.2}$$

where N is the number of bins of the histogram and p_n is the probability mass associated with the n-th bin. The marginal posterior distributions are discretized as follows: for the receptive field center $\mathbf{x}_o$ is discretized on a grid of 120×68 bins along the horizontal and vertical direction respectively, corresponding to a resolution of 1 pixel; the spatial frequency κ onto 34 bins in the range $[0, 0.34]$; the orientation θ onto 360 bins in $[0, 2\pi]$ corresponding to a resolution of 1 degree; the 2D spatial frequency $\mathbf{k} = [\kappa\cos\theta, \kappa\sin\theta]^\top$ into 64×64 bins in the range $[-0.34, 0.34]^2$; width and height of the receptive field, respectively w and h, onto 30 bins in the range $[0, 30]$; the temporal filter rate α into 80 bins in the range $[40, 120]$; the temporal filter shape parameter k_{12} into 20 bins in the range $[0, 1]$.

6.4.6 Estimation Error

RMSE from ground truth for a parameter ξ is defined as

$$\mathrm{RMSE}(\xi) = \sqrt{\mathbb{E}_{\xi|\mathcal{D}}\|\xi - \xi^*\|^2} \approx \sqrt{\sum_{i=1}^{N} w_i \|\xi^{(i)} - \xi^*\|^2}, \tag{6.3}$$

where ξ^* is the ground-truth value, $\xi^{(i)}$ is a weighted sample from the marginal posterior distribution $p(\xi|\mathcal{D})$ and w_i is the corresponding weight. A special treatment is reserved to the orientation parameter θ, because it represents a circular variable. In this case, we compute the cosine distance, instead of the magnitude of the difference:

$$\mathrm{RMSE}(\theta) = \frac{180}{\pi} \arccos\Big(\mathbb{E}_{\theta|\mathcal{D}}[\cos(\theta - \theta^*)]\Big). \tag{6.4}$$

The multiplicative factor converts from radiants to degrees, for a more easily interpretable result.

6.5 Data classification

6.5.1 Control model

Concerning the data classification task, the control M_0 is provided the constant firing-rate Poisson neuron

$$\Pr(Y_i = y) = \frac{{\lambda_0}^y}{y!} \exp(-\lambda_0), \tag{6.5}$$

where $\lambda > 0$ is the firing rate parameter. Under the assumption that spike counts in different bins are i.i.d. according to eq. (6.5), the log-likelihood of an entire spike train $\mathbf{r}$ is

$$\ln p(\mathbf{r}|\lambda_0) = S \ln \lambda_0 - N\lambda_0 - \sum_{i=1}^{N} \ln y_i!, \tag{6.6}$$

where $S = \sum_{i=1}^{N} y_i$ is the sufficient statistic for this model. The maximum likelihood solution for λ_0 is the empirical mean $\hat{\lambda_0} = S/N$. Assuming a uninformative prior $p(\lambda_0) = \lambda_0^{-1}$, we obtain the following posterior distribution and marginal likelihood:

$$p(\lambda_0|\mathbf{r}) = \text{Gamma}(S, N) \tag{6.7}$$

$$p(\mathbf{r}|M_0) = \frac{\Gamma(S)}{N^S \prod_{i=1}^{N} y_i!} \tag{6.8}$$

6.5.2 Classification based on Bayes factors

The Bayesian classifier perform its decisions based on Bayes factors (see Chapter 2.2.1). In the context of this chapter, the control model $\mathcal{M}_0$ is the constant firing rate model, and the alternative $\mathcal{M}_1$ is the receptive field model. For numerical convenience, the logarithm of the Bayes factor was used. The classifier detects the presence of a receptive field in the data if strong evidence is found in favor of $\mathcal{M}_1$. Conversely, if strong evidence is found in favor of the control model, the presence of the receptive field is excluded. If there is no strong evidence in favor of either model, the data segment is left unclassified. The decision was based on whether the BF would exceed a certain threshold ϑ. Multiple candidate values were tested on a grid ranging from 0 to 5 with a sampling step of 0.1. The optimal cutoff that minimizes maximizing the probability of correct detection while simultaneously minimizing the occurrence of false positives was found at $\ln \mathrm{K} = 2.7$ nats, which is consistent with what is considered a sign of strong evidence in favor of the alternative model [48, 53].

6.5.3 CVLL classifier

This classifier bases its decision on comparing prediction performances on some held-out data $\{\mathbf{r}_{\text{test}}, \mathbf{s}_{\text{test}}\}$, according to

$$R = \mathbb{E}_{p(\boldsymbol{\theta}_{RF}|\mathcal{D})}\left[\ell(\boldsymbol{\theta}_{RF})\right] - \ell_0(\hat{\lambda_0}), \tag{6.9}$$

where $\ell(\boldsymbol{\theta}_{RF}) = \log p(\mathbf{r}_{\text{test}}|\boldsymbol{\theta}_{RF}, \mathbf{s}_{\text{test}})$ is the predicted log-likelihood of the receptive field model on the held-out data, $\ell_0(\hat{\lambda}) = \ln p(\mathbf{r}_{\text{test}}|\hat{\lambda}_0)$ the one of the null model, according to eq. (6.6) and $\hat{\lambda}_0$ is the maximum likelihood solution for λ_0 on the training set. The expectation is taken with respect to the posterior distribution of the RF model parameters given the training data and it is needed to take into account our uncertainty about the model parameters. The classification rule is very simple: the presence of a receptive field is detected if $R > 0$; conversely, it is rejected if $R < 0$. Considering that $\ell(\boldsymbol{\theta}_{RF})$ is itself a random variable, a more rigorous Bayesian treatment taking into account not only its mean, but its overall probability distribution would make some form of soft labeling possible.

6.5.4 Classification performances

The *selectivity* of a classifier refers to its ability to correctly reject negative instances. Mathematically it is defined as the ratio between the number of rejected negatives (True Negatives, TN) and the total number of negative cases (N), hence also its alternative name "True Negative Rate" (TNR):

$$\text{Sensitivity} = \frac{\text{Number of detected positives}}{\text{Total number of positives}}$$

The *sensitivity* of a classifier is its ability to correctly detect positive instances. Mathematically it is defined as the ratio between the number of detected positives (True Positives, TP) and the total number of positive cases (P), hence also its alternative name "True Positive Rate" (TPR):

$$\text{Sensitivity} = \frac{\text{Number of detected positives}}{\text{Total number of positives}}$$

Its complement is the *miss rate*, which measures how many positive instances are dismissed as negatives. It is defines as the ratio between the number of false negatives (FN) and the total number of positives, therefore it also known under the name "False Negative Rate" (FNR):

$$\text{Miss rate} = \frac{\text{Number of false negatives}}{\text{Total number of positives}}$$

Finally, the *fall-out* measures how likely it is to classify a negative instances as positive. Is is computed as the False Positive Rate, the ratio of wrongly

detected negative instances and the total number of positive instances:

$$\text{Fall-out} = \frac{\text{Number of false positives}}{\text{Total number of negatives}}$$

Fall-out and miss rate provide a quantification of the relative occurrence of type I and type II errors, respectively [23].

NB: For a binary classifier, TPR $= 1 -$ FNR and TNR $= 1 -$ FPR is always true; this is however not the case for a 3-way classifier, like the Bayesian classifier discussed in this chapter. Talking about TPR and FNR as well TNR and FPR as two separate entities actually makes sense in this context.

6.6 Summary and Discussion

In this chapter we addressed the detection of a receptive field and of the identification of its parameters. We showed that a Bayesian classifier can reliably detect the presence of a receptive field from a few tens of spikes. The absence of a receptive field is instead better asserted by a classier based on cross-validate log-likelihoods on held-out data. Both classifiers have their perks and disadvantages but it seems that their properties complement each other. Pure noise trials pose a challenge to the Bayesian classifier, which appears to be unable to distinguish between a very noisy neuron and no RF at all, whereas the CVLL classifier shows virtually perfect performance. On the other hand, on scenarios where data is really scarce, the Bayesian classifier makes use of all the available data, whereas the CVLL classifier must reserve some data for testing. Yet, we showed that validating on held-out data strongly penalizes any kind of over-fitting. The development of a composite classifier built upon the Bayesian and the CVLL classifiers constitutes a promising line of research for possible future work.

We also showed how the quality of estimated model parameters improves with the amount of observed data and provided an interpretable quantification of this trend. The empirical finding that, on average, variance accounts for the largest portion of the identification error is an encouraging starting point for a further, more rigorous investigation of consistency properties of the algorithm presented in this Chapter 5.

Chapter 7

Analysis of an electrophysiological dataset

In this chapter, we will illustrate the power of a fully Bayesian approach to the problem of receptive field identification on a real-world dataset, consisting of the activity of 80 neurons acquired by means of electrophysiological recordings in rat primary visual cortex. The analysis presented in this chapter will unfold as follows: first, we will tackle the problem of receptive field detection, to establish how many cells in the dataset are responsive to visual stimulation; we will then illustrate a detailed analysis of a single cell, showing how access to the complete posterior distributions allows us to test different hypotheses and associate a given, rigorously estimated, confidence level to different assertions about the data; we will then report the results of our analysis on the entire dataset.

7.1 Detection and model identification

7.1.1 Properties of the dataset

ù For the analysis described in this chapter, we considered only the data from 80 well isolated cells, acquired during binocular visual stimulation (for details see Methods and Appendix E) The average firing rate (FR) in this dataset is 3.91 ± 8.56 Hz (mean $\pm$ sd). The FR was computed using the number of observed spikes during the entire duration of each 4 minutes segment of data. The distribution of measured FR is very skewed (Fig. 7.1A) and it spans two orders of magnitude between a minimum of 0.05Hz to a maximum of 61.37Hz. Half of the cells fire very sparsely (median FR: 1.43Hz). The distributions of observed spike counts for segments consisting of 1, 2, and 4 minutes of data are illustrated in Fig. 7.1B, and detailed information is reported in Table 7.1.

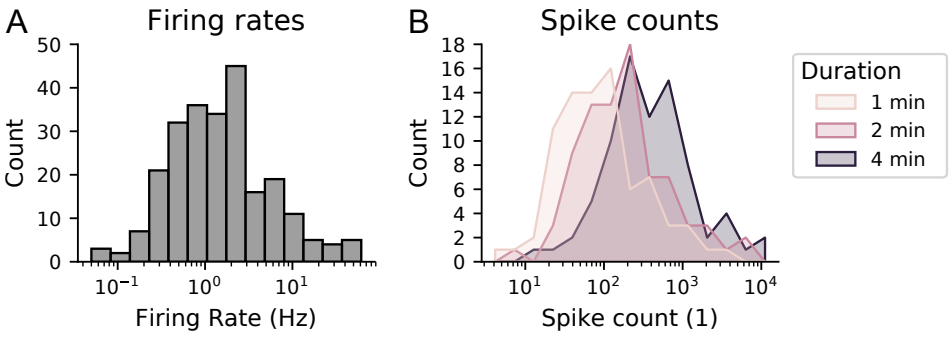

Fig. 7.1: **Distribution of firing rates and spike counts. A)** Distribution of observed firing rates. **B)** Distributions of observed spike counts within segments consisting of 1, 2, or 4 minutes of data.

Duration	min.	2.5%	25%	50%	75%	97.5%	max.
1 min	3	1	3	76	171	1389	3682
2 min	6	24	67	179	347	2913	7170
4 min	15	47	162	361	703	6033	14215

Table 7.1: **Distribution of spike counts.** The table reports the observed spike counts corresponding to different quantiles, together with the minimum and maximum observed values. Each row summarizes the data corresponding to different segment lengths.

7.1.2 Models used for the analysis

We will compare the predictive properties of four different configurations of the generative model presented in Chapter 4, which will allow us to test different hypothesis about the process underlying the generation of the spikes. These four models are: LO, separable RF and linear response; LD, non separable RF and linear response; QO, separable RF and quadratic response; QD, non separable RF and quadratic response. We denote the set of all models as $\mathcal{M} = \{\text{LO}, \text{LD}, \text{QO}, \text{QD}\}$. A summary of the properties defining these models is available in Table 7.2. With these four configurations, we aim to characterize the selectivity properties and the linearity of the cells in this dataset. In particular, we include the purely linear configurations to assess whether the quadratic terms substantially contribute to the quality of the firing rate prediction.

Null model For any given cell, the null model m_0 models all observed spike counts y_t (for $t = 1, \ldots, T$) as i.i.d. random Poisson variables with constant rate $\lambda_0 > 0$:

$$y_t|\lambda_0 \sim \text{Poisson}(\lambda_0), \quad t = 1, \ldots, T; \tag{7.1}$$

Assuming an uninformative prior $p(\lambda_0) = 1/\lambda_0$, the evidence of the null model for data $D = \{y_1, \ldots, y_T\}$ can be computed analytically:

$$p(\mathcal{D}|m_0) = \frac{\Gamma(S)T^{-S}}{\prod_{t=1}^{T} y_t!}, \tag{7.2}$$

where $S = \sum_{t=1}^{T} y_t$ is the total spike count and $\Gamma(\cdot)$ is the gamma function.

ID	Separable	Linear	k_{dir}	Nonlinearity
LO	✓	✓	0	$\mathbf{C} = \mathbf{0}$
LD		✓	1	$\mathbf{C} = \mathbf{0}$
QO	✓		0	
QD			1	

Table 7.2: **Model configurations.** Settings and constraints of the four model configurations considered in this chapter. "Ori-Sel" and "Dir-Sel" stand for "orientation selective" and "direction selective", respectively. The purely linear models are obtained from the original quadratic models by enforcing $\mathbf{C} = \mathbf{0}$.

7.1.3 Receptive Field identification

The first question of our extensive analysis concerns the number of cells in this dataset that are actually responsive to visual stimulation. In other

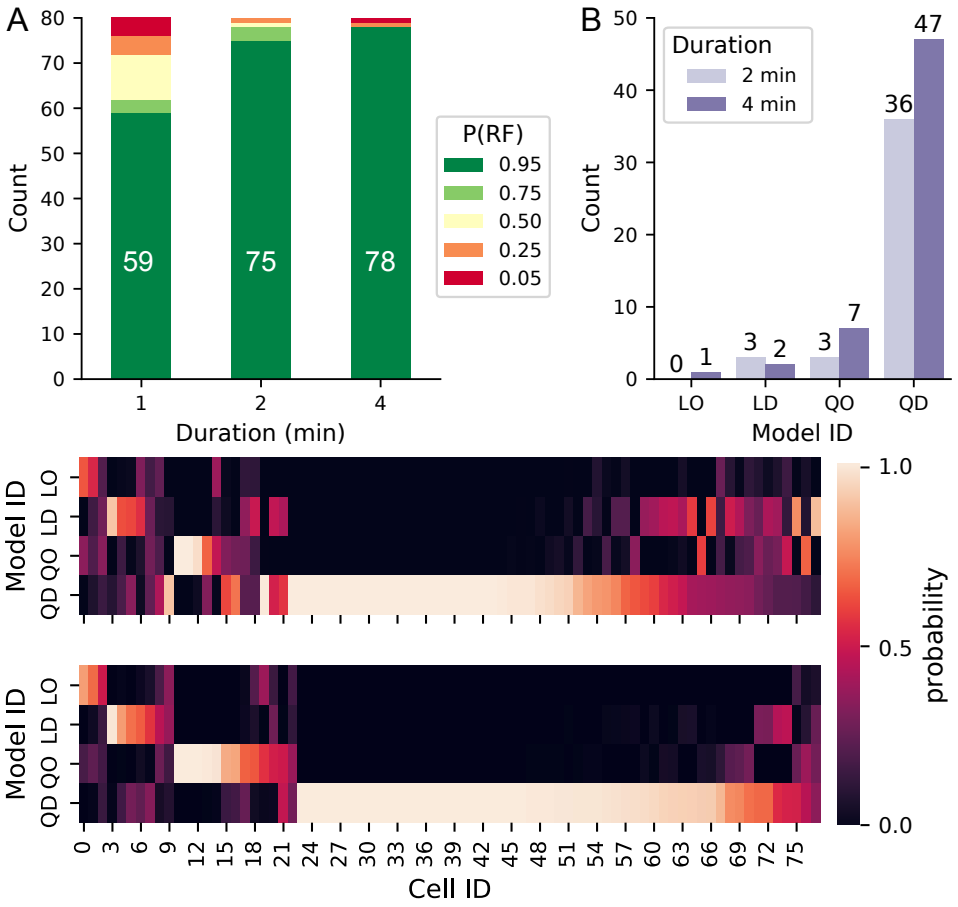

Fig. 7.2: **Receptive field detection. A)** Number of identified receptive field, as a function of amount of data used for inference. **B)** Number of cells assigned to each model class with at least moderate confidence ($P > 75\%$). **C)** Detailed model posterior probability from 2 minutes of data. **D)** Detailed model posterior probability from 4 minutes of data. The QD model is the most represented in this dataset.

words, how many RFs can we detect? To answer this question, we score each model $m \in \mathcal{M}$ against a null model m_0, defined below, using the Bayes factor (BF)[1]

$$BF_{m,m_0} = \frac{p(\mathcal{D}|M = m)}{p(\mathcal{D}|M = m_0)}. \tag{7.3}$$

If $\ln BF_{m,m_0} > 3$ for at least one model there is substantial evidence for the presence of a receptive field. This value of the BF corresponds to $20 \div 1$ odds against the null model. Conversely, if $\ln BF_{m,m_0} < -3$ for all models, odds are at least $20 \div 1$ in favor of the null model against all the alternative explanations, and we can conclude that spike generation is completely random – or at least not coupled to the stimulus – with 95% confidence. W found substantial evidence in favor of the presence of a receptive field in 78 cells ($\ln BF_{m,m_0} > 3$), while moderate to strong evidence in favor of m_0 was found in only 2 cells ($\ln BF_{m,m_0} < -3$; Fig. 7.2A). We found that evidence was always strong in either directions; in other words, the classification was always backed by high confidence. We repeated the same analysis considering only segments consisting of 1 or 2 minutes of data, and we found strong evidence in favor of at least one RF model in 59 and 75 cells, respectively(Fig. 7.2A). The complete outcome of this analysis is reported in Table F.3. From these results, we concluded that using 2 minutes of data provides enough evidence to detect a RF in most of the cells in this dataset. Nevertheless, we can expect than using more data is beneficial to identify more precisely which model provides a better explanation of the data. From now on, we will consider only those 78 cells for which one receptive field was detected using 4 minutes of data.

7.1.4 Model identification

Once we have established the presence of a receptive field, we would like to know which model, if any, provides the best explanation for the data. To this end, we inspect how the posterior belief is distributed across the four models. For each cell n, we compute the model posterior probabilities

$$P_n(m|\mathcal{D}_n) = \frac{p(\mathcal{D}_n|m)p(m)}{\sum_{m' \in \mathcal{M}} p(\mathcal{D}_n|m')p(m')}, \tag{7.4}$$

where $p(\mathcal{D}_n|m) = \exp(\hat{\mathcal{Z}}_{m,n})$ and $\hat{\mathcal{Z}}_{m,n}$ is the log-evidence estimated by the nested sampler for the model m fitted to the neuron n ($\mathcal{D}_n$ is the data associated to neuron n). We classify each cell according to the following rule: if $p(M =^* m|\mathcal{D}_n) > 95\%$, the cell is classified as an instance of model m^*. This threshold is equivalent to posterior odds at least as large as $19 \div 1$ favoring m^* against all competing alternatives. Overall, 44 out of 78 cells

[1]For the definition and meaning of Bayes factors, see Chapter 2.2.1.

Duration	LO	LD	QO	QD
1 min	9.3%	18.6%	17.1%	55.0%
2 min	6.2%	18.1%	14.4%	61.3%
4 min	5.7%	9.9%	17.5%	66.9%

Table 7.3: **Average posterior model probabilities.**

can be classified with high confidence as an instance of a specific model, while, for the remaining 34, the odds were not high enough in favor of any hypothesis, leaving these cells unclassified. 57 out of 78 cells are assigned to one specific model if we relax the classification criterion and classify based on whether $P(M = m^*|\mathcal{D}) > 75\%$ (corresponding to at least $3\div1$ posterior odds in favor of m^* against all the alternatives). The number of unclassified cells decreases accordingly, and is 21 in this scenario. Most cells in this dataset are best explained by the QD model, which represents 47 out 78 according to the relaxed classification criterion, whereas only 10 are instead classified as an instance of the remaining three models (Fig. 7.2B, dark bars). We repeated this analysis based on the posterior distributions inferred using only the first 2 minutes of data for each cell. The number of classified cells decreased to 42 out of 78 cells (Fig. 7.2B, light bars), supporting our earlier intuition that additional data would improve model identification, rather than receptive field detection. The model posterior probabilities, for each cell, are illustrated in Fig. 7.2C and D (2 and 4 minutes results, respectively). The average posterior model probabilities across all cells are reported in Table 7.3. Considering 4 minutes instead of 1 minute of data, roughly 12% of the mass shifts from the two linear models to QD, suggesting that the proper characterization of the additional parameter $\mathbf{C}$ in eq. (4.6) of a quadratic model may require more data than to characterize the linear parameter $\mathbf{b}$.

7.2 Single cell analysis

We will now illustrate a full Bayesian analysis of one single example cell. The cell we chose has an average firing rate of 0.33 Hz, for a total of 79 spikes within a 4 minutes observation window, and a measured signal-to-noise ratio of -19.11 ± 0.55 dB. It is therefore a quite typical cell in this dataset.

7.2.1 Model comparison

We found no strong evidence in favor of any of the tested models. Nevertheless, the QD model provides slightly better explanation of the data than

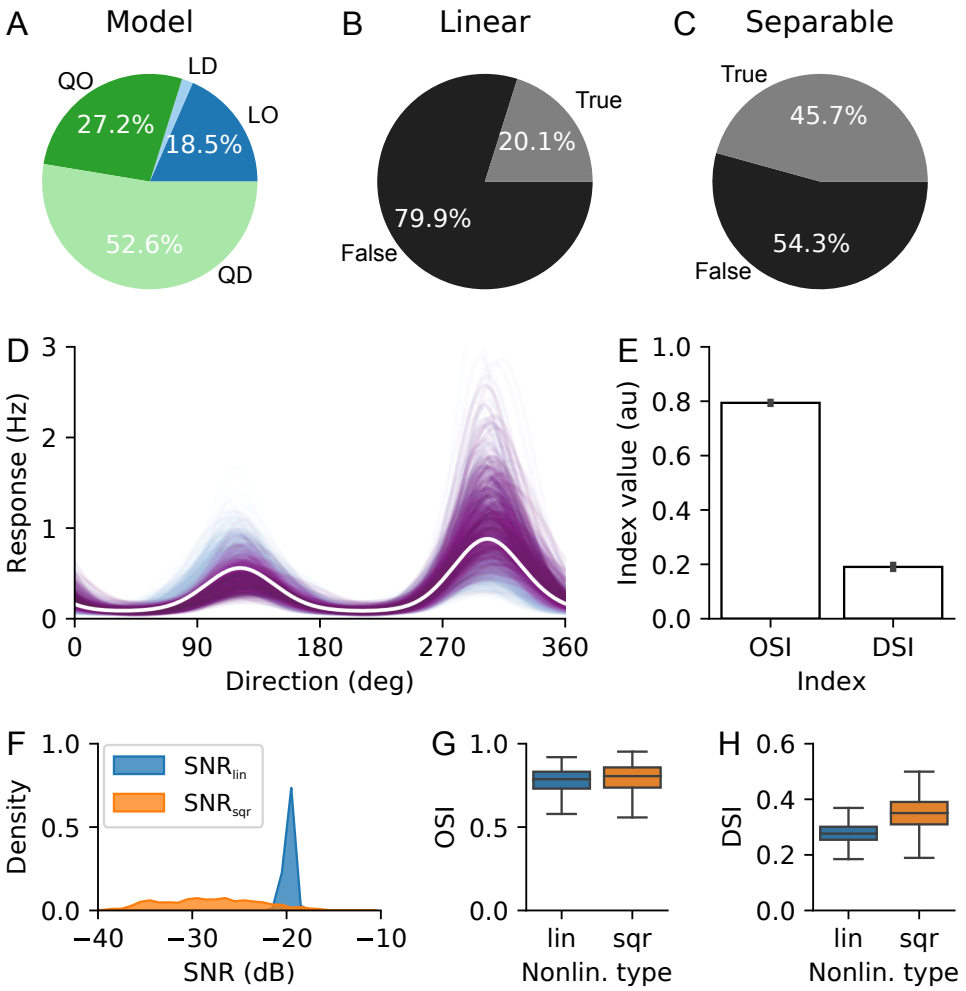

Fig. 7.3: **Model comparison and predicted properties. A)** Posterior probabilities of all 4 tested models. Most evidence points in favor of the QD model, although not strongly. **B)** Posterior belief in favor of linear (LO and LD; gray) or non-linear (QO and QD; black) models. **C)** Posterior belief in favor of separable (LO and QO; gray) or non-separable models (LD and QD; black); we observe no substantial evidence supporting either model class. **D)** Tuning curves predicted from the posterior distribution (thin lines) and posterior-predicted mean tuning curve (solid white line). **E)** Posterior mean OI and DI computed from the tuning curves in D, bars indicated 95% confidence intervals. This cells is clearly orientation selective, but can only slightly discriminate direction. **F)** Distribution of the linear (blue) and quadratic (orange) SNR computed from the posterior distribution of the QD model parameters. **G)** Predicted OI for the linear (LO and LD, blue) and quadratic (QO and QD, orange) models. **H)** Predicted DI for the LD (blue) and QD (orange) model; the quadratic model is more selective.

the alternatives (odds are about $2 \div 1$ for QD vs QO, $3 \div 1$ for QD vs LO, but $30 \div 1$ for QD vs LD; Fig. 7.3A). There is substantial evidence supporting a quadratic nonlinearity (odds are $4 \div 1$ for {QO, QD} vs {LO, LD}; Fig. 7.3B), but no clear indication whether a separable receptive field provides a better explanation than a non-separable one (Fig. 7.3C). Marginal posterior distributions of the QD model for 1 and 4 minutes of data are reported in Fig. F.9 and F.10

7.2.2 Orientation and direction selectivity

We then investigated the orientation and direction selectivity properties of this cell. Orientation selectivity is quantified by means of the *orientation index* (OI) [65] computed as

$$\mathrm{OI} = \frac{R_{pref} - R_{orth}}{R_{pref}}, \tag{7.5}$$

where R_{pref} and R_{orth} are, respectively, the average firing rates in response to a grating drifting in the preferred direction, θ_{pref}, and its two orthogonal directions. More specifically, $R_{orth} = (R_{orth+} + R_{orth-})/2$, where R_{orth+} and R_{orth-} are the average firing rates for a grating moving in the direction $\theta_{orth+} = \theta_{pref} + 90°$ and $\theta_{orth-} = \theta_{pref} - 90°$ (modulo 360°). As the name suggests, the preferred direction is the one eliciting the strongest response. Similarly, direction selectivity is quantified by the *direction index* (DI) [65], computed as

$$\mathrm{DI} = \frac{R_{pref} - R_{null}}{R_{pref}}, \tag{7.6}$$

where R_{null} is the response for a stimulus moving in the null direction, i.e. $\theta_{null} = \theta_{pref} + 180°$ (modulo 360°).

Since no model is a clear winner, we must resort to Bayesian model averaging (see Chapter 2.2.2) to investigate the posterior-predicted selectivity properties of this cell. First of all, we simulated the cell's response to multiple moving gratings

$$s(\mathbf{x}, t) = \cos\big(\kappa_0(\mathbf{n} \cdot \top\mathbf{x} - vt)\big) = \cos(\mathbf{k}_0 \cdot \mathbf{x} - \omega_0 t),$$

where $\mathbf{n}$ is a unit vector $\mathbf{n} = (\cos\theta_0, \sin\theta_0)$, with direction θ_0 discretized in 1° steps in the range 0° to 360°. We followed a Cartesian convention to represent directions, in which 0° corresponds to a vertical grating moving to the right. The spatial frequency of the grating, κ_0, is fixed to the posterior-predicted expected preferred frequency $\mathbb{E}_{\boldsymbol{\theta}_{RF}|\mathcal{D}}[\|\mathbf{k}_{RF}\|]$, while the temporal frequency to $\omega_0 = \mathbb{E}_{\boldsymbol{\theta}_{RF}|\mathcal{D}}[\alpha_{RF}]/5$. The posterior predicted means of (functions of) the model parameters is derived from eq. (2.7), which is

$$\mathbb{E}_{\boldsymbol{\theta}|\mathcal{D}}[f(\boldsymbol{\theta})] = \sum_{m \in \mathcal{M}} p(m|\mathcal{D}) \underbrace{\int f(\boldsymbol{\theta}) p(\boldsymbol{\theta}|m, \mathcal{D}) \mathrm{d}\boldsymbol{\theta}}_{\mathbb{E}_{\boldsymbol{\theta}|m,\mathcal{D}}[f(\boldsymbol{\theta})]} \tag{7.7}$$

Based on the closed-form solution reported in eq. (4.18) (for details, see methods), we evaluated the posterior-predicted tuning curves (Fig. 7.3D), from which we computed the OI and DI. This cell is quite tuned to the stimulus orientation (OI $= 0.88 \pm 0.06$, $P(\text{OI} > 0.66) = 99.0\%$), but only poorly to its direction (DI $= 0.28 \pm 0.26$; Fig. 7.3E). The small value of the DI was expected, given the amount of evidence supporting either one of the separable models (Fig. 7.3C), which deflate its value: the separable models cannot discriminate between stimuli moving in two opposing directions, therefore the response to θ_{pref} and θ_{null} are always equally strong, hence DI $= 0$ always for LO or QO.

7.2.3 Role of the nonlinearity

Since we observed substantial evidence supporting one of the two quadratic models (Fig. 7.3B), we will investigate more in detail the role played by the quadratic terms in eq. (4.6). First, we ask how much of the observed response variability can be explained by the linear and by the quadratic terms alone. We computed the signal-to-noise ratio of the linear (SNR_L) and of the quadratic terms (SNR_Q) for the QD model according to eqs. ((D.14), (D.15))), respectively. While the posterior predicted SNR_L is well constrained by the data ($\text{SNR}_L = -19.72 \pm 0.46$ dB), the quadratic component is not and it has a lower explanatory power than the linear term ($\text{SNR}_Q = -28.20 \pm 5.10$ dB). Nevertheless, despite its moderate contribution, the quadratic term must play a significant role in predicting the response of this neuron, otherwise we could not explain the larger amount of evidence in favor of the quadratic models.

The effect of the quadratic term is not easily interpretable from the posterior distribution of the matrix parameter $\mathbf{C}$ alone (see Chapter 4.2.4): the trace of $\mathbf{C}$ does not indicate whether the phase invariance component of the quadratic response is excitatory or suppressive (Fig. F.11A), while the second harmonic term (see eq. (4.23)) is clearly not zero (Fig. F.11B). We look instead at the effect that using a quadratic model has on the predicted OI and DI of the model. We compute again each index, but this time based only on the posterior distributions of the linear models (OI_L, DI_L) or of the quadratic models (OI_Q, DI_Q). While we observed no big differences for the predicted OI ($\text{OI}_Q = 0.88 \pm 0.07$; $\text{OI}_L = 0.88 \pm 0.06$; Fig. 7.3G), including a quadratic term improves the direction selectivity of this cell ($\text{DI}_Q = 0.52 \pm 0.07$; $\text{DI}_L = 0.44 \pm 0.04$; Fig. 7.3H; DI computed using only from LD and QD). As we will see in the next section, this is a general finding of this study.

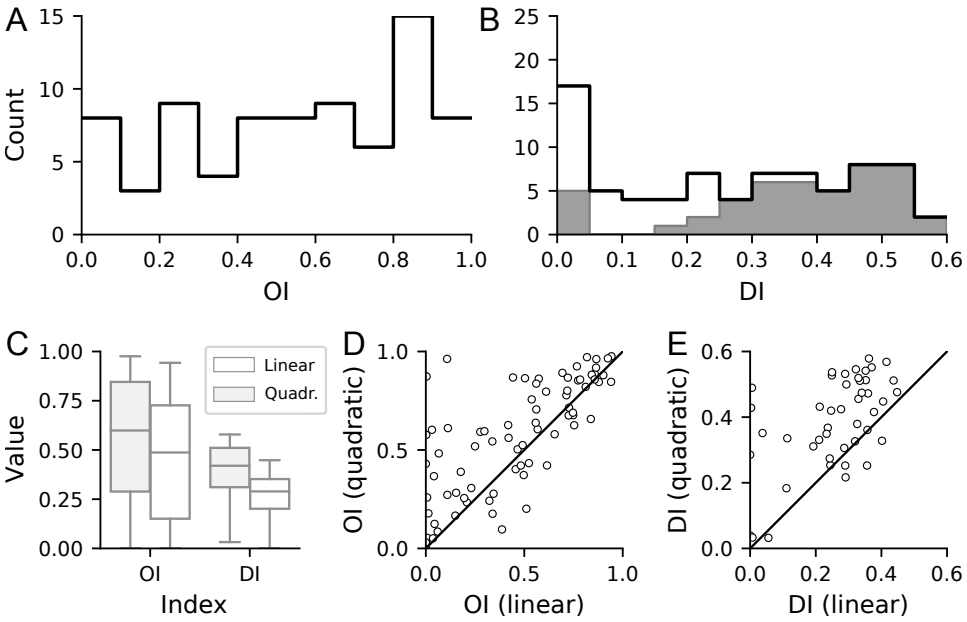

Fig. 7.4: **Orientation and direction selectivity. A)** OI distribution across all cells. **B)** DI distribution across all cells. The solid line outlines the DI distribution for those cells that are well described by one of the non-separable models (LD or QD). **C)** OI and DI distributions, conditioned on the type of model (linear or quadratic). **D)** OI values predicted by the quadratic models vs the values predicted by a linear model. **E)** DI values predicted by the quadratic models vs the values predicted by a linear model, for the same cells used in (B). Using quadratic features of the filtered stimulus improves the orientation and direction selectivity of the model, as measured by the posterior-predicted OI and DI.

7.3 Aggregate results

7.3.1 Orientation and direction selectivity

We will now characterize the orientation and direction selectivity properties of all cells in the dataset. In order to quantify orientation selectivity of a given cell n, we compute the OI posterior mean, $\mathbb{E}_{\boldsymbol{\theta}|\mathcal{D}_n}[\mathrm{OI}]$. Direction selectivity is instead quantified by the DI posterior mean, $\mathbb{E}_{\boldsymbol{\theta}|\mathcal{D}_n}[\mathrm{DI}]$. All aggregate values are reported as mean ± sd, if not otherwise specified.

OI values are quite homogeneously distributed between 0 and 1 (Fig. 7.4; $\mathrm{OI} = 0.55 \pm 0.30$) with only with 33 cells in this dataset being sharply tuned for orientation (i.e., they satisfy the criterion $\mathbb{E}[\mathrm{OI}] > 0.66$). Overall, cells in this dataset do not show signs of strong direction selectivity ($\mathrm{DI} = 0.27 \pm 0.19$; Fig. 7.4B). DI are higher (0.37 ± 0.16, mean ± SD) if we restrict our analysis to the 47 cells for which the non-separable models provide

a good explanation (Fig. 7.4B, shaded histogram), which we quantify as $P(M = \text{LD}|\mathcal{D}) + P(M = \text{QD}|\mathcal{D}) > 95\%$. From here on, all direction-selectivity analyses are performed only on these 47 cells.

Using quadratic features of the filtered stimulus increases both orientation and direction selectivity (for details, see methods). Not only the overall means are significantly larger ($\text{OI}_L = 0.49$, $\text{OI}_Q = 0.63$, $p = 2.1 \cdot 10^{-2}$; $\text{DI}_L = 0.25$, $\text{DI}_Q = 0.39$, $p = 2.2 \cdot 10^{-6}$; Wilcoxon rank-sum test; Fig. 7.4C), but OI and DI are significantly larger on a cell-by-cell basis (Fig. 7.4D and E, respectively; OI: $p = 1.9 \cdot 10^{-5}$, DI: $p = 5.7 \cdot 10^{-8}$, one-sided Wilcoxon signed-rank test).

7.3.2 Non-linear response properties

In the previous section we have established that our model predicts that the neurons in this dataset respond with a diverse range of selectivity to sinusoidal gratings drifting through their receptive fields. In this section, we want to assess the strength of the modulation that a stimulus exerts on the neural response. A primary choice to quantify this is the modulation index (MI) [69, 70] defined as the ratio of the magnitude of the response at the fundamental frequency of the stimulus (F_1 component) to the magnitude of the average response (F_0), or in other words, the F_1 to F_0 ratio (see Methods section below). MI was introduced to assess the degree of linearity in spatial summation within receptive fields of single neurons in the primary visual cortex. MI values close to 1 suggest a strong linear modulation of the response. Low values, on the other hand, indicate that either the neuron is not responsive to the stimulus or that the response is modulated in a non-linear fashion. Our analysis yields $\text{MI} = 0.57 \pm 0.25$ (mean $\pm$ sd) across the entire dataset. For cells that are well described in terms of a quadratic model, i.e. $P(m \in \{\text{QO}, \text{QD}\}|\mathcal{D}) > 95\%$ (54 cells), which we will refer to as "quadratic cells", the posterior-predicted $\text{MI} = 0.50 \pm 0.26$ is significantly lower than for the remaining cells, which is $\text{MI} = 0.70 \pm 0.14$ (mean $\pm$ sd; $p = 2.87 \cdot 10^{-4}$, Wilcoxon rank-sum test; Fig. 7.5A). This result is expected, since the quadratic models include a phase invariant component in the predicted response. Furthermore, $\text{MI} > 0.66$ for 17 non quadratic cells, which accounts for 71% of this group, but only for 18 quadratic cells, i.e. 33% of them.

In order to understand if the lower predicted MI is due to a strong role played by the quadratic term, or whether instead these quadratic neurons are just less responsive to the stimulus, we assess the distribution of signal-to-noise ratios (SNR) of the entire population (the details of computing the SNR for a GLM neuron are outlined in Appendix D). The SNR compares the strength of the response to the level of background noise, which in our case is the intrinsic noise of the spike-generating process (for details, see method). A high SNR indicates that the stimulus-driven component plays

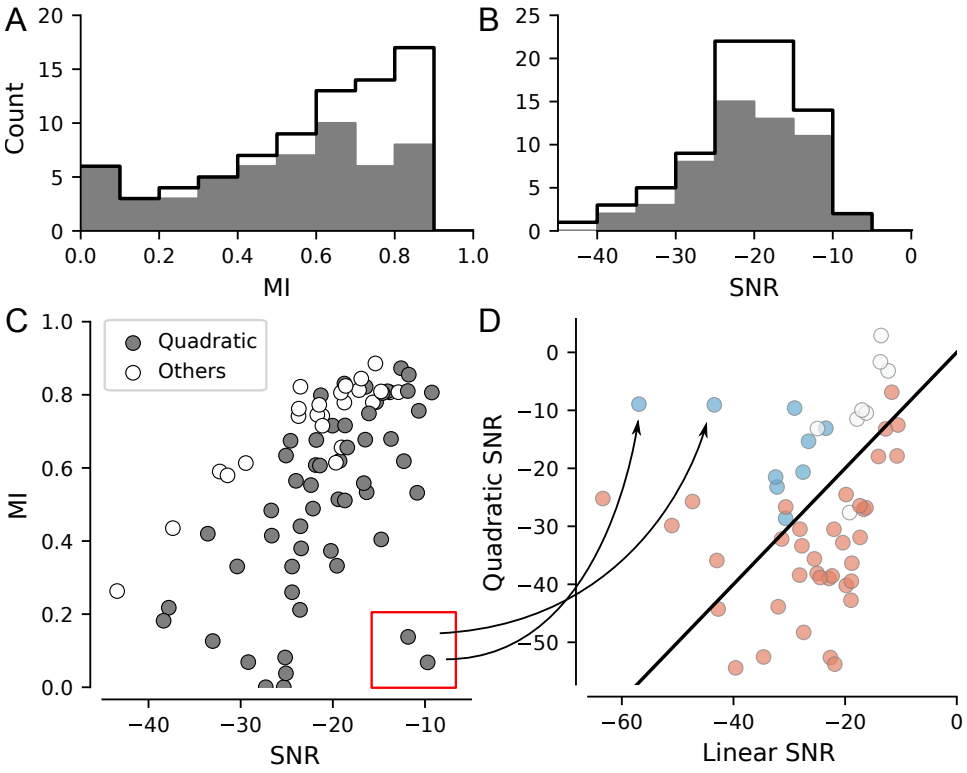

Fig. 7.5: **Non-linear response properties. A)** MI distribution across the dataset (solid black line), and across quadratic cells (shaded area); for the definition of quadratic cells, see main text. **B)** SNR distribution across the dataset (solid black line), and across quadratic cells (shaded area). **C)** MI vs SNR; the two are quite correlated, suggesting that a low MI is indicative of poorly responsiveness to the stimulus, rather than of some non linear interaction. **D)** Linear vs quadratic SNR, across quadratic cells; the color of the dot encodes the effect of the quadratic term: inhibitory (blue), excitatory (red) or undefined (white). There is no significant correlation between linear and quadratic SNR; however, for all cells with inhibitory quadratic terms, the quadratic SNR is larger than the linear SNR.

a major role in shaping the response. In our dataset, the SNR ranges from -37.8dB to -10.6 dB, with a median value of -20.12dB, values compatible with those measured also in other areas of the brain [18]. Furthermore, the predicted distribution for the quadratic cells is not significantly different than the one for the remaining cells value are not significantly differently ($p = 0.31$, Mann Whitney U test), suggesting that the stimulus drive is not different across these two subgroups (Fig. 7.5B). MI and SNR are positively correlated ($\rho = 0.51$, $p = 2.07 \cdot 10^{-6}$, Pearson's rho), even more so if we consider only the non quadratic cells ($\rho = 0.90$, $p = 2.92 \cdot 10^{-9}$, Pearson's rho; Fig. 7.5C), suggesting that indeed low MI values are due to a small stimulus drive rather than to a quadratic (e.g. phase invariant) modulation, except for two cells which have a high SNR but pretty low MI (Fig. 7.5C, encircled dots).

Nevertheless, for about 2 out of 3 cells in this dataset, we found substantial evidence for quadratic models. Therefore, we think it is appropriate to investigate a more deeply the role of the quadratic terms in the response. To this end, we compute SNR_L, the SNR attributed exclusively to the linear component of the response, and SNR_Q, the SNR of the quadratic term Fig. 7.5D; for details, see Appendix D.4) – we restricted this analysis only to the 54 quadratic cells. These two indexes are informative of how much of the firing rate variability is explained by the linear and the quadratic terms of the nonlinearity. Their distributions are not significantly different ($p = 0.22$, Mann-Whitney U test), nor we could measure any significant correlation ($\rho = 0.22$, $p = 0.11$, Pearson's rho). For 32 cells $\mathrm{SNR}_L > \mathrm{SNR}_Q$, but result does not point to any significant underlying relationship ($p = 0.28$, Wilcoxon signed-rank test). We then investigated the effect of the quadratic terms. We classified each cell as "quadratically inhibited" if the posterior probability of $\mathrm{tr}(\mathbf{C}) < 0$ was larger than 0.95 (Fig. 7.5D, blue dots); conversely, as "quadratically excited" if the trace of $\mathbf{C}$ was positive with at least 95% confidence (Fig. 7.5D, red dots); we left them unlabeled if none of the criteria were met. Very interestingly, for all cells with clearly inhibitory quadratic interactions (9 cells) SNR_Q was always (significantly) larger than SNR_L ($p = 3.91 \cdot 10^{-3}$, Wilcoxon signed-rank test; Fig. 7.5D, blue dots). Furthermore, for these 9 cells, DI was significantly smaller than for the rest of the subset ($DI = 0.047 \pm 0.062$, mean $\pm$ sd; $p = 1.53 \cdot 10^{-3}$, Wilcoxon rank-sum test). No other significant effect was measured on the other indexes.

7.4 Material and Methods

Data acquisition

Data was acquired by means of juxtacellular electrophysiological recording in layer 2/3 or rat visual cortex during visual stimulation. The details of the experimental procedure are reported in Appendix E.

The duration of each recording session was 5 minutes.Should I give this detail? An initial screening of the data revealed that the stability of the recordings degraded with time, therefore we used only the first 4 minutes of each recording for the analyses reported in this chapter.

Spikes times were binned according to stimulus presentation times.

Models details

We used a $\ln(1 + e^x)$ nonlinearity for all model configurations considered in this chapter. We used the same prior distributions for all four configurations, since they all share the same parameters. We used the same priors we used in Chapter 6. Details can be found in Table 6.4.

Posterior sampling

The posterior distribution of model parameters was estimated using the algorithm outlined in Chapter 5, using 128 live points. New proposals from the likelihood-restricted prior were generated by evolving a random live point with 25 random-walk steps.

Tuning curve

For each sample $\boldsymbol{\theta}^(i)$ from the posterior distribution, we simulated the response of the model to a moving sinusoidal grating with spatial frequency $\mathbf{k}_0$ and temporal frequency ω_0. Let us denote the response as $r(t; \mathbf{k}_0, \omega_0, \boldsymbol{\theta}_i)$. The tuning curve was computed by averaging the response over one period of the stimulus:

$$\mathrm{tc}(\mathbf{k}_0, \omega_0, \boldsymbol{\theta}^(i)) = \frac{1}{T} \int_0^T r(t; \mathbf{k}_0, \omega_0, \boldsymbol{\theta}^(i)) \mathrm{d}t, \tag{7.8}$$

where $T = 2\pi/\omega_0$ is the period. We discretized on temporal period into a grid of 100 points and approximated the integral using the trapezoidal rule.

The direction of motion was discretized in 360° steps in the range $[0, 2\pi)$, corresponding to a resolution of 1°. The spatial frequency was fixed to the expected preferred spatial frequency $\|\mathbf{k}_0\| = \mathbb{E}_{\boldsymbol{\theta}|\mathcal{D}}\big[\|\mathbf{k}_{RF}\|\big]$, while the temporal frequency was fixed to the expected posterior-predictive preferred temporal frequency $\omega_0 = \mathbb{E}_{\boldsymbol{\theta}|\mathcal{D}}\big[\alpha_{RF}/5\big]$.

Modulation index

The modulation index was computed as follows. Denoting the response to a moving grating with temporal frequency ω_0 as $r(t)$, and its Fourier transform as $\hat{r}(\omega)$, we define $F_1 = \hat{r}(\omega_0)$ and $F_0 = \hat{r}(0)$. Therefore MI $= \hat{r}(\omega_0)/\hat{r}(0)$.

7.5 Summary and Discussion

In this chapter we showed how a full Bayesian analysis applied to real electrophysiological recordings may lead to interesting insights about the properties of the cells in the dataset. We first showed that is possible to categorize cells according to the type of their response to visual stimulation and that for about half of the cells in the dataset this classification can be performed with high confidence. Concretely, we found that most cells in this dataset are best described in terms of one of the two quadratic models proposed. Only a small minority of cells are best described by one of the linear models, and specifically only 1 out of 80 with high confidence after observing 4 minutes of data. Although they not always play a major role in terms of explained neural variability, this analysis reveals that these features contribute enough to justify their presence in the model, in contrast to preferring a more parsimonious, purely linear model. This suggests that the contribution of these three second order features to the firing of the neuron is not negligible. In a future study, we may drop the linear models altogether.

We characterized the orientation and direction selectivity properties of these neurons. Our analysis revealed an heterogeneous landscape of orientation selectivities, predicting strong orientation selectivity for about 40% of the cells in the dataset. The predicted direction selectivity was instead way milder. This fact does not necessarily reflect a true underlying feature of these cells, but may be imputable to a limitation of our RF model: with the current parameterization, stimuli moving in the null direction always elicit a positive response –albeit smaller in magnitude than for the preferred direction– therefore preventing the DI from taking values close to 1. This shortcoming can be fixed by adopting a different parameterization that explicitly models a property called motion opponency. We discuss how to extend the model in this direction in Appendix C.1. We also showed that the role of the quadratic features consists mostly in sharping the orientation and direction tuning.

Finally, our analysis revealed that for the vast majority the cells in this dataset exhibit a mostly linear response. Their responsiveness to the visual stimulus is similar in strength to that of sensory neurons in other sensory areas.

Chapter 8

Conclusions

In this thesis we have addressed the problem of the characterization of receptive fields in primary visual cortex in terms of high-level, interpretable features. We placed particular emphasis in the quantification of the estimation uncertainty when data is sparse or noisy. Since we have summarized and discussed the results at the end of each chapter, here we provide a short summary and an overarching overview of our findings, and suggest further potential developments.

Functional receptive field models are routinely fitted to non-parametric estimates to infer high-level properties of the stimulus encoding (e.g, the orientation selectivity) [89, 93]. These "second hand" estimates may be used to test some hypothesis about the properties or the structure of the receptive field. According to this approach, one implicitly assumes that MLE or MAP non-parametric point estimates present a good characterization of the receptive field. This strategy does not account for any uncertainty on the MLE or MAP estimates on which the inference of the functional model is based, with two major drawbacks: confidence intervals on the derived parameters are too tight, which, in turn, may lead to wrong conclusions. This problems are exacerbated with sparse or noisy data.

We adopted a different strategy and inferred the relevant high-level receptive field features directly from the data. We operated within a Bayesian framework, which naturally allows to account for estimation uncertainty by encoding it in the posterior distribution of model parameters. With sparse or noisy data, access to the full posterior is critical, both to avoid over-fitting and to quantify uncertainty. Furthermore, posterior distributions provide a principled base to test different or competing hypothesis regarding the data-generation process.

We included prior knowledge by modeling the known structure of V1 receptive fields using a well-defined functional model. In Chapter 4 we presented a compact generative model of neural responses in V1 relying on 10 free parameters to describe the entire spatio-temporal structure of

a receptive field. Thanks to its flexibility, this model can reproduce most stereotypical properties of visual cortical neurons such as orientation and direction selectivity, linear modulation and phase invariance. Similarly to some recent work on spline-base receptive field modeling [44], our aim is to facilitate the inference procedure by focusing the (possibly scarce) evidence in the data on a small number of parameters. Clearly, this model is a stark over-simplification of stimulus selectivity in visual cortical neurons, [1] however in Chapters 6 and 7 we showed it is nevertheless useful for obtaining a fast characterization of a receptive field directly from sparse or noisy data without any intermediate analysis steps.

In Chapter 6, we studied how confidently we can recover ground-truth generating parameters as a function of the amount of available data (measured in terms of observed spike counts) and overall noisiness of the neuron. We showed that in physiological settings a receptive field can be detected from a few tens of observed spikes and that the identification error quickly shrinks to a few percent points accordingly. These are of course optimistic lower bounds on the potential identification errors, obtained from a simulation study with no model mismatch, but they may nevertheless provide a useful guideline for planning the amount of data to collect within an experiment if one wants to attain a certain level of confidence on the estimates. In Chapter 7 we analyzed a dataset of electrophysiological recordings acquired in rat primary visual cortex and were able to show that a full Bayesian treatment of the problem can lead to interesting insights despite the scarcity of available data. Not only that, we could also associate a degree of confidence to each statement.

The computational resources required by a full Bayesian approach remains one of the main challenges to its widespread adoption in data analysis. Our application is not different: despite the small number of parameters, the evaluation of the receptive field response – which is a necessary step to evaluate the goodness of each new proposed sample – is the main computational bottleneck of our model. To address this issue, in Chapter 5 we introduced Collapsed Nested Sampling (CNS). This algorithm exploits some geometrical properties of our model to marginalize some of its parameters and explore the resulting lower-dimensional parameter space using ordinary nested sampling (ONS). Marginalizing out some of the model parameters smooths the likelihood landscape of the remaining ones, hopefully making it easier for the sampler to explore. We benchmarked CNS against ONS on the task of estimating the posterior distribution of the parameters of our generative model, and showed that CNS has comparatively better convergence properties and reduced storage requirements. The potential application of CNS goes beyond the generative model used in this work: it can indeed be

[1] In Appendix C, we discuss some extensions to overcome some of the limitations of this model that became evident during our analyses.

adopted to any GLM making use of parametric basis functions to efficiently sample from the joint distribution of basis functions parameters and GLM coefficients.

We developed this method to infer visual receptive field properties from high-dimensional, unstructured visual stimuli. As mentioned above, due to the size of the tensor operations involved in computing the receptive field response to unstructured video data, a few minutes are required to analyze even short segments of data, even when all tensor operations take advantage of the hardware acceleration provided by the GPU. Certain types of stimuli, however, can be conveniently expressed by a small number of control parameters – this is, e.g., the case of the counterphase or drifting sinusoidal gratings considered in Chapter 4. In such cases, it is possible to derive an analytical expression to compute the receptive field response, which can be evaluated almost instantaneously compared to the general-purpose, tensor-based implementation. We have not thoroughly tested this idea in our work, but we expect inference to be much faster in this context. This opens the door to a series of interesting scenarios: we could infer receptive field properties under different types of visual stimulation and test the consistency of the neural representation across stimulus classes; we could analyze short segments of responses to structured stimuli to get a first rough idea of the frequency-response properties of a neuron, in order to design optimal unstructured stimuli for further experiments; the possibilities offered here are really many, and mostly limited by our creativity.

Although in this thesis we focused exclusively on modeling neural activity in the primary visual cortex, the potential scope of application of our work is broader. We have already mentioned that it is possible to extend the model or to adopt a slightly different functional parameterization of the spatial filters (see Appendix C). As a matter of fact, the entire functional form of the receptive field can be changed without invalidating our approach: neural data from any sensory area for which canonical receptive fields models have been developed can potentially be analyzed with the algorithms presented in this thesis. For example, many neurons in the retina and in the LGN have center-surround receptive fields[22], which can be modeled as a temporally-weighted difference of 2D Gaussian using just 7 parameters [20]. All considerations on deriving an analytical expression to evaluate the response to structured stimuli are valid also in this scenario, provided the core model structure does not change.

Overall, we believe that this work may benefit the neuroscientific community to make use of very sparse or noisy data from a variety of sensory areas, in turn playing a major role in contributing to the progress of our knowledge of how sensory information is represented by the living brain. Considering an even broader context, the computational advantages offered by CNS may increase the feasibility of a full Bayesian approach to the wide model family represented by GLMs with parametric basis functions, with

potential applications to shallow neural architectures in machine learning.

Appendix A

Generalized Linear Models

Generalized Linear Models (GLMs) are a flexible generalization of ordinary multilinear regression to response variable following a distribution other than normal [66, 72]. We consider here a dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}$ consisting of pairs of independent and dependent variables. We can also compactly represent such a dataset using the design matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^\top$ and a vector $\mathbf{y} = [y_1, y_2, \ldots, y_n]^\top$. Each observation of the dependent variable y_i is assumed to be generated from a particular distribution within an exponential family [4, 57, 86],[1] which can be expressed in the form

$$f_Y(y_i; \theta_i, \phi) = h(y, \phi) \exp\left(\frac{y_i\theta_i - A(\theta_i)}{\delta\phi}\right),$$

where $A(\theta_i)$, $h(y, \phi)$ and $\delta(\phi)$ are known functions. The dispersion parameter ϕ is typically known and is usually related to the variance of the distribution. The parameter θ_i is the natural parameter of the distribution and it is related to its mean. The function $A(\theta_i)$ is a convex function, called the *log-partition function* because it is the logarithm of the normalization factor that makes $f_Y(y; \theta_i, \phi)$ a probability distribution (or probability mass function, for the case of a discrete distribution). The mean and the variance of the distribution are $\mu = \mathbb{E}[Y] = A'(\theta_i)$ and $\mathrm{Var}[y_i] = A''(\theta_i)\delta(\phi)$ [72]. In a GLM, the mean of the distribution depends on the independent variables $\mathbf{x}_i$ through:

$$\mathbb{E}[y_i|\mathbf{x}_i] = \mu_i = g^{-1}\big(\mathbf{x}_i^\top \boldsymbol{\beta}\big), \tag{A.1}$$

where $\mathbb{E}[y|\mathbf{x}_i]$ is the expected value of y_i conditional on $\mathbf{x}_i$; g is the *link function*[2] relating the mean of the observed variable to a *linear predictor* $\eta_i = \mathbf{x}_i^\top \boldsymbol{\beta}$, a linear combination of unknown parameters $\boldsymbol{\beta}$.The linear predictor incorporates the information about the independent variables into the model. For each family, there is a particular link function that makes

[1] A large class of probability distributions that include the normal, Bernoulli and Poisson distributions, among many others.

[2] Its inverse is the *mean function*.

Distribution	Support	Link Function	Mean function
Normal	real: $(-\infty, +\infty)$	$\mathbf{x}^\top \boldsymbol{\beta} = \mu$	$\mu = \mathbf{x}^\top \boldsymbol{\beta}$
Poisson	integer: $0, 1, 2, \ldots$	$\mathbf{x}^\top \boldsymbol{\beta} = \ln(\mu)$	$\mu = \exp(\mathbf{x}^\top \boldsymbol{\beta})$
Bernoulli	integer: $\{0, 1\}$	$\mathbf{x}^\top \boldsymbol{\beta} = \ln\left(\frac{\mu}{1-\mu}\right)$	$\mu = \frac{1}{1+\exp(-\mathbf{x}^\top \boldsymbol{\beta})}$

Table A.1: **GLM examples.** Support and canonical link-function for the exponential families discussed in Chapter 3.

the linear predictor the natural parameter of the distribution: this function is called the *canonical link function* of the GLM. A single-filter linear-nonlinear cascade model (see Chapter 3) can be interpreted as a GLM. The static nonlinearity is the inverse (of a possibly non canonical) link function. Table A.1 table lists the canonical link functions and their inverse for the three exponential-family distributions encountered in Chapter 3.

A.1 Learning the model parameters

The optimal value of the model parameters $\boldsymbol{\beta}$ can be found by maximizing its likelihood under the observed dataset; mathematically, by solving the following optimization problem:

$$\hat{\boldsymbol{\beta}}_{ML} = \arg\max_{\boldsymbol{\beta}} p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}). \tag{A.2}$$

Since all observation are assumed to be independent conditional on the value of the predictor variables, the conditional probability in the above equation can be expressed as a product of independent terms:

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) = \prod_{i=1}^{n} p(y_i|\mathbf{x}_i, \boldsymbol{\beta}). \tag{A.3}$$

Since directly maximizing (A.2) may be numerically unstable task, it is customary to maximize the corresponding negative log-likelihood instead:

$$\hat{\boldsymbol{\beta}}_{ML} = \arg\min_{\boldsymbol{\beta}} -\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) = \arg\min_{\boldsymbol{\beta}} -\sum_{i=1}^{n} \log p(y_i|\mathbf{x}_i, \boldsymbol{\beta}). \tag{A.4}$$

By making use of the canonical link function, the quantity $\mathbf{X}^\top \mathbf{y}$ becomes a sufficient statistic for $\boldsymbol{\beta}$, and the above equation can be expressed as:

$$\hat{\boldsymbol{\beta}}_{ML} = \arg\min_{\boldsymbol{\beta}} \left(-(\mathbf{y}^\top \mathbf{X})\boldsymbol{\beta} + \sum_{i=1}^{n} A(\mathbf{x}_i \boldsymbol{\beta}) \right) + \text{const}, \tag{A.5}$$

where const is a constant term in $\boldsymbol{\beta}$, which can be ignore for optimization purposes. Since $A(\theta)$ is convex and $(\mathbf{y}^\top \mathbf{X})\boldsymbol{\beta}$ is linear, the negative log-likelihood is convex in $\boldsymbol{\beta}$ and a single global minimum exists [72]. Therefore, the maximum likelihood solution can be found with standard convex-optimization techniques like the Newton method.

For a non-canonical link function, the optimization problem is expressed in a more complex form. For a Poisson GLM, this would be:

$$\hat{\boldsymbol{\beta}}_{ML} = \arg\min_{\boldsymbol{\beta}} - \sum_{i=1}^{n} \Big(y_i \ln f(\mathbf{x}_i\boldsymbol{\beta}) - f(\mathbf{x}_i\boldsymbol{\beta}) \Big), \tag{A.6}$$

where f is the inverse of the link function. This expression is convex as long as f is convex and log-concave, i.e. it grows at least linearly and at most exponentially [76]. Examples of mean function satisfying these constraints are, among others, $f(x) = e^x$ and $f(x) = \ln(1 + e^x)$.

Maximum *a posteriori*

For a series of reasons, it may be desirable or needed to add additional regularization terms to the cost function in (A.4). For example to regularize the estimate and avoid overfitting (when the dimensionality of the parameters space is large compared to the size of the dataset) or to include prior information we may have about the system we are modeling. We can interpret this extra term as a probability distribution over the model parameter, $p(\boldsymbol{\beta})$, encoding our prior belief over $\boldsymbol{\beta}$ before any data is acquired. After observing the data, $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$, the belief over $\boldsymbol{\beta}$ is updated by including the newly acquired information into a posterior probability distribution $p(\boldsymbol{\beta}|\mathbf{X}, \mathbf{y})$. Following Bayes' rule, this is proportional to $p(\mathbf{y}, \boldsymbol{\beta}|\mathbf{X}) = p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta})$. Instead of maximizing the likelihood, we maximize the joint probability distribution $p(\mathbf{y}, \boldsymbol{\beta}|\mathbf{X})$, obtaining the *maximum a posteriori* estimate:

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{MAP} &= \arg\max_{\boldsymbol{\beta}} \, p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) p(\boldsymbol{\beta}) \\ &= \arg\min_{\boldsymbol{\beta}} \, - \big[\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) + \log p(\boldsymbol{\beta}) \big]. \end{aligned} \tag{A.7}$$

Provided the prior is log-concave, the optimization problem (A.7) is still convex and therefore it still enjoys the same convergence properties of the maximum likelihood estimates, provided the same requirements for (A.5) or (A.6) are met.

A.2 Generalized Quadratic Models (GQM)

Generalized quadratic models extend of GLMs to model an additional quadratic dependence on the predictor variables:

$$\mathbb{E}[y|\mathbf{x}] = \mu = g^{-1}\big(a + \mathbf{b}^\top \mathbf{x} + \mathbf{x}^\top \mathbf{C} \mathbf{x}\big), \tag{A.8}$$

where $\mathbf{b}$ and $\mathbf{C}$ are a scalar, a vector and a square symmetric matrix of appropriate size; the scalar a models the implicit bias term that is often assumed by a GLM.[3] While quadratic on its inputs, a GQM is still linear on its parameters, therefore it can be conceived as a GLM on the space of quadratically-transformed inputs [32]. As such, a GQM enjoys the same convergence properties a GLM does. It is important to notice that the number of parameters scales as $O(m^2)$, where m is the dimensionality of the input. Fitting such a large number of parameters would require massive amounts of data or very strong regularization of the estimate. However, even if these were not an issue, for large values of m, the amount of memory required to store all this parameters may grow beyond the capacity of the system.[4] In such a scenario it could be desirable to reduce the dimensionality of the problem by adopting a low-rank factorization of $\mathbf{C}$: $\mathbf{C} = \mathbf{W}\mathbf{D}\mathbf{W}^\top$, where $\mathbf{D} \in \mathbb{R}^{p \times p}$ is a diagonal matrix with entries $d_{ii} \in \{-1, +1\}$, and $\mathbf{W} \in \mathbb{R}^{m \times p}$. This parameterization effectively reduces the number of parameters from $m(m+3)/2+1$ to $m(p+1)+1$, but does not guarantee the same convergence properties of a GLM, since now the model is no more linear in its parameters.

A.3 Basis functions

GLMs offer quite some flexibility in building the design matrix. So far we have assumed that all independent variables are used as they are in building the design matrix. For a dataset $\{\mathbf{s}_i, y_i\}_{i=1\dots n}$ like the one described at the beginning of this chapter,[5] this means that each rows of design matrix $\mathbf{X}$ is given by the corresponding set of independent variables, i.e.

$$\mathbf{X} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n]^\top$$

. This, however is just one of many possible design choices. As we have already mentioned in the previous section, a GQM can be treated as a GLM on the space of quadratically-transformed input, i.e. we can write

$$\mathbf{x}_i = [1, s_1, s_2, \dots, s_n, s_1^2, s_2^2, \dots, s_n^2, s_1 s_2, s_1 s_3, \dots, s_{n-1} s_n].$$

We can generalize this concept by letting each entry in the design matrix be some fixed function of the vector of covariates [24, 32]:

$$\mathbf{x}_i = \mathbf{h}_{\boldsymbol{\psi}}(\mathbf{s}_i) = [h_1(\mathbf{s}_i; \boldsymbol{\psi}), h_2(\mathbf{s}_i; \boldsymbol{\psi}), \dots, h_p(\mathbf{s}_i; \boldsymbol{\psi})], \tag{A.9}$$

[3] For consistency with (A.8), we should rewrite (A.1) as $\mathbb{E}[y|\mathbf{x}] = \mu = a + \mathbf{b}^\top \mathbf{x}$, where the linear predictor is now $\eta = a + \mathbf{b}^\top \mathbf{x}$. However, this expression is more compactly represented using the formalism in (A.1), taking $\boldsymbol{\beta} = [a, b_1, b_2, \dots]^\top$ and prepending a constant entry with value 1 to each vector of covariates $\mathbf{x}$.

[4] For $150k$ parameters (100 pixels, integrating 15 frames in the past, i.e. to 0.5s, at a sampling rate of 30Hz), the total number of parameters is $2.25 \cdot 10^1 0$.

[5] Here we denote the independent variables of the problem using the symbol $\mathbf{s}_i$ to distinguish them from the rows of the design matrix, $\mathbf{x}_i$.

where $\boldsymbol{\psi}$ is a parameter vector controlling the behavior of each $h_j(\cdot)$. This technique has been used, for example, to model the dependency of the firing rate of a neuron on a linear mixture of smoothed versions of its own past activity [84, 110]. In general, the conditional probability of the dependent variables given the inputs must take into account this new set of parameters, that is $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}) = p(\mathbf{y}|\mathbf{S}, \boldsymbol{\beta}, \boldsymbol{\psi})$, where $\mathbf{S}$ is a matrix with each row equal to the corresponding observation of the independent variables (i.e. the "default" design matrix). The parameters $\boldsymbol{\psi}$ and p are usually considered fixed and their value is a modeling choice. In Chapter 5 we describe an algorithm to perform Bayesian inference on $\boldsymbol{\psi}$, which exploits the convenient convergence properties of GLMs.

Appendix B

Frequency response properties of the receptive field model

In this appendix we derive the frequency response properties of the receptive field model, which provide some context for the study presented in Chapter 4.2. Throughout this appendix we will use $\mathbf{u}$ and ω to denote the normalized 2D spatial frequency and temporal frequency, respectively, and j is used to denote the imaginary unit.

B.1 Spatial filter

The Fourier transform (FT) of the spatial filter (4.4) is

$$\hat{g}(\mathbf{u}) \triangleq \mathcal{F}\{\hat{g}\}(\mathbf{u}) = \exp\left(-\frac{1}{2}(\mathbf{k}+\mathbf{u})^\top \mathbf{\Sigma}\,(\mathbf{k}+\mathbf{u})\right) e^{j\varphi}. \tag{B.1}$$

By applying some basic properties of the FT, we find also that

$$\hat{g}_c(\mathbf{u}) = \frac{1}{2}\Big(\hat{g}(\mathbf{u}) + \overline{\hat{g}(-\mathbf{u})}\Big) \quad \text{and} \quad \hat{g}_s(\mathbf{u}) = \frac{1}{2j}\Big(\overline{\hat{g}(-\mathbf{u})} - \hat{g}(\mathbf{u})\Big). \tag{B.2}$$

When fixing $\varphi = \pi/$, the corresponding power spectra $|g_c(\mathbf{u})|^2$ and $|g_s(\mathbf{u})|^2$ are identical and equal to

$$|\hat{g}(\mathbf{u})|^2 + |\hat{g}(-\mathbf{u})|^2 = \exp\big(-2(\mathbf{u}^\top \mathbf{\Sigma}\,\mathbf{u} + \mathbf{k}^\top \mathbf{\Sigma}\,\mathbf{k})\big)\cosh(2\mathbf{u}^\top \mathbf{\Sigma}\mathbf{k}). \tag{B.3}$$

From this expression we can derive the orientation and frequency tuning curves of the spatial filters and have a better understanding of how the model parameters shape the selectivity of the filters (Fig. B.1A and B). The preferred spatial frequency is $|\mathbf{u}| = |\mathbf{k}|$, from the fact that all curves in Fig. B.1A have a peak at $|\mathbf{u}|/|\mathbf{k}| = 1$. There orientation tuning is periodic with period 180° and has a peak at $\angle\mathbf{u} = \angle\mathbf{k} + n\pi$ for $n \in \mathbb{Z}$.

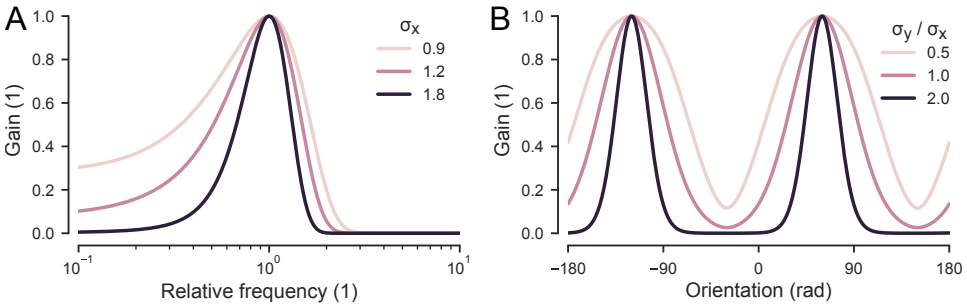

Fig. B.1: **Selectivity of the spatial filters. A)** Frequency tuning, computed by evaluating the power spectrum (B.3) along a line such that $\angle\mathbf{u} = \angle\mathbf{k}$; the horizontal axis is the relative frequency $|\mathbf{u}|/|\mathbf{k}|$. **B)** Orientation tuning, computed by evaluating the power spectrum (B.3) on a circle with constant radius, i.e. for $|\mathbf{u}| = |\mathbf{k}|$.

B.2 Temporal filter

The basic building block of the temporal filter is the the gamma density with integer shape parameter m

$$\gamma_m(x;\alpha) = \frac{\alpha^{m+1}}{m!} x^m e^{-\alpha x} \cdot u(x), \tag{B.4}$$

out of which we can build the spatial kernels h_n as

$$h_n(t;\alpha) = \gamma_n(t;\alpha) - \kappa_{bp}\,\gamma_{n+2}(t;\alpha). \tag{B.5}$$

The Laplace transform of a gamma density with integer shape parameter m has a convenient recursive structure for $m > 2$, which allows it to be expressed as a product of transforms of lower-order filters:

$$\Gamma_m(s) \triangleq \mathcal{L}\{\gamma_m\}(s) = \frac{\alpha^{m+1}}{(s+\alpha)^{m+1}}, \quad \Gamma_m(s) = \Gamma_{m-2}(s)\Gamma_1(s). \tag{B.6}$$

We can use the above identities to arrive at an expression for the FT of the temporal filter defined in eq. (4.10). First, we consider the Laplace transform of the generic filter $h_n(t)$, which is

$$H_n(s) \triangleq \mathcal{L}\{h_n\}(s) = \Gamma_n(s) - k_{bp}\Gamma_{n+2}(s) \tag{B.7a}$$

$$= \Gamma_n(s)\big(1 - k_{bp}\Gamma_1(s)\big) \tag{B.7b}$$

For $n \geq 3$, the above expression also has a recursive structure,

$$H_n(s) = \Gamma_1(s) H_{n-2}(s), \tag{B.8}$$

from which we obtain that

$$H(s) \triangleq \mathcal{L}\{h\}(s) = H_5(s) - j \cdot k_{\text{dir}} H_3(s) \tag{B.9}$$
$$= H_3(s)[\Gamma_1(s) - j \cdot k_{\text{dir}}] \tag{B.10}$$

The unilateral and the bilateral Laplace transform coincide for causal kernels. Furthermore, the bilateral transform, $\mathcal{B}h(s)$, and the FT $\hat{h}(\omega) = \mathcal{F}(h)$ are related according to

$$\mathcal{F}\{h\}(\omega) = \mathcal{B}\{h\}(s = j\omega). \tag{B.11}$$

We can then finally obtain the following expression for the FT of the temporal filter $h(t)$:

$$\hat{h}(\omega) = \hat{h}_3(\omega)[\hat{\gamma_1}(\omega) - j \cdot k_{\text{dir}}]. \tag{B.12}$$

B.3 Direction selectivity

In this section we will provide derivations and proof for the statements in Chapter 4.2.3 concerning the direction-selectivity properties of the model.

B.3.1 Power of the response to a moving grating

In order to compute $P_{\tilde{s}}$ we first need to evaluate $|\tilde{s}(t)|^2$. From eq. (4.18) we can derive

$$\begin{aligned}|\tilde{s}(t)|^2 =& \big[P_h(-\omega_0)P_g(\mathbf{k}_0) + P_h(\omega_0)P_g(-\mathbf{k}_0)\big] \cdot |\xi(\mathbf{x}_o, t)|^2 \\ &+ 2\text{Re}\big[\hat{h}(\omega_0)\,\overline{\hat{h}(-\omega_0)}\,\hat{g}(\mathbf{k}_0)\,\overline{\hat{g}(-\mathbf{k}_0)}\,\xi(\mathbf{x}_o, t)^2\big].\end{aligned}$$

The r.h.s. of the above expression consists of a constant offset, the first term, and a periodic signal with period $2\pi/\omega_0$, the second terms. When plugged in eq. (4.7), the net contribution of the second term to the integral is zero and only the constant offset remains. We can pull out of the integral all the remaining terms that are not depending on the temporal variable t, obtaining

$$P_{\tilde{s}} = \big[P_h(-\omega_0)P_g(\mathbf{k}_0) + P_h(\omega_0)P_g(-\mathbf{k}_0)\big] \cdot \frac{2\pi}{\omega} \int_{-\pi/\omega_0}^{+-\pi/\omega_0} |\xi(\mathbf{x}_o, t)|^2 \mathrm{d}t.$$

The integral is the power of the input signal $\xi(\mathbf{x}_o, t)$, P_ξ. Therefore we obtained the result reported in eq. (4.19).

B.3.2 Assymetry of the response

Let $\tilde{s}_{pos}(t)$ be the filter response to a moving grating $s_{pos}(\mathbf{x}, t)$ with spatial orientation $\mathbf{k}_{pos} = \mathbf{k}_{RF}$ and speed $\omega_{pos} = \omega_0$, and let $P_{pos} = \langle \tilde{s}_{pos} \rangle^2_{RMS}$ be

the power of the response. Similarly, we define P_{neg} as the power of the response to a second grating, $s_{neg}(\mathbf{x}, t)$, with the same spatial orientation $\mathbf{k}_{neg} = \mathbf{k}_{RF}$ but opposite speed, $\omega_{neg} = -\omega_0$. We want to know for which values of ω_0 the condition $\Delta P = P_{pos} - P_{neg} > 0$ is satisfied. We start by expanding P_{pos} and P_{neg} (we omit the constant term $A^2/4$, the effect of which is merely a constant rescaling and does not otherwise change the qualitative result):

$$P_{pos} = P_h(-\omega_0)P_g(\mathbf{k}_{RF}) + P_h(\omega_0)P_g(-\mathbf{k}_{RF})$$
$$P_{neg} = P_h(\omega_0)P_g(\mathbf{k}_{RF}) + P_h(-\omega_0)P_g(-\mathbf{k}_{RF}).$$

We then subtract the second from the first, obtaining:

$$\Delta P = -\underbrace{\big(P_h(\omega_0) - P_h(-\omega_0)\big)}_{\Delta P_h(\omega_0)} \cdot \underbrace{\big(P_g(\mathbf{k}_0) - P_h(-\mathbf{k}_0)\big)}_{\Delta P_g(\mathbf{k}_0)}$$

Therefore for ΔP to be positive, $\Delta P_g(\mathbf{k}_{RF})$ and $\Delta P_h(\omega_0)$ must have opposite signs. Since $\Delta P_g(\mathbf{k}_{RF})$ is negative[1], $\Delta P_h(\omega_0)$ must be positive to make ΔP positive. After expanding $\Delta P_h(\omega_0)$ (derivation in the next proof below), we get

$$\Delta P_h(\omega_0) = 8\, P_{h_3}(\omega_0)\frac{\alpha^3 k_{\text{dir}}\omega_0}{(\alpha^2+\omega_0^2)^2} > 0 \Leftrightarrow k_{\text{dir}}\omega_0 > 0.$$

From here, we can see that when $k_{\text{dir}} > 0$ a positive ω_0 results in a positive ΔP, i.e. the response s_{pos} is stronger than the response s_{neg}, meaning that $\angle\mathbf{k}_{RF}$ is the preferred direction.

Deriving the value of $\Delta P_h(\omega)$

We start by expanding the expression for the power of $\hat{h}(\omega)$:

$$\begin{aligned} P_h(\omega) \triangleq |\hat{h}(\omega)|^2 &= |\hat{h}_3(\omega)|^2 \cdot |\hat{\gamma}_1(\omega) - j \cdot k_{\text{dir}}|^2 \\ &= P_{h_3}(\omega)\big(P_{\gamma_1}(\omega) + k_{\text{dir}}^2 - 2k_{\text{dir}}\text{Im}[\gamma_1(\omega)]\big) \end{aligned} \tag{B.13}$$

Now, we consider the power at opposite temporal frequency, $P_h(-\omega)$:

$$P_h(-\omega) = P_{h_3}(-\omega)\big(P_{\gamma_1}(-\omega) + k_{\text{dir}}^2 - 2k_{\text{dir}}\text{Im}[\gamma_1(-\omega)]\big) \tag{B.14}$$
$$= P_{h_3}(\omega)\big(P_{\gamma_1}(\omega) + k_{\text{dir}}^2 + 2k_{\text{dir}}\text{Im}[\gamma_1(\omega)]\big), \tag{B.15}$$

where the second line follows from $h_3(\omega)$ and $\gamma_n(\omega)$ being Hermitian functions, since they are the FT of real-valued functions.[2] After subtracting the two expressions and rearranging the resulting terms, we obtain

$$P_h(\omega) - P_h(-\omega) = 8\, P_{h_3}(\omega)\frac{\alpha^3\omega k_{\text{dir}}}{(\alpha^2+\omega^2)^2}. \tag{B.16}$$

[1] $\Delta P_g(\mathbf{u}) = -2e^{-\mathbf{u}^\top\boldsymbol{\Sigma}\mathbf{u}}e^{-\mathbf{k}_{RF}^\top\boldsymbol{\Sigma}\mathbf{k}_{RF}}\sinh(2\,\mathbf{u}^\top\boldsymbol{\Sigma}\mathbf{k}_{RF})$, which is negative for $\mathbf{u} = \mathbf{k}_{RF}$.

[2] A Hermitian function $f(x)$ satisfies $f(x) = \overline{f(-x)}$.

Appendix C

Extending the receptive field model

This appendix proposes two possible extensions of the receptive field model. Since we concretely formalized these concepts after all analyses were performed, we considered it would be more appropriate to talk about this additional work on a dedicated appendix rather than in the may body of the thesis, in order to avoid any potential confusion for any discrepancy between the model presented in Chapter 4 and the model used in our analyses.

C.1 Motion opponency

The receptive field model discussed in Chapter 4 was used in Chapter 7 to characterize neural receptive fields in actual electrophysiological recordings of neural activity. This model is quite flexible and it can reproduce some of the most salient features often observed in V1 neurons (orientation and direction selectivity; linearity or phase invariance), but it is not an exhaustive model of the response properties of these neurons. More specifically, its capability to model direction selectivity is limited and it cannot entirely reproduce the direction selectivity properties often observed in V1. Experimental evidence suggests that some direction selective cells in V1 are sometimes inhibited by stimuli moving in the null direction[1, 41, 90, 102, 112], i.e. the opposite of the preferred direction. In the present model, however, a stimulus moving in the null direction always elicit a positive, albeit small, response (see Fig. 4.5). Denoting with θ_{pref} and $\theta_{null} = \theta_{pref} + 180°$ the preferred and the null direction respectively, one way to model motion opponency is by subtracting the response of an hypothetical "θ_{null}-selective" neuron from that of a "θ_{pref}-selective" one, in order to obtain a signal with strong excitation for motion along θ_{pref} and inhibition for motion in the opposite direction [1, 115]. This could be done by adding to our model a quadrature pair of receptive field filters with opposite selectivity. By mir-

roring the construction of the kernels (4.3), we obtain

$$w_3(\mathbf{x}, t) = h_5(t)g_c(\mathbf{x}) - k_{\text{dir}}h_3(t)g_s(\mathbf{x})$$
$$w_4(\mathbf{x}, t) = h_5(t)g_s(\mathbf{x}) + k_{\text{dir}}h_3(t)g_c(\mathbf{x}),$$

which are built from the same separable components (see Fig. 4.1). We need of course to adjust the quadratic nonlinearity to perform the required computation. Denoting with $\tilde{\mathbf{s}}_p = [\tilde{s}_1, \tilde{s}_2]$ the stacked response of w_1 and w_2, and with $\tilde{\mathbf{s}}_n = [\tilde{s}_3, \tilde{s}_4]$ the response of w_3 and w_4, we specify the following new quadratic function:

$$Q(\tilde{\mathbf{s}}_p, \tilde{\mathbf{s}}_n) \triangleq \mathbf{b}^\top \tilde{\mathbf{s}}_p + \tilde{\mathbf{s}}_p^\top \mathbf{C}\, \tilde{\mathbf{s}}_p - \tilde{\mathbf{s}}_n^\top \mathbf{D}\, \tilde{\mathbf{s}}_n \tag{C.2}$$

where $\mathbf{b}$ and $\mathbf{C}$ are defined as before and $\mathbf{D}$ is a 2×2 real matrix. In order to avoid singularities both $\mathbf{C}$ and $\mathbf{D}$ must be non-negative definite matrices. If that were not the case, one could change the sign of their respective eigenvalues and change the spatial filter orientation by 180° and the response would still be the same. This would introduce an undesired singular parameterization in our model. Note also that only $\tilde{\mathbf{s}}_p$ participates in the linear response, for an analogous reason. Imposing the non-negativity constraint would effectively restrict the feasible domain of the GLM parameters, therefore potentially limiting the application of the sampling scheme discussed in Chapter 5. In order to not sacrifice inference, we could consider a stricter parameterization, i.e. imposing that $\mathbf{D} = \mathbf{C}$ and introduce a new parameter $k_{\text{opp}} \in [0, 1]$ to control the degree of motion opponency expressed by the model. We must therefore adapt the quadratic non-linearity:

$$Q(\tilde{\mathbf{s}}_p, \tilde{\mathbf{s}}_n) \triangleq \mathbf{b}^\top \tilde{\mathbf{s}}_p + \tilde{\mathbf{s}}_p^\top \mathbf{C}\, \tilde{\mathbf{s}}_p - k_{\text{opp}} \cdot \tilde{\mathbf{s}}_n^\top \mathbf{C}\, \tilde{\mathbf{s}}_n. \tag{C.3}$$

This way, the model can still be treated as a 6-dimensional GLM, albeit with a different design matrix. If we expand and rearrange the terms in eq. (C.3), we obtain that a single row of the design matrix is

$$[1, \tilde{s}_1, \tilde{s}_2, \tilde{s}_1^2 - k_{\text{opp}}\, \tilde{s}_3^2, \tilde{s}_2^2 - k_{\text{opp}} \tilde{s}_4^2, 2\,(\tilde{s}_1 \tilde{s}_2 - k_{\text{opp}}\, \tilde{s}_3 \tilde{s}_4)].$$

C.2 Non-zero DC gain

An other potential limitation of the present model is the presence of a non-zero DC gain for constant stimuli. This is not an issue in itself, but in some cases this behavior may be undesirable. This is a direct consequence of parameterizing the spatial filters using Gabor wavelets: the non zero DC gain of a Gabor wavelet (see eq. (B.1)) implies that the response of the filter depends also on the overall average value of the input signal. In other words, the filter would respond differently to two otherwise identical signals which differ only by a constant offset in their intensities. This may not be the way a V1 neuron would respond in this scenario.

Solution 1

We may adjust the existing Gabor kernels by subtracting the corresponding offset

$$g'(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}\Sigma^{-1}\mathbf{x}\right)\left[\exp(-j(\mathbf{k}^\top\mathbf{x}-\phi)) - \hat{g}(\mathbf{0})\right], \tag{C.4}$$

where $\hat{g}(\mathbf{0})$ is the DC gain, obtain by evaluating the FT of the Gabor at the zero frequency. Adopting this new filter is equivalent to having an extra spatial filter consisting only of the envelope of the original Gabor kernel, scaled by the DC offset factor. This is the most economical solution in terms of changes to apply to the existing code base. Moreover, if the output of the extra spatial filter is explicitly computed, its value could be of used to normalize the response of the linear filter (i.e. to implement some kind of gain modulation mechanism [2, 40]), but we will not discuss this aspect in the current work.

Solution 2

Two dimensional Log-Gabor wavelets[19, 28] offer an alternative solution to the DC-gain problem. We propose a parameterization emphasizing the periodic nature of the orientation parameter:

$$\hat{g}_{\log}(\kappa,\theta) = \frac{\kappa_0}{\kappa}\exp\left(-\frac{\ln^2(\kappa/\kappa_0)}{2\ln^2(\sigma_f/\kappa_0)}\right)\exp\left(-\frac{1-\cos\theta}{2\sigma_\theta^2}\right), \tag{C.5}$$

where κ_0 is the center frequency, σ_f the width parameter for the frequency, θ_0 the center orientation, and σ_θ the width parameter of the orientation. The frequency and orientation bandwidths are encoded independently of each other, in contrast to a conventional Gabor parameterization[1], and are easily expressed in terms of other model parameters:

$$B_\kappa = 2\sqrt{\frac{2}{\ln 2}}\cdot\left|\ln\frac{\sigma_f}{\kappa_0}\right| \quad \text{and} \quad B_\theta = 2\sigma_\theta\sqrt{2\ln 2}.$$

Log-Gabor wavelets have been shown to better encode natural images than the original Gabor filter[28].

[1] In a conventional Gabor, the frequency bandwidth depends on $|\mathbf{k}|$ and σ_x, while the orientation bandwidth depends on σ_x and σ_y, therefore the two are entangled.

Appendix D

Signal-to-Noise Ratio for GLMs

In this appendix we discuss how the concept of signal-to-noise ratio (SNR) can be generalized to GLMs. Sections D.1 to D.3 provide a summary, based on the work in [18, 37]. The last section applies these concepts to the generative model presented in Chapter 4.

D.1 SNR in a linear system with additive Gaussian noise

We start by considering a linear system with additive Gaussian observation noise, defined by the following observation model:

$$y_k = \mathbf{x}_k^\top \boldsymbol{\beta} + \varepsilon_k, \quad \varepsilon_k \sim \mathcal{N}(0, \sigma_\epsilon^2), \tag{D.1}$$

where $\mathbf{x}_k = [1, x_1^{(k)}, \ldots, x_m^{(k)}]^\top$ is vector of fixed and known covariates, $\boldsymbol{\beta}$ is a parameter vector, and ε_k is the observation noise.

The SNR is defined as the ratio of the variance of the signal component to the variance of the noise, that is

$$\text{SNR} = \frac{\sigma_{signal}^2}{\sigma_{noise}^2}, \tag{D.2}$$

where σ_{signal}^2 is the variance explained by the linear predictor, and σ_{noise}^2 is the intrinsic variance due to noise. Assuming $\mathbb{E}[x_i] = 0$ (for $i = 1, \ldots, m$),

$$\sigma_{signal}^2 = \mathbb{E}_{\mathbf{x}}[(\mathbf{x}^\top \boldsymbol{\beta} - \beta_0)^2]. \tag{D.3}$$

Under this model, the variance of the noise can be expressed as the variance of the residuals $r_k = y_k - \mathbf{x}_k^\top \boldsymbol{\beta}$, meaning that σ_{noise}^2 can be interpreted as

the expected prediction error (EPE) if we use $\mathbf{x}_k^\top \boldsymbol{\beta}$ to predict the observed data point y_k:

$$\sigma^2_{noise} = \mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta}) = \mathbb{E}_{y|\mathbf{x}}[(y - \mathbf{x}^\top \boldsymbol{\beta})^2]. \tag{D.4}$$

Similarly, the total variance of the data can be defined as the EPE when predicting the value of y_k using its unconditional mean,

$$\sigma^2_{total} = \mathrm{EPE}(y, \beta_0) = \mathbb{E}_y[(y - \beta_0)^2], \tag{D.5}$$

where we assumed that all covariates have zero mean, so that $\mathbb{E}_y[y] = \beta_0$. Since $\sigma^2_{signal} = \sigma^2_{total} - \sigma^2_{noise}$, we can then express the SNR as

$$\mathrm{SNR} = \frac{\mathrm{EPE}(y, \beta_0) - \mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})}{\mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})}, \tag{D.6}$$

which is the reduction of the EPE after accounting for the signal, divided by the intrinsic EPE due to noise.

D.2 SNR for a GLM

Based on eq. (D.6), we can now extend the definition of SNR to a GLM system where the expected value of the observed variable is $\mathbb{E}[y] = \mu = g^{-1}(\mathbf{x}^\top \boldsymbol{\beta})$, and observation are distributed according to an exponential family with density $p(y|\mu) = f(y; \mu)$. We substitute the square-error EPE in eq. (D.6) with the KL-based EPE

$$\mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta}) = \mathbb{E}_{y|\mathbf{x}^\top \boldsymbol{\beta}}[-2 \ln f(y; \mu)], \tag{D.7}$$

The factor of two is included to make the KL-based EPE identical to the squared-error based EPE for Gaussian variables. Similarly, the unconditional prediction error is the expected error, in terms of negative log-likelihood loss, when predicting y using only its unconditional mean $\mu_0 = g^{-1}(\beta_0)$:

$$\mathrm{EPE}(y, \beta_0) = \mathbb{E}_{y|\mathbf{x}^\top \boldsymbol{\beta}}[-2 \ln f(y; \mu_0)]. \tag{D.8}$$

If we substitute (D.7) and (D.8) in (D.6), we obtain a KL-based expression for the SNR:

$$\mathrm{SNR} = \frac{\mathrm{EPE}(y, \beta_0) - \mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})}{\mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})} = -\frac{D_{\mathrm{KL}}(\mu \| \mu_0)}{\mathbb{E}_{y|\mathbf{x}^\top \boldsymbol{\beta}}[\ln f(y; \mu)]}, \tag{D.9}$$

where the numerator is the Kullback-Liebler divergence

$$D_{\mathrm{KL}}(\mu \| \mu_0) = \mathbb{E}_{y|\mathbf{x}^\top \boldsymbol{\beta}}\left[\ln \frac{f(y; \mu)}{f(y; \mu_0)} \right] \tag{D.10}$$

D.3 Partitioning the SNR

We now generalize the SNR to quantify the contribution of different covariates to the prediction of the observed variables. We start by assuming that linear predictor can be partitioned in two components as

$$\mathbf{x}^\top \boldsymbol{\beta} = \mathbf{x}_1^\top \boldsymbol{\beta}_1 + \mathbf{x}_2^\top \boldsymbol{\beta}_2. \tag{D.11}$$

If we consider the values of $\boldsymbol{\beta}$, $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ which minimize respectively $\mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})$, $\mathrm{EPE}(y, \mathbf{x}_1^\top \boldsymbol{\beta}_1)$ and $\mathrm{EPE}(y, \mathbf{x}_2^\top \boldsymbol{\beta}_2)$ and replace $\mathrm{EPE}(y, \beta_0)$ with $\mathrm{EPE}(y, \mathbf{x}_1^\top \boldsymbol{\beta}_2)$ in (D.9), we can define a SNR with respect to $\mathbf{x}_2$ while controlling for the effect of $\mathbf{x}_1$:

$$\mathrm{SNR}_2 = \frac{\mathrm{EPE}(y, \mathbf{x}_1^\top \boldsymbol{\beta}_1) - \mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})}{\mathrm{EPE}(y, \mathbf{x}^\top \boldsymbol{\beta})}. \tag{D.12}$$

The numerator is the reduction in EPE due to $\mathbf{x}_2^\top \boldsymbol{\beta}_2$ when controlling for the systematic changes in y attributed to $\mathbf{x}_1^\top \boldsymbol{\beta}_1$. The denominator is the the EPE due to noise, like in (D.6) and (D.9).

D.4 Application to the V1 generative model

The total SNR of the V1 generative model presented in Chapter 4 is

$$\mathrm{SNR}_L = \frac{\mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{C}\tilde{\mathbf{s}}) - \mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b} + \tilde{\mathbf{s}}^\top \mathrm{C}\tilde{\mathbf{s}})}{\mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b} + \tilde{\mathbf{s}}^\top \mathrm{C}\tilde{\mathbf{s}})}, \tag{D.13}$$

where the numerator is the reduction in EPE after accounting for the contribution of the stimulus to the total firing rate of the neuron, and the denominator is the intrinsic EPE due to the noise in the spike generation process. The linear SNR is defined as

$$\mathrm{SNR}_L = \frac{\mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{C}\tilde{\mathbf{s}}) - \mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b} + \tilde{\mathbf{s}}^\top \mathrm{C}\tilde{\mathbf{s}})}{\mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b} + \tilde{\mathbf{s}}^\top \mathrm{C}\tilde{\mathbf{s}})}, \tag{D.14}$$

where the numerator measures the net contribution of the stimulus after controlling for the effect of the quadratic term and the offset. Similarly, we defined the quadratic SNR as

$$\mathrm{SNR}_Q = \frac{\mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b}) - \mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b} + \tilde{\mathbf{s}}^\top \mathrm{C}\tilde{\mathbf{s}})}{\mathrm{EPE}(y, a + \tilde{\mathbf{s}}^\top \mathbf{b} + \tilde{\mathbf{s}}^\top \mathrm{C}\tilde{\mathbf{s}})}, \tag{D.15}$$

where the numerator measures the net contribution of the stimulus after controlling for the effect of the linear term and the offset.

Appendix E

Electrophysiological recordings

This appendix contains the details concerning the dataset of electrophysiological recordings analyzed in Chapter 7. This dataset was collected in the Department of Behavior and Brain Organization lead by Dr. Jason Kerr at the Max Planck Institute for Neurobiology of Behavior – caesar, Bonn, Germany. The data was kindly made available for this study as part of a collaboration.

Contributions The data was acquired by Dr. Takashi Handa and Dr. Carl Holmgren, who both also curated the spike sorting and performed an initial screening of the data. Experiments were supervised Dr. Damian Wallace.

Experimental setup Neural activity was recorded by means of juxtacellular recordings in layer 2/3 of awake, head-fixed rats. Animals were placed in front of a flat screen at a distance of 30 cm. The screen size was 106×60cm. The screen therefore covered a visual field portion of $121° \times 63°$ (horizontally and vertically, respectively). The stimulus was repeatedly presented to the animals, either binocularly or covering alternatively the left or the right eye. This monocular stimulation protocol, however, was performed only for a subset of all cells.

Visual stimulus The visual stimulus consisted of 5 minutes of spatially and temporally correlated Gaussian noise, displayed at 30Hz, for a total of 9000 frames. The stimulus contrast was temporally modulated at a frequency of 10 Hz [73]. The frame size was 120×68 pixels (columns, rows), corresponding to a resolution of for a vertical resolution of 0.93 and a horizontal resolutions of 1.01 degrees/pixels.

Appendix F

Supplementary figures and tables

model	param stat	xo	yo	freq	hsize	vsize
lin	avg	1.474e-03	1.474e-03	1.474e-03	1.474e-03	1.474e-03
	sd	1.474e-03	1.474e-03	1.474e-03	1.474e-03	1.474e-03
sqr	avg	1.474e-03	1.474e-03	5.772e-03	1.474e-03	2.366e-03
	sd	1.474e-03	1.474e-03	1.474e-03	1.474e-03	2.977e-03

Table F.1: Wilcoxon rank-sum test results.

model	param stat	xo	yo	freq	hsize	vsize
lin	avg	1.598e-04	1.306e-05	9.226e-04	7.847e-08	8.674e-09
	sd	2.226e-05	2.153e-04	2.324e-05	2.105e-09	3.915e-05
sqr	avg	1.688e-04	5.543e-06	1.062e-02	1.137e-04	1.651e-04
	sd	8.676e-05	8.018e-05	3.067e-04	1.609e-06	2.632e-03

Table F.2: Student's t test for paired samples

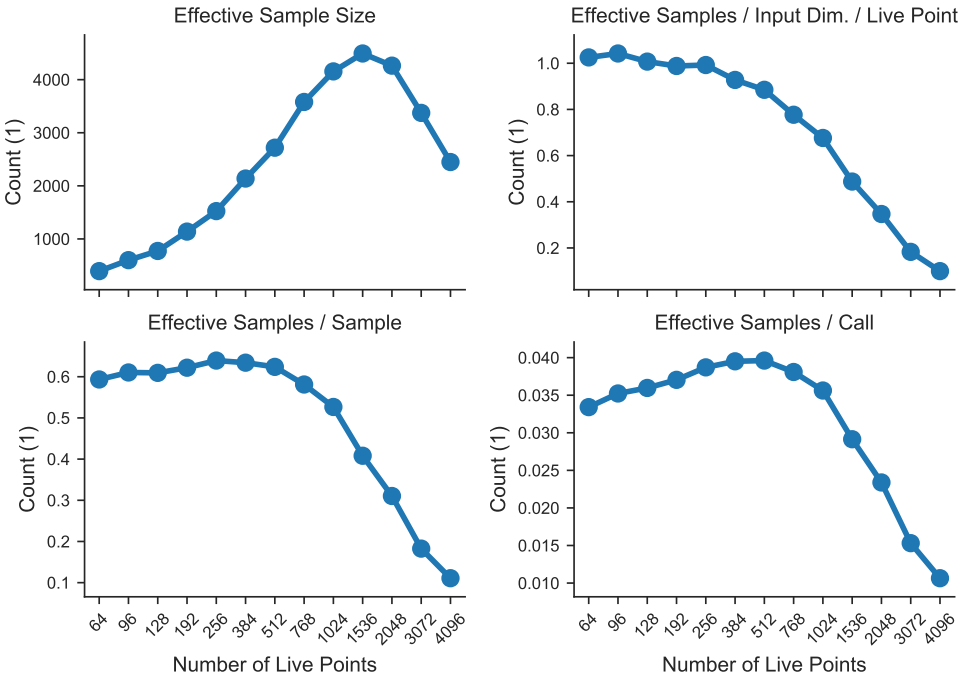

Fig. F.1: **Sampling efficiency on a toy model.** We repeated the same analysis represented in Fig. 5.1, but on a toy problem represented by a 6-dimensional Gaussian (for details see main text). The performance pattern is qualitatively similar to the one observed in the actual problem. **A)** Effective sample size as a function of N_{live}. **B)** Effective samples per live point per input dimension. **C)** Average number of effective samples generated by each actual sample. **D)** Average number of effective samples generated for each evaluation of the log-likelihood function. CNS performs better than ONS on all fronts, except A.

Duration	$P < 0.05$	$P < 0.25$	$0.25 \leq P \leq 0.75$	$P > 0.75$	$P > 0.95$
1 min	4	4	10	3	59
2 min	0	1	1	3	75
4 min	1	1	0	0	78

Table F.3: **Details on RF detection – Supplement to Fig. 7.2A.** This table reports the number of cells corresponding to each class.

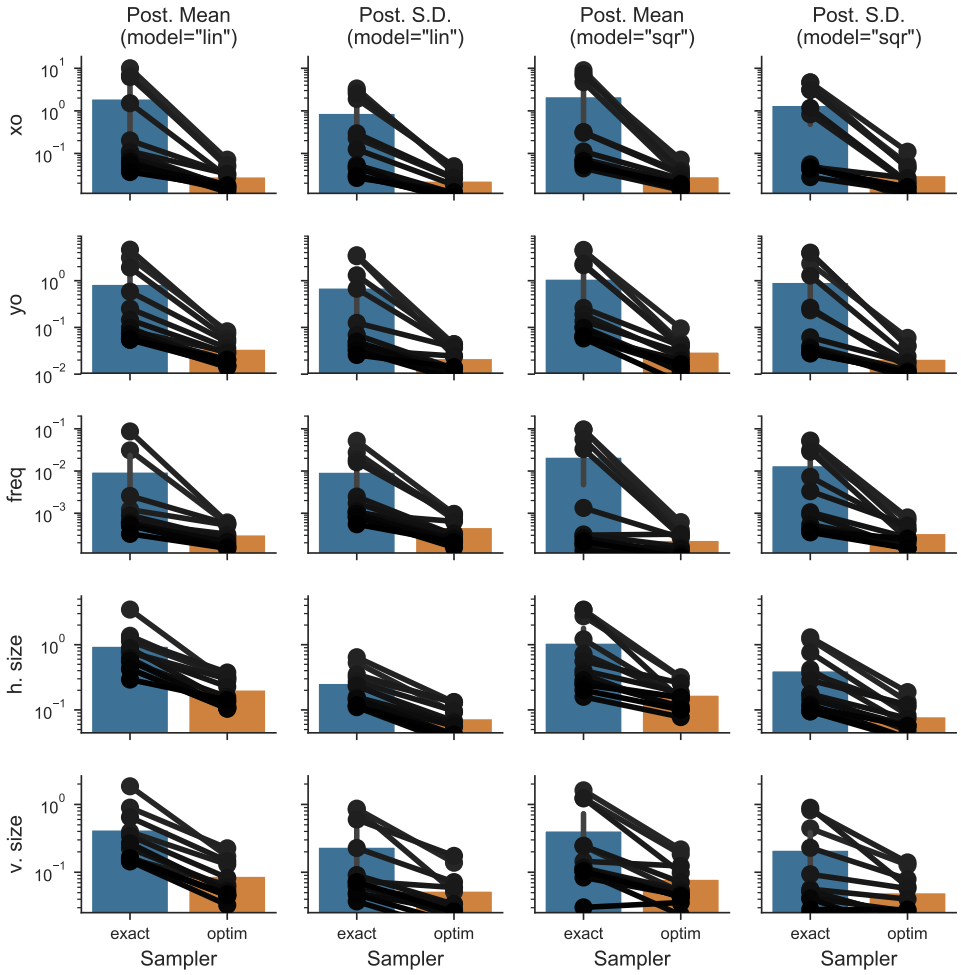

Fig. F.2: **Supplement 1 to Fig. 5.3.** Variability of the estimated posterior means and standard deviations under the two models here considered (linear and quadratic output); Each row correspond to one RF parameter, each column to one statistic. Variability of the basic sampler is reported in blue and the one of the collapsed sampler in orange. Notice the log scaling of the y axis.

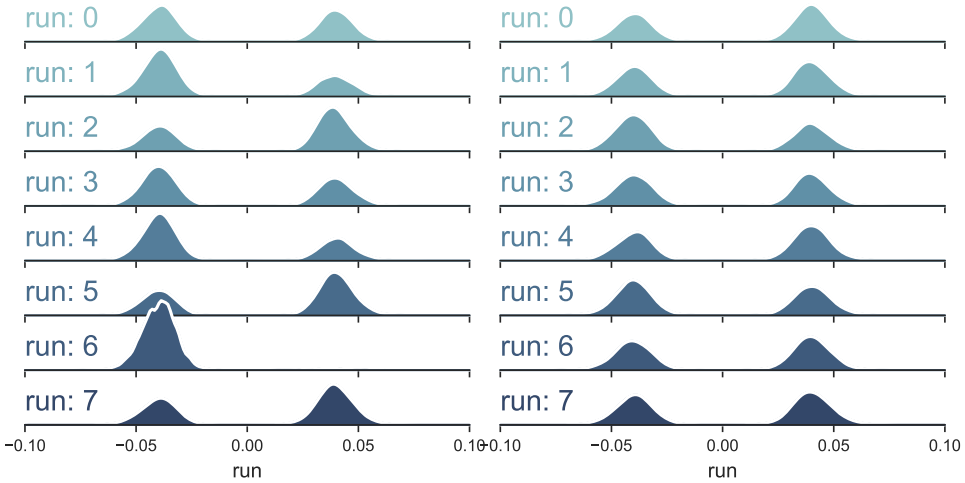

Fig. F.3: **Supplement 2 to Fig. 5.3.** Variability of the estimated posterior means and standard deviations under the two models here considered (linear and quadratic output); Each row correspond to one RF parameter, each column to one statistic. Variability of the basic sampler is reported in blue and the one of the collapsed sampler in orange. Notice the log scaling of the y axis.

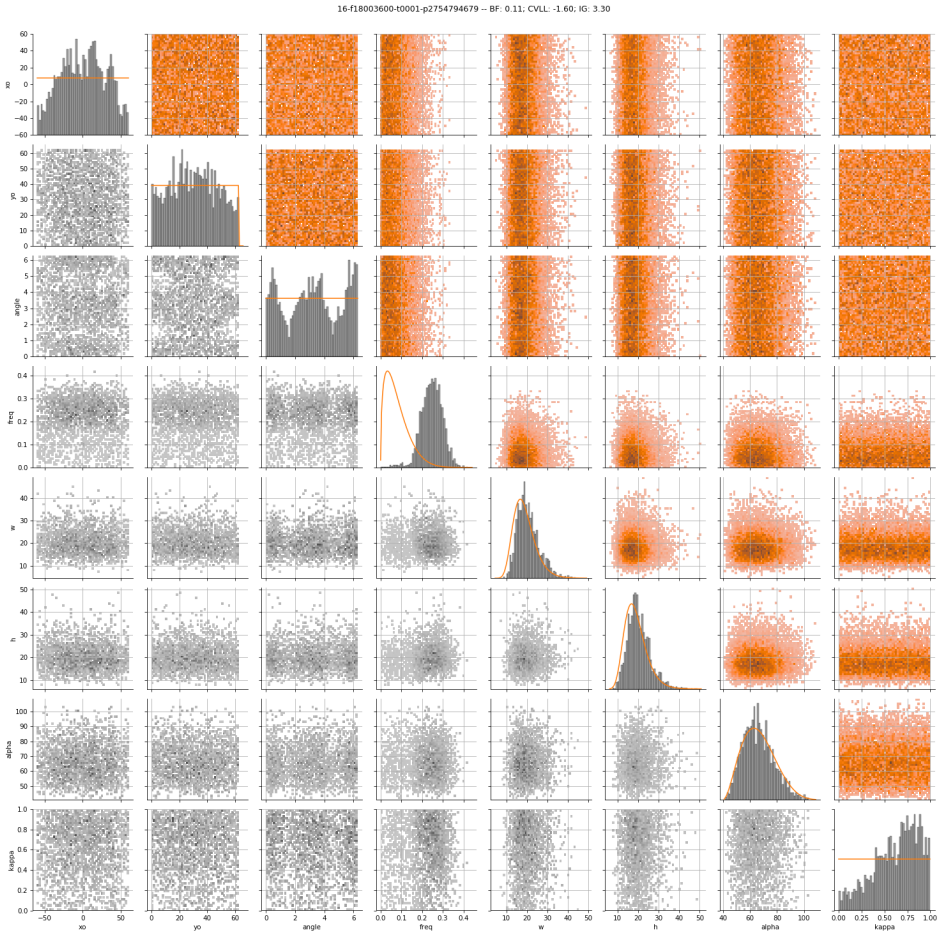

Fig. F.4: **Posterior marginals: false detection.** Full posterior distribution corresponding to 1 minute of noise data wrongly classified as containing a receptive field.

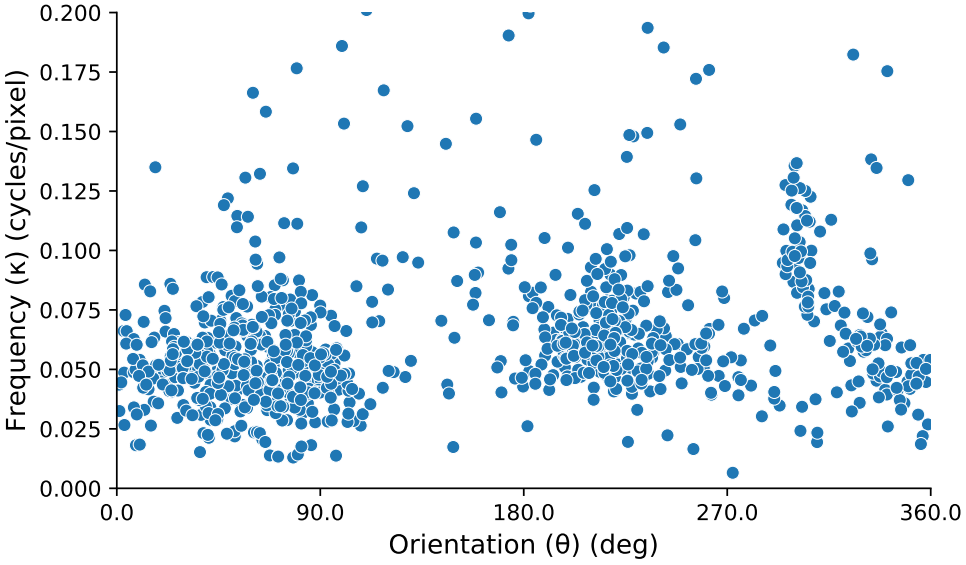

Fig. F.5: **Orientation-frequency joint marginal.** Data from a randomly chosen simulation.

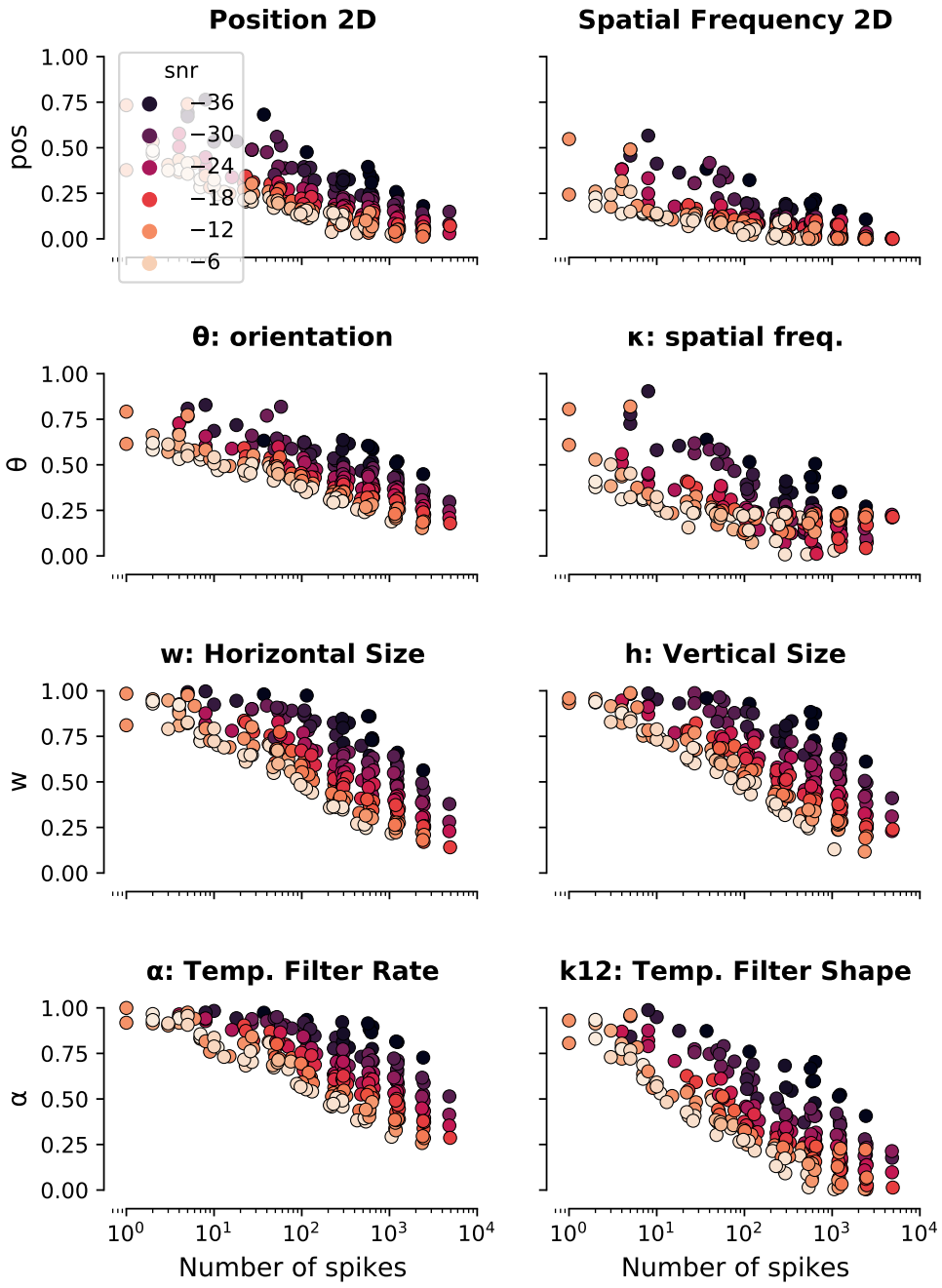

Fig. F.6: **Per-Parameter RU vs observed spike count.**

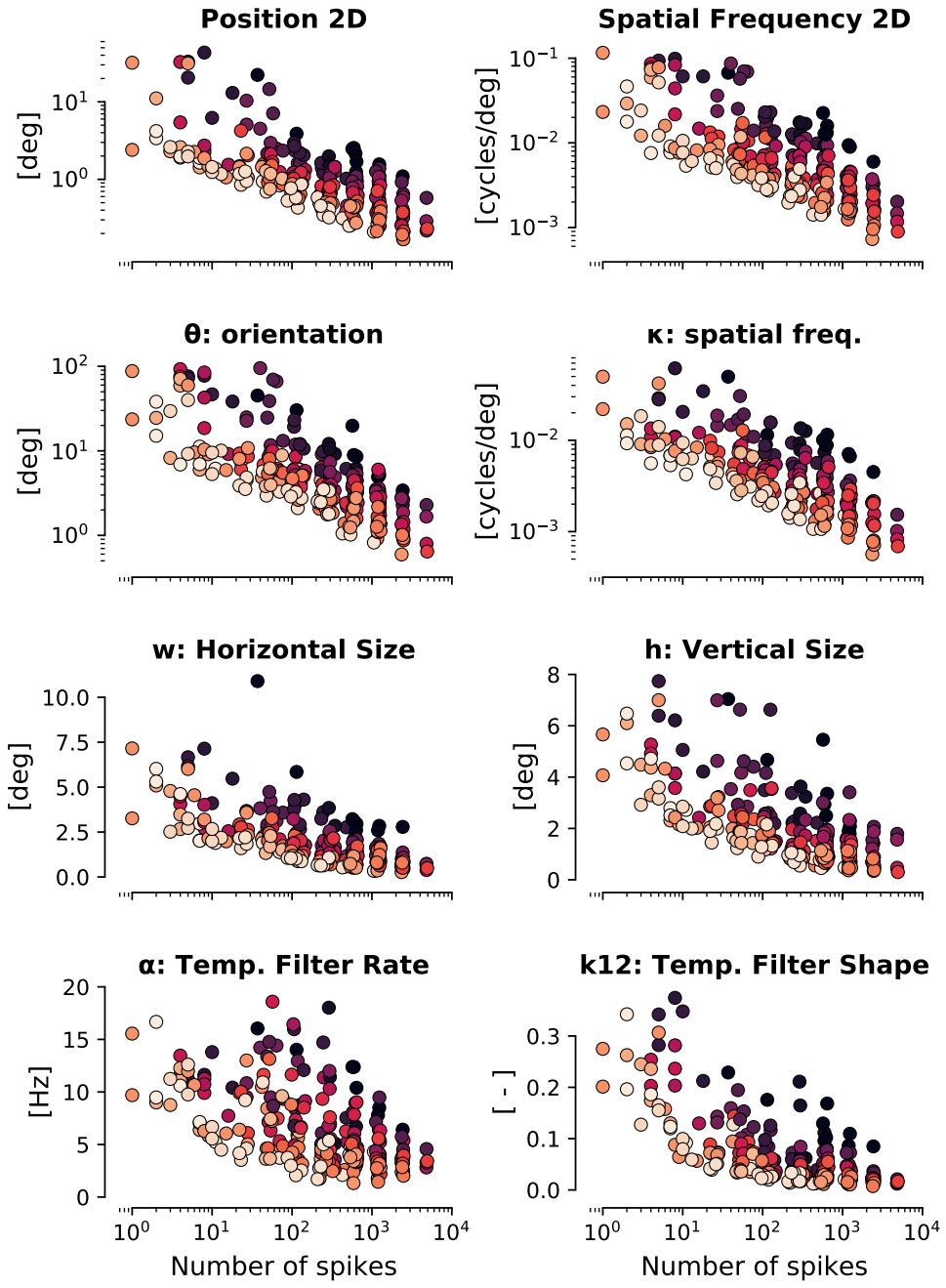

Fig. F.7: **Per-Parameter RMSE vs observed spike counts.**

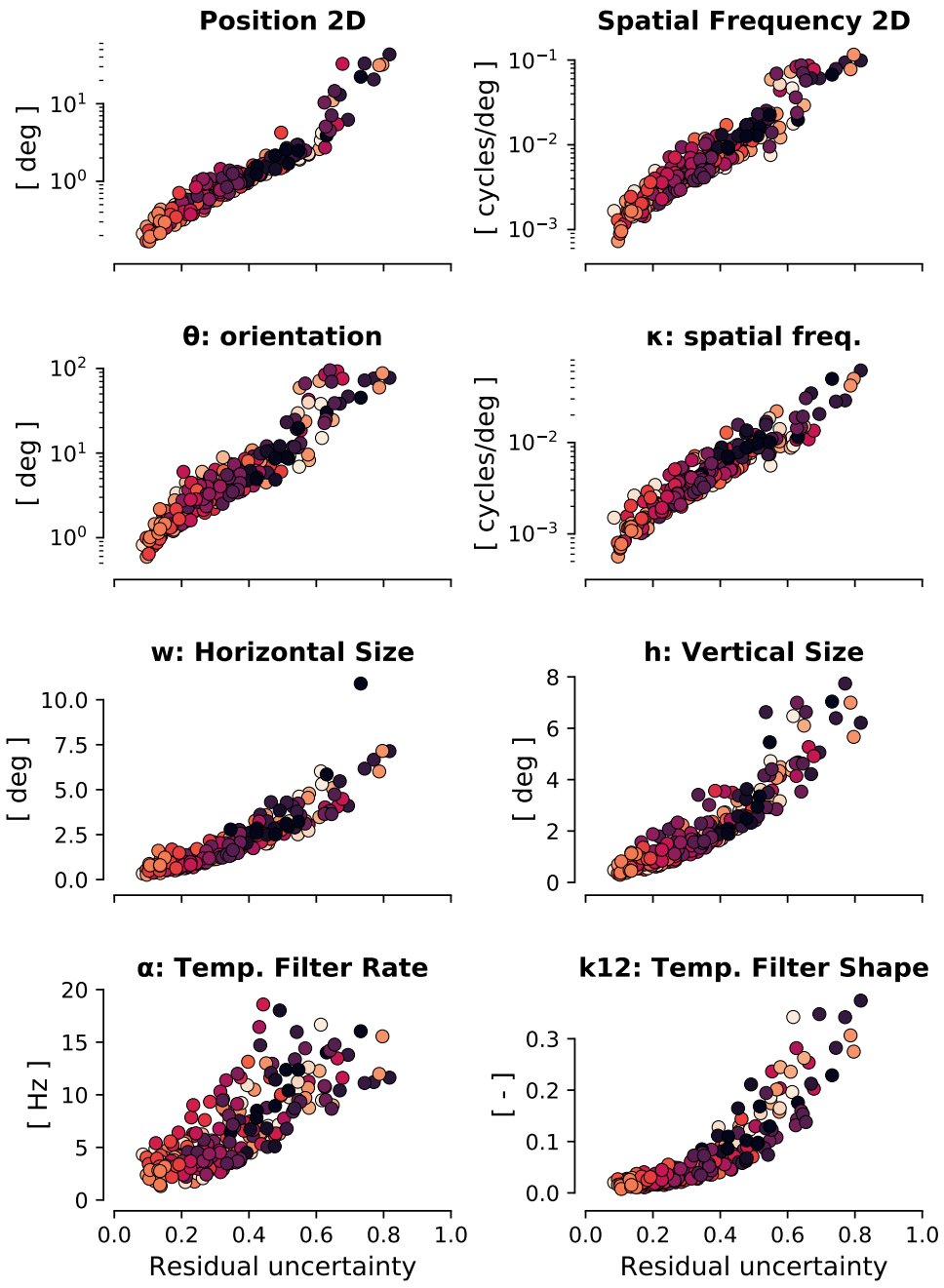

Fig. F.8: **Per-Parameter RMSE vs Total RU.**

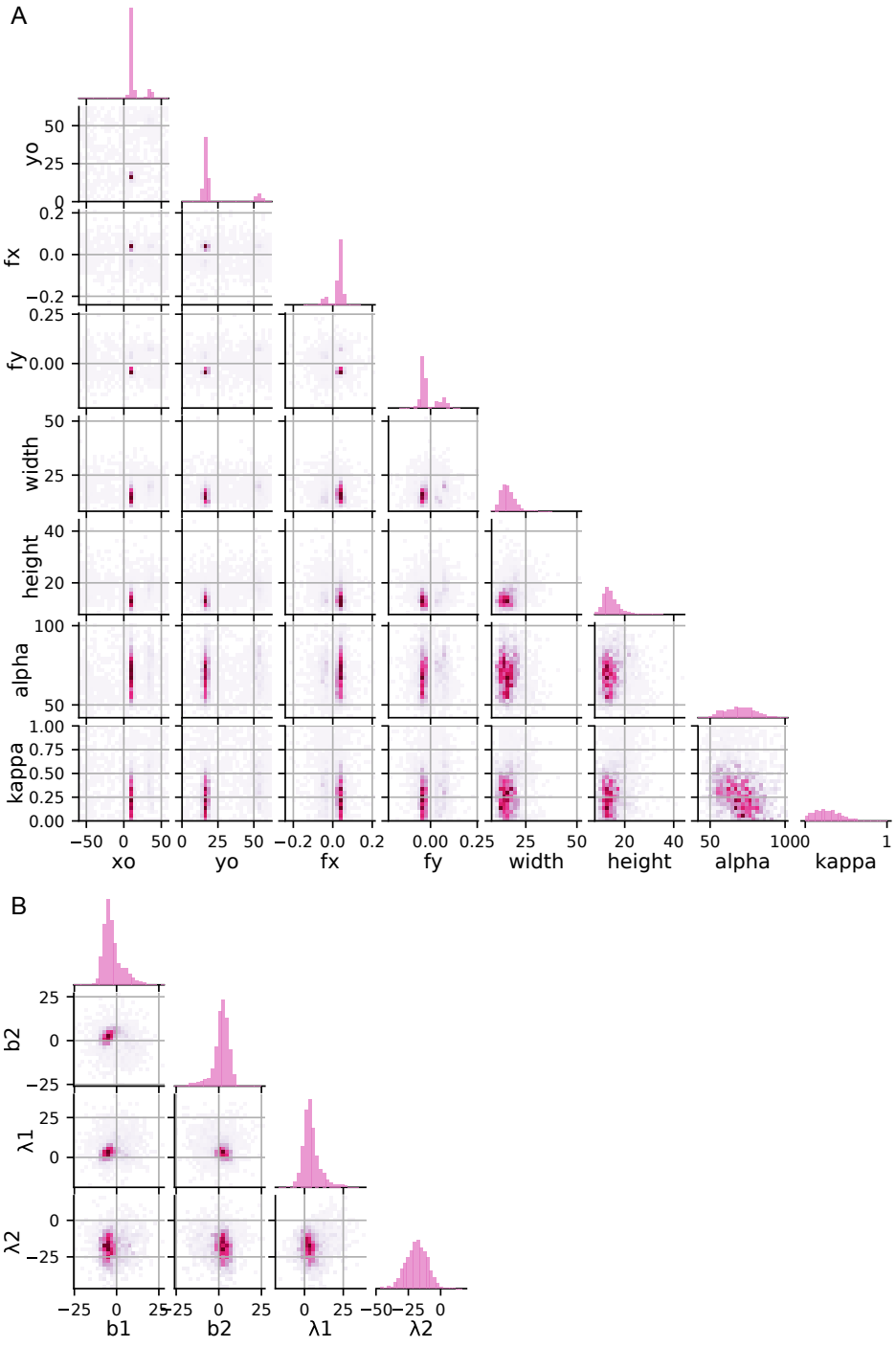

Fig. F.9: **Marginal posterior distributions using 1 minute of data.** Example cell used in Chapter 7.2, posterior of the QD model. **A)** Receptive field parameters. **B)** Non-linearity parameters.

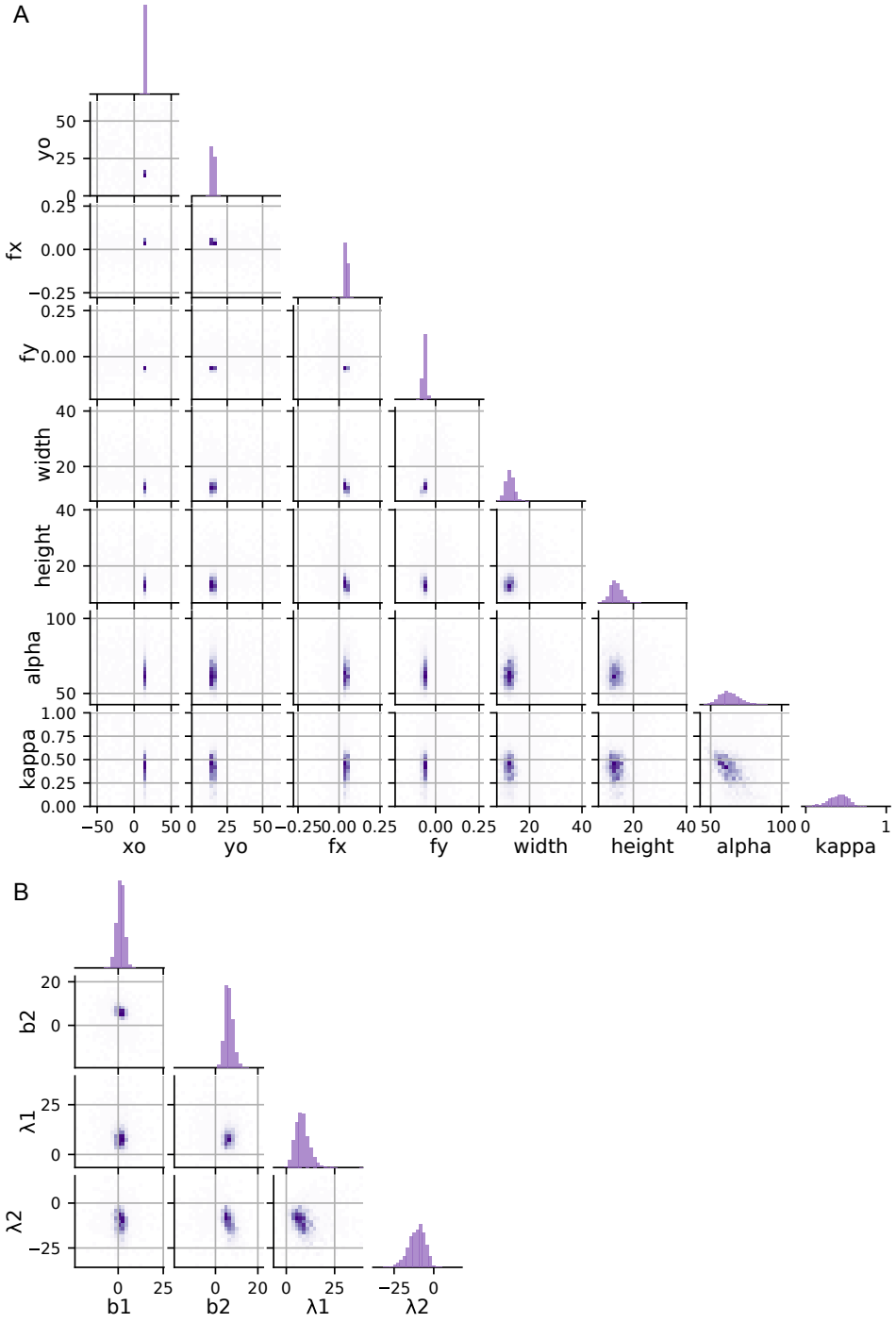

Fig. F.10: **Marginal posterior distributions using 4 minute of data.** Example cell used in Chapter 7.2, posterior of the QD model. **A)** Receptive field parameters. **B)** Non-linearity parameters.

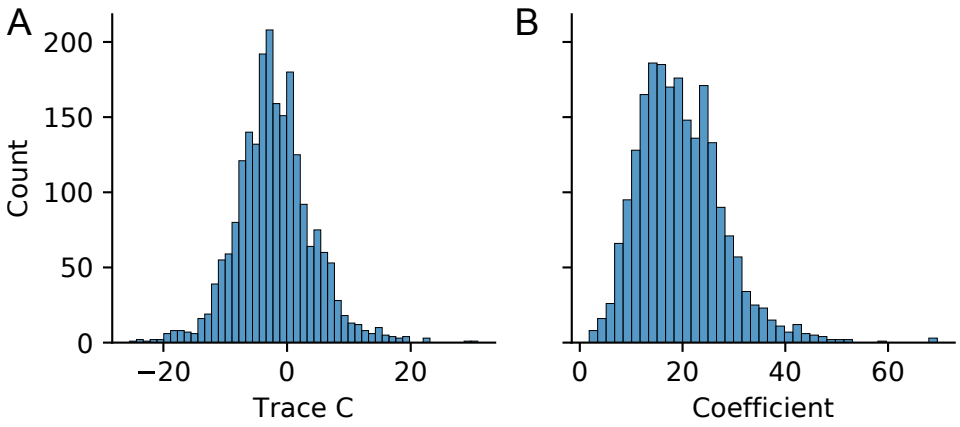

Fig. F.11: **Properties of the quadratic term. A)** Trace of **C**. **B)** Magnitude of $\dot{C}$. For the definitions, see eq. (4.23).

Bibliography

[1] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. Journal of the Optical Society of America A, 2(2):284–299, 1985.

[2] D. G. Albrecht and W. S. Geisler. Motion selectivity and the contrast-response function of simple cells in the visual cortex. Vis. Neurosci., 7(6):531–46, 1991.

[3] P. Alper. A consideration of the discrete Volterra series. IEEE Transactions on Automatic Control, 10(3):322–327, 1965.

[4] E. B. Andersen. Sufficiency and exponential families for discrete sample spaces. Journal of the American Statistical Association, 65(331), 1970.

[5] M. C. Aoi and J. W. Pillow. Scalable Bayesian inference for high-dimensional neural receptive fields. bioRxiv, 2017.

[6] G. Ashton, N. Bernstein, J. Buchner, X. Chen, G. Csányi, A. Fowlie, F. Feroz, M. Griffiths, W. Handley, M. Habeck, E. Higson, M. Hobson, A. Lasenby, D. Parkinson, L. B. Pártay, M. Pitkin, D. Schneider, J. S. Speagle, L. South, J. Veitch, P. Wacker, D. J. Wales, and D. Yallup. Nested sampling for physical scientists. Nature Reviews Methods Primers, 2(1):39, 2022.

[7] R. Ben-Yishai, R. L. Bar-Or, and H. Sompolinsky. Theory of orientation tuning in visual cortex. Proceedings of the National Academy of Sciences, 92(9):3844–3848, 1995.

[8] C. M. Bishop and N. M. Nasrabadi. Pattern Recognition and Machine Learning. Springer, Berlin, Heidelberg, 2006.

[9] E. N. Brown, R. E. Kass, and P. P. Mitra. Multiple neural spike train data analysis: state-of-the-art and future challenges. Nature Neuroscience, 7:456–461, 2004.

[10] J. Buchner. Nested sampling methods. arXiv preprint arXiv:2101.09675, 2021.

[11] M. Carandini. Amplification of trial-to-trial response variability by neurons in visual cortex. PLoS Biology, 2, 2004.

[12] M. Carandini. What simple and complex cells compute. Journal of Physiology, 577(Pt 2):463–6, 2006.

[13] G. Casella, C. P. Robert, and M. T. Wells. Generalized accept-reject sampling schemes. Lecture Notes-Monograph Series, 45:342–347, 2004.

[14] F. S. Chance, S. B. Nelson, and L. F. Abbott. Complex cells as cortically amplified simple cells. Nature neuroscience, 2(3):277–282, 1999.

[15] E. J. Chichilnisky. A simple white noise analysis of neuronal light responses. Network, 12(2):199–213, 2001.

[16] N. Chopin and C. P. Robert. Properties of nested sampling. Biometrika, 97(3):741–755, 2010.

[17] B. Cronin, I. H. Stevenson, M. Sur, and K. P. Körding. Hierarchical Bayesian modeling and Markov chain Monte Carlo sampling for tuning-curve analysis. Journal of Neurophysiology, 103(1):591–602, 2010.

[18] G. Czanner, S. V. Sarma, D. Ba, U. T. Eden, W. Wu, E. Eskandar, H. H. Lim, S. Temereanca, W. A. Suzuki, and E. N. Brown. Measuring the signal-to-noise ratio of a neuron. Proceedings of the National Academy of Sciences, 112(23):7141–7146, 2015.

[19] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. Journal of the Optical Society of America A, 2(7):1160–1169, 1985.

[20] P. Dayan and L. F. Abbott. Theoretical Neuroscience: Computational and Mathematical M The MIT Press, 2005.

[21] N. G. de Bruijn. Asymptotic Methods in Analysis. Bibliotheca mathematica. Dover Publications, 1981.

[22] G. C. DeAngelis, I. Ohzawa, and R. D. Freeman. Receptive-field dynamics in the central visual pathways. Trends in Neurosciences, 18(10):451–458, 1995.

[23] F. M. Dekking, C. Kraaikamp, H. P. Lopuhaä, and L. E. Meester. A modern introduction to probability and statistics: understanding why and how, volume 488 of Springer Texts in Statistics. Springer, 2005.

[24] T. J. Dodd and C. J. Harris. Identification of non-linear time series via kernels. International Journal of Systems Science, 33(9):737–750, 2002.

[25] R. C. Emerson, M. C. Citron, W. J. Vaughn, and S. A. Klein. Nonlinear directionally selective subunits in complex cells of cat striate cortex. Journal of Neurophysiology, 58(1):33–65, 1987.

[26] F. Feroz and M. P. Hobson. Multimodal nested sampling: an efficient and robust alternative to Markov Chain Monte Carlo methods for astronomical data analyses. Monthly Notices of the Royal Astronomical Society, 384(2):449–463, 2008.

[27] F. Feroz, M. P. Hobson, and M. Bridges. MultiNest: an efficient and robust Bayesian inference tool for cosmology and particle physics. Monthly Notices of the Royal Astronomical Society, 398(4):1601–1614, 2009.

[28] DJ Field. Relations between the statistics of natural images and the response properties of cortical cells. Journal of the Optical Society of America A, 4(12):2379–2394, 1987.

[29] I. Fogel and D. Sagi. Gabor filters as texture discriminator. Biological Cybernetics., 61(2):103–113, 1989.

[30] D. Gabor. Theory of communication. Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering, 93:429–441, 1946.

[31] S. Geman and S. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 6:721–741, 1984.

[32] S. Gerwinn, J. H. Macke, and M. Bethge. Bayesian inference for generalized linear models for spiking neurons. Frontiers in Computational Neuroscience, 4(May):12, 2010.

[33] P. I. Good and J. W. Hardin. Common Errors in Statistics (and How to Avoid Them). EBL-Schweitzer. Wiley, 2012.

[34] S. N. Goodman. Toward evidence-based medical statistics. 1: The p value fallacy. Annals of Internal Medicine, 130(12):995–1004, 1999. PMID: 10383371.

[35] S. N. Goodman. Toward evidence-based medical statistics. 2: The Bayes factor. Annals of Internal Medicine, 130(12):1005–1013, 1999. PMID: 10383350.

[36] W. J. Handley, M. P. Hobson, and A. N. Lasenby. polychord: next-generation nested sampling. Monthly Notices of the Royal Astronomical Society, 453(4):4385–4399, 2015.

[37] T. Hastie. A closer look at the deviance. The American Statistician, 41(1):16–20, 1987.

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

[39] D. Heeger. Nonlinear model of neural responses in cat visual cortex, pages 119–133. MIT Press, 1991.

[40] D. J. Heeger. Normalization of cell responses in cat striate cortex. Visual Neuroscience, 9(2):181–197, 1992.

[41] D. J. Heeger, G. M. Boynton, J. B. Demb, E. Seidemann, and W. T. Newsome. Motion opponency in visual cortex. Journal of Neuroscience, 19(16):7162–7174, 1999.

[42] E. Higson, W. Handley, M. Hobson, and A. Lasenby. Sampling errors in nested sampling parameter estimation. Bayesian Analysis, 13(3):873–896, 2018.

[43] J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky. Bayesian model averaging: a tutorial. Statistical Science, 14(4):382 – 417, 1999.

[44] Z. Huang, Y. Ran, J. Oesterle, T. Euler, and P. Berens. Estimating smooth and sparse neural receptive fields with a flexible spline basis. Neurons, Behavior, Data analysis and Theory, 5, 2021.

[45] D. H. Hubel and T. N. Wiesel. Receptive fields of single neurones in the cat's striate cortex. Journal of Physiology, 148(3):574–91, 1959.

[46] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. Journal of Physiology, 160(1):106–154, 1962.

[47] D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. Journal of Physiology, 195(1):215–243, 1968.

[48] H. Jeffreys. The Theory of Probability. Oxford Classic Texts in the Physical Sciences. OUP Oxford, 1998.

[49] J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in car striate cortex. Journal of Neurophysiology, 58(6):1233–58, 1987.

[50] J. H. Kaas and C. E. Collins. The Primate Visual System. CRC Press, 2004.

[51] J. W. Kalat. Biological Psychology. Cengage Learning, 2015.

[52] P. Kara, P. Reinagel, and R. C. Reid. Low response variability in simultaneously recorded retinal, thalamic, and cortical neurons. Neuron, 27(3):635–646, 2000.

[53] R. E. Kass and A. E. Raftery. Bayes factors. Journal of the American Statistical Association, 90(430):773–795, 1995.

[54] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[55] L. Kish. Survey Sampling. A Wiley Interscience Publication. Wiley, 1965.

[56] T. W. Kjaer, T. J. Gawne, J. A. Hertz, and B. J. Richmond. Insensitivity of V1 complex cell responses to small shifts in the retinal image of complex patterns. Journal of Neurophysiology, 78(6):3187–3197, 1997. PMID: 9405538.

[57] B. O. Koopman. On distributions admitting a sufficient statistic. Transactions of the American Mathematical Society, 39(3), 1936.

[58] K. P. Körding, C. Kayser, W. Einhäuser, and P. König. How are complex cell properties adapted to the statistics of natural stimuli? Journal of Neurophysiology, 91(1):206–212, 2004. PMID: 12904330.

[59] S. Kullback. Information Theory and Statistics. Courier Corporation, 1997.

[60] S. Kullback and R. A. Leibler. On Information and Sufficiency. The Annals of Mathematical Statistics, 22(1):79 – 86, 1951.

[61] D. Lopez-Paz and M. Oquab. Revisiting classifier two-sample tests. arXiv preprint arXiv:1610.06545, 2016.

[62] A. Ly, A. Stefan, J. van Doorn, F. Dablander, D. van den Bergh, A. Sarafoglou, Š. Kucharský, K. Derks, Q. F. Gronau, A. Raj, U. Boehm, E. J. van Kesteren, M. Hinne, D. Matzke, M. Marsman, and E. J. Wagenmakers. The Bayesian methodology of Sir Harold Jeffreys as a practical alternative to the p value hypothesis test. Compututatoinal Brain & Behavior, 3(2):153–161, 2020.

[63] C. K. Machens, M. S. Wehr, and A. M. Zador. Linearity of cortical receptive fields measured with natural sounds. Journal of Neuroscience, 24(5):1089–1100, 2004.

[64] D.J.C. MacKay. Information Theory, Inference and Learning Algorithms. Cambridge University Press, 2003.

[65] M. Mazurek, M. Kager, and S. D. Van Hooser. Robust quantification of orientation selectivity and direction selectivity. Frontiers in Neural Circuits, 8, 2014.

[66] P. McCullagh and J.A. Nelder. Generalized Linear Models, Second Edition. Chapman and Hall/CRC Monographs on Statistics and Applied Probability Series. Chapman & Hall, 1989.

[67] A. F. Meyer, J. P. Diepenbrock, M. F. K. Happel, F. W. Ohl, and Jörn Anemüller. Discriminative learning of receptive fields from responses to non-gaussian stimulus ensembles. PLoS One, 9(4):e93062, 2014.

[68] A. F. Meyer, R. S. Williamson, J. F. Linden, and M. Sahani. Models of neuronal stimulus-response functions: elaboration, estimation, and evaluation. Frontiers in Systems Neuroscience, 10:109, 2017.

[69] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst. Receptive field organization of complex cells in the cat's striate cortex. Journal of Physiology, 283(1):79–99, 1978.

[70] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst. Spatial summation in the receptive fields of simple cells in the cat's striate cortex. Journal of Physiology, 283:53–77, 1978.

[71] R. M. Neal. Slice sampling. The Annals of Statistics, 31(3):705 – 767, 2003.

[72] J. A. Nelder and R. W. M. Wedderburn. Generalized linear models. Journal of the Royal Statistical Society. Series A (General), 135(3):370–384, 1972.

[73] C. M. Niell and M. P. Stryker. Highly selective receptive fields in mouse visual cortex. Journal of Neuroscience, 28(30):7520–7536, 2008.

[74] A. V. Oppenheim, R. W. Schafer, and J. R. Buck. Discrete-time signal processing. Prentice Hall, 1999.

[75] L. Paninski. Convergence properties of some spike-triggered analysis techniques. Advances in neural information processing systems, 15, 2002.

[76] L. Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. Network: Computation in Neural Systems, 15(4):243–62, 2004.

[77] I. M. Park, E. W. Archer, N. Priebe, and J. W. Pillow. Spectral methods for neural characterization using generalized quadratic models. In Advances in Neural Information Processing Systems, volume 26. Curran Associates, Inc., 2013.

[78] I. M. Park and J. W. Pillow. Bayesian spike-triggered covariance analysis. In Advances in Neural Information Processing Systems, volume 24. Curran Associates, Inc., 2011.

[79] M. Park and J. W. Pillow. Receptive field inference with localized priors. PLoS Computational Biology, 7(10), 2011.

[80] M. Park and J. W Pillow. Bayesian inference for low rank spatiotemporal neural receptive fields. In Advances in Neural Information Processing Systems, volume 26. Curran Associates, Inc., 2013.

[81] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.

[82] W. D. Penny, J. Mattout, and N. Trujillo-Barreto. Bayesian model selection and averaging. In K. Friston, J. Ashburner, S. Kiebel, T. Nichols, and W. Penny, editors, Statistical Parametric Mapping, chapter 35, pages 454–467. Academic Press, London, 2007.

[83] J. W. Pillow, L. Paninski, V. J. Uzzell, E. P. Simoncelli, and E. J. Chichilnisky. Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model. Journal of Neuroscience, 25(47), 2005.

[84] J. W. Pillow, J. Shlens, L. Paninski, A. Sher, A. M. Litke, E. J. Chichilnisky, and E. P. Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. Nature, 454(7207):995–9, 2008.

[85] J. W. Pillow and E. P. Simoncelli. Dimensionality reduction in neural models: An information-theoretic generalization of spike-triggered average and covariance analysis. Journal of Vision, 6(4):9–9, 2006.

[86] E. J. G. Pitman. Sufficient statistics and intrinsic accuracy. Mathematical Proceedings of the Cambridge Philosophical Society, 32(4), 1936.

[87] D. A. Pollen and S. F. Ronner. Phase relationships between adjacent simple cells in the visual cortex. Science, 212(4501):1409–1411, 1981.

[88] D. A. Pollen and S. F. Ronner. Visual cortical neurons as localized spatial frequency filters. IEEE Transactions on Systems, Man, and Cybernetics, SMC-13(5):907–916, 1983.

[89] S. J. D. Prince, A. D. Pointon, B. G. Cumming, and A. J. Parker. Quantitative analysis of the responses of V1 neurons to horizontal disparity in dynamic random-dot stereograms. Journal of Neurophysiology, 87(1):191–208, 2002.

[90] N. Qian, R. A. Andersen, and E. H. Adelson. Transparent motion perception as detection of unbalanced motion signals. III. Modeling. Journal of Neuroscience, 14(12):7381–7392, 1994.

[91] A.G. Ramakrishnan, S. Kumar Raja, and H.V. Raghu Ram. Neural network-based segmentation of textures using Gabor features. In Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing, pages 365–374, 2002.

[92] C. E. Rasmussen and C. K. I. Williams. Gaussian Processes for Machine Learning. Adaptive Computation and Machine Learning series. MIT Press, 2005.

[93] D. L. Ringach. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. Journal of Neurophysiology, 88(1):455–463, 2002.

[94] D. Rose. An analysis of the variability of unit activity in the cat's visual cortex. Experimental Brain Research, 37(3), 1979.

[95] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review, 65(6):386, 1958.

[96] M. Sahani and J. Linden. Evidence optimization techniques for estimating stimulus-response functions. In S. Becker, S. Thrun, and K. Obermayer, editors, Advances in Neural Information Processing Systems, volume 15. MIT Press, 2002.

[97] P. H. Schiller, B. L. Finlay, and S. F. Volman. Quantitative studies of single-cell properties in monkey striate cortex. III. Spatial frequency. Journal of Neurophysiology, 39(6):1334–1351, 1976. PMID: 825623.

[98] O. Schwartz, J. W. Pillow, N. C. Rust, and E. P. Simoncelli. Spike-triggered neural characterization. Journal of Vision, 6(4):13–13, 2006.

[99] T. Sharpee, N. C. Rust, and W. Bialek. Analyzing neural responses to natural signals: Maximally informative dimensions. Neural Computation, 16(2):223–250, 2004.

[100] T. O. Sharpee. Computational identification of receptive fields. Annual Review of Neuroscience, 36(1):103–120, 2013.

[101] E. Simoncelli, J. W. Pillow, L. Paninski, and O. Schwartz. Characterization of neural responses with stochastic stimuli, pages 327–338. MIT Press, 2004.

[102] E. P. Simoncelli and D. J. Heeger. A model of neuronal responses in visual area MT. Vision Research, 38(5):743–761, 1998.

[103] J. Skilling. Nested sampling. In AIP conference proceedings, volume 735, pages 395–405. American Institute of Physics, 2004.

[104] J. Skilling. Nested sampling for general bayesian computation. Bayesian analysis, 1(4):833–859, 2006.

[105] J. Skilling. Nested sampling's convergence. In AIP Conference Proceedings, volume 1193, pages 277–291. American Institute of Physics, 2009.

[106] D. L. Snyder and M. I. Miller. Poisson processes, pages 41–112. Springer New York, New York, NY, 1991.

[107] J. S. Speagle. dynesty: a dynamic nested sampling package for estimating Bayesian posteriors and evidences. Monthly Notices of the Royal Astronomical Society, 493(3):3132–3158, 2020.

[108] M. E. Tipping. Sparse Bayesian learning and the relevance vector machine. Journal of machine learning research, 1(Jun):211–244, 2001.

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

[110] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. Journal of Neurophysiology, 93(2):1074–1089, 2005. PMID: 15356183.

[111] D. A. van Dyk and T. Park. Partially collapsed Gibbs samplers. Journal of the American Statistical Association, 103(482):790–796, 2008.

[112] J. P. H. van Santen and G. Sperling. Temporal covariance model of human motion perception. Journal of the Optical Society of America A, 1(5):451–473, 1984.

[113] A. Venkataraman and . V. Oppenheim. Signal approximation using the bilinear transform. In 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, pages 3729–3732, 2008.

[114] V. Volterra, L. Fantappiè, and M. Long. Theory of Functionals and of Integral and Integro-differential Equations. Blackie & Son Limited, 1930.

[115] A. B. Watson and A. J. Ahumada. Model of human visual-motion sensing. Journal of the Optical Society of America A, 2(2):322–342, 1985.

[116] R. S. Williamson, M. Sahani, and J. W. Pillow. The equivalence of information-theoretic and likelihood-based methods for neural dimensionality reduction. PLoS Computational Biology, 11(4):1–31, 2015.