IRCAM

ATIAM INTERNSHIP

Predicting Neural Responses to Music and Speech with Structured Neural Networks

---

Student:
Alexis ADONIADIS

Supervisor:
Sam NORMAN-HAIGNERE

Internship performed

in

Laboratoire des Systèmes Perceptifs (LSP)
Ecole Normale Superieure

August 20, 2018

IRCAM

Abstract

Sorbonne Université - Télécom PariTech
Ecole Normale Superieure

Master ATIAM

Predicting Neural Responses to Music and Speech with Structured Neural Networks

by Alexis ADONIADIS

Modeling and predicting the neural activity achieved in the auditory brain is a main concern in the field of auditory neuroscience. Our main goal, which is a current challenge, was to create a computational model which can predict neural activity in response to natural stimuli. Much of the computational work has been focusing on simple stimuli such as tones and synthetic noise. Here we try to understand the neural computation applied to real life sounds. To do so, neural spiking activity from the auditory cortex of passively listening ferrets was used to train our models and test their predictive accuracies.
We have investigated the simple linear model (STRF), which is widely used in the domain. And have extended this model, based on the hypothesis that nearby neurons share computation. A common low dimensional linear subspace was created and trained to fit the spiking data. Each neuron's activity was described by a weighted sum of a set of STRF kernels. The low dimensional subspace discovered let us outperform the predictive accuracy of the STRF model, and find interesting properties of the subspace.
We have also investigated a linear transformation which reduces the dimensionality while smoothing the data (GPFA). The improvement in predictive performance based on this transformation wasn't observed when returning to the initial spike subspace. Finally, an attempt to beat actual performances by adapting a deep convolutional network, tailored for the prediction of neural activity from the visual cortex, was achieved.

Acknowledgements

I would first like to thank Daniel Pressnitzer who introduced me to the discovery of this whole new field of research and science. Manu, for his passion for machine learning and his gift for communication. Giovanni, for his everyday joy. The two Thomas's, Sara, Garance and Felix for their friendship as interns and our common exploits. I would like to thank all Ircam, Sorbonne Université and Télécom PariTech researchers who have filled our brains with wonderful knowledge. Not forgetting all my Atiam friends who have made this year even greater.
I would also like to thank Stephen, who I have never met physically, but who provided all the data and who was of great help any time needed, in record time.
Finally, Sam, who was there everyday for me, who has an astonishing scientific mind, who taught me a great amount of useful knowledge and skills, and who took advantage of our beautiful French mountains as often as he could. Thank you for this invaluable experience.

Contents

List of Figures

Introduction

Sensory neuroscience is in need of computational models that can predict brain activity in response to a wide range of stimuli. Natural stimuli (eg : wind, speech and music in the auditory domain) are of great interest both because they are complex and varied, and because sensory systems are plausibly adapted to represent them [Simoncelli and Olshausen, 2001]. We lack good models for how the sensory system encodes natural stimuli, particularly in the auditory domain. Where most of the prior work is based on responses to simple tones and synthetic noise stimuli.
Obtaining a computational model with good predictive accuracy opens up many possibilities. For example, one can perform an unlimited number of synthetic experiments via simulation. Predictions from these simulations can then be tested with actual experiments, which are often invasive and time consuming, and which can only be used to test a small number of specific hypotheses. Computational models can also be used to perform experiments that would be impossible in the lab. For example, given a computational model one can synthesize sounds that should yield a strong response as a way to investigate the properties of that response [Freeman and Simoncelli, 2011, Norman-Haignere and McDermott, 2018]. A related creative example of application, as it has been investigated in vision using deep generative networks (DGN) and deep neural networks (DNN) [Nguyen et al., 2016], is the synthesis of totally novel sounds highly activating a set of desired neurons.

Lastly, computational models can yield insights into the computational function of a sensory system. For example, many hearing organisms species are able to recognize various common sounds across a wide range of variation in the acoustic environment (e.g due to reverberation, background noise, source direction and intensity). Presumably, the auditory system has been adapted to capture these invariances and computational models that can predict responses to natural sounds provide one way to understand how these invariances are neurally implemented.
Most existing computational models are neuron-specific: distinct instances of the model are used to predict the activity of each neuron. The Spectro Temporal Receptive Field (STRF) is one of the most commonly used neuron specific model in the auditory domain. Neural responses are predicted by linearly filtering a time-frequency representation of the stimulus, similar to a spectrogram. Physiology recordings and natural sounds were used during this project, further described in the Data recording section.
Responses from many neurons can often be predicted by a small number of components response patterns [Cunningham and Byron, 2014]. Here we focus on linear subspaces which provide a set of basis functions which can be combined via weighted sums to approximate neural activity. Linear subspaces have been shown to capture much of the behaviorally relevant information in the neural population. For example work on visual cortex has shown that linear decoders, which operate on the neural subspace, are sufficient to predict complex behaviors to a wide range of natural stimuli [Hung et al., 2005,Hong et al., 2016].

A second key property of sensory responses is that they are often highly nonlinear with respect to the stimulus, particularly in the cortex [Kozlov and Gentner, 2016, Sahani and Linden, 2003, Atencio, Sharpee, and Schreiner, 2008]. Our goal is thus to develop low-dimensional representations of sound that can be used to predict neural responses to many neurons, and which are nonlinear with respect to the input stimulus. This is a challenging problem, and we have made two steps along these lines. First, we developed a low-dimensional extension of the standard single-neuron STRF model (which is mostly linear). Second, we have inferred a low-dimensional representation from neural responses to natural sounds, and tested how much of this low-dimensional space can be explained by our low-dimensional STRF model. During our discussion, we suggest how the STRF model can be extended to account for more nonlinear response properties.

Data recording

We initially had an opportunity to collect recordings from human subjects. Electrocorticography (EcOG) data precisely. This type of recording is made by placing a grid of electrodes at the surface of the cortex. It is similar to electroencephalography (EEG), technology wise. The main difference being that opening the skull is necessary to place the electrodes. Hence, a better sensitivity is obtained as no bone reduces electrical conductivity. This technique is used to monitor, locate and remove epileptogenic zones of human patients with severe epilepsy. Some patients can then perform experimental tasks such as actively listening to natural sounds during the recording of their auditory cortex.

Not being able to fetch this type of data, we decided to work with single unit, microelectrode array recordings on ferrets, also passively listening to natural sounds. Our audition team at ENS are experts in ferret data collection, but our data was provided by Stephen David from the Oregon Hearing Research Center. A long surgery, a craniotomy, is made on the ferrets to place a headpost and a cap under sterile conditions, permitting head-fixation and access to auditory brain areas. The probe includes 64 electrodes that all fit in $1mm$ of height and 85μm of width (Figure 1). Their size are small enough to theoretically measure single neurons' activity in a ferret. Additional processing is needed to ensure single unit recordings : spike sorting. It performs segregation of the signals from single unit as opposed to multi unit, using clustering methods from the polytrode sorting software.

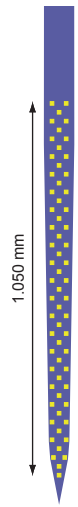

FIGURE 1: The probe inserted in the ferrets brains

The choice of the ferret is shared across many laboratories studying auditory neuroscience. It is big enough to have properties similar to the human and it is far more docile than monkeys when performing experiments.
The ferret was trained to passively listen to all the natural auditory stimuli selected. These contain ferret sounds, environmental sounds and human speech. All stimuli last 3 seconds with a 2 second silence transition. 288 train sounds were presented once. While 18 test sounds, different from the train stimuli, were presented 10 times each. The whole length of the recording was 35 minutes. During all that time, the spiking activity of the neurons is recorded and binned at 100Hz, resulting in a Post Stimulus Time Histogram (PSTH). We have worked on 4 different recordings, three were made on an unique ferret but in various zones of the auditory cortex (A1) and the fourth recording in another ferret, also in A1. The hole from the craniotomy is let to regrow after enough recordings have been made from it.

Chapter 1

Investigating the subspace for auditory neural computation

The data we have been working on was single unit recordings made in the auditory cortex (A1) of ferrets passively listening to natural sounds. A detailed description of the data can be found in the Data Recording section.

1.1 The initial Spectro-Temporal Receptive Field model

Many computational models exist for predicting neural information. One of the simplest and most standard is the Spectro Temporal Receptive Field (STRF) model. It is similar to the impulse response in the acoustic field, the difference being that it is bi-dimensional and that it models a neuron's spiking activity. Convolving the time frequency representation of the input stimulus with a 2D impulse response (the STRF) provides the model's prediction of the time-varying spike rate (convolution is performed across the time axis, but not frequency). The goal is to predict the peristimulus time histogram (PSTH), which provides an estimate of the neuron's spiking activity, in response to stimulus s, in fixed time bins (here we use 10ms).

For neuron n, stimulus s, time t and frequency f :

$$Stimulus_s(f,t) * STRF_n(f,t) = PSTH_{s,n}(t)$$

One benefit of this representation is that it is easily interpretable. Plotting the image of the STRF shows the time-frequency region the neuron is sensitive to. In addition, most of the auditory brain areas have a tonotopic mapping [Romani, Williamson, and Kaufman, 1982]. This refers to the fact that neurons are spatially distributed in brain areas in relation to their tuning to specific frequency bands. Neurons located in a same spatial area will have similar frequency tuning. This is a property which is used, in combination with others, to determine the function region where the probe was inserted in the brain of the animal.
The STRF also has the property of being relatively simple to fit, since it is a linear model. In the simplest case, the loss function is convex which obviates any worry about local optima. A visual example of a STRF can be seen in figure 1.1.

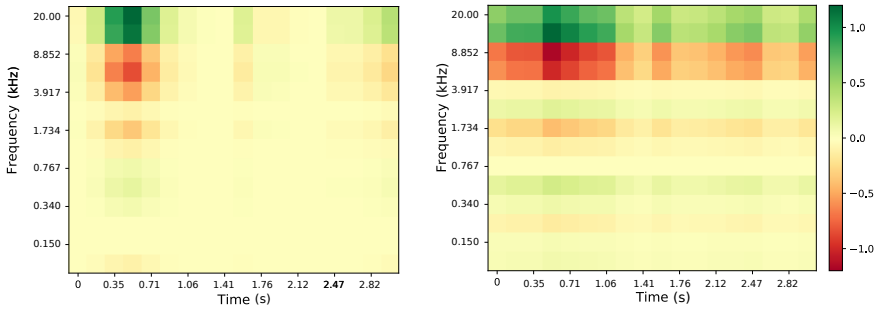

FIGURE 1.1: A STRF example fit on spiking data from cochleograms // 2 recurring kernel examples from the 5 kernel model

A widely used extension of this model is the Generalized Linear Model [Calabrese et al., 2011]. GLMs contain a linear model followed by an output non linearity. For example, a rectified linear unit (ReLU) can be used to set negative activations to zero, which is useful when modeling spike rates which are strictly positive. For simplicity, here we refer to STRFs with an output nonlinearity as STRFs rather than noting the nonlinearity every time.

The first step in STRF fitting was to convert the auditory stimuli into "cochleagrams": a spectrogram-like time frequency representation where physiological properties of the cochlea are modeled [Slaney, 1998]. The frequency scale is logarithmic (approximately mimicking how frequency is mapped along the basilar membrane of the cochlea), and cochlear amplification of low-amplitude sounds is modeled with a compressive nonlinearity. We used estimates of cochlear bandwidths in ferrets to compute the cochleagrams to account for the relatively coarse frequency resolution of the ferret cochlea compared with humans [Alves-Pinto et al., 2016].

Our implementation of the STRF model was made with Keras and Tensorflow, using a simple one layer convolutional neural network. Once the network was trained, we correlated the measured and predicted spike rates for the test data. As a sanity check of our code, we compared our prediction accuracy scores with existing code for STRF fitting that has been engineered to produce good predictions by placing constraints on the properties of the STRFs (we refer to this as the fine-tuned model, from [Thorson, Liénard, and David, 2015], see figure 1.2).

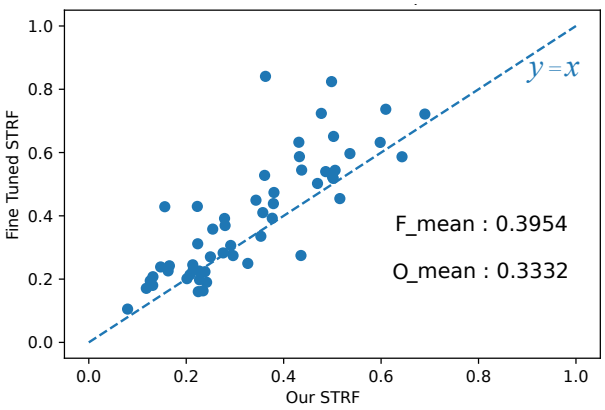

FIGURE 1.2: Test correlation comparison of two STRF models : the fine tuned vs ours

As we can see on this figure, our STRF predictions are comparable to those for the fine-tuned model . Our code yields test correlations that are only 16% lower than this model. We didn't go into deep tuning of this model as the goal was to demonstrate that we could fit STRFs to our recordings using the optimization package provided by TensorFlow (since this allowed us to explore subspace models as discussed below).

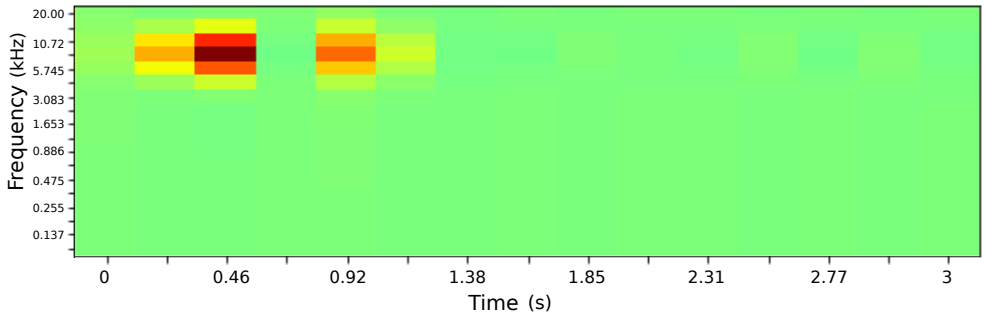

FIGURE 1.3: Best predictive STRF from the fine tuned model

Figure 1.3 shows the STRF corresponding to the best-predicted neuron (0.84 test correlation), as determined by the fine-tuned model. Most of the neurons from the recordings we have obtained are high-frequency tuned, perhaps because the recordings were made in the high-frequency region of the tonotopic map.

1.2 A shared STRF subspace

In the initial model implemented, a STRF is fit separately for each neuron with no constraints on the similarity of responses across neurons.
In this model, we use the property that neurons located in the same brain area share common computation. This key idea enables the benefit of using the potential of large numbers of neurons recorded at once to constrain the models learned for each neuron. Modern recording techniques, such as two photon imaging [Stosiek et al., 2003] promise the ability of recording thousands of neurons at once. General dimensionality reduction techniques over this data have been studied to capture the variance shared across neurons [Cunningham and Byron, 2014].
Klindt et al., 2017 use this idea and fit a multi layered CNN with a shared subspace to predict a large number of neurons' responses to images, in the visual cortex. We decided to pursue a variant of this idea here.

The visual description of the shared subspace is shown in figure 1.4 : A number S of STRFs are fit during the training of the whole set of N neurons. Instead of each neuron having a set of best frequency bands and time bins, each neuron now has a set of best STRFs, which we refer to as kernels. By constraining the number of component kernels to be small, we force the dimensionality of neural responses to be low and their responses to be similar. A re-weighting layer (S by N matrix in figure 1.4) is inserted in the neural network, which linearly maps the S STRFs to the N neurons. During training, each cochleogram of the training stimulus set was presented a number of times depending on the number of iterations and the batch size.

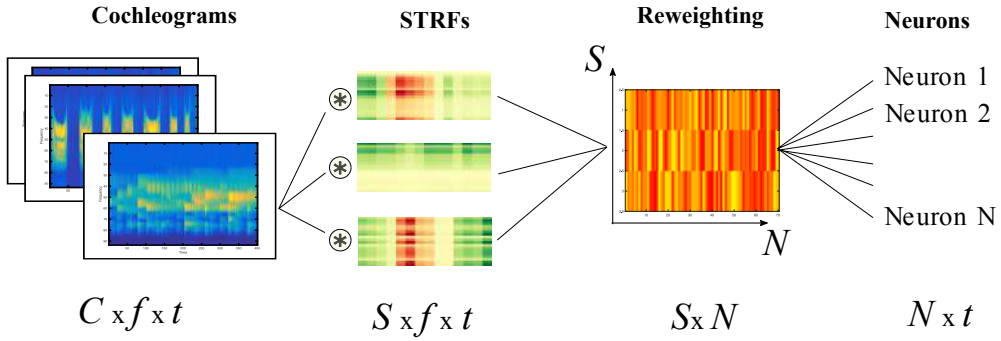

FIGURE 1.4: Shared Kernel STRF model's principle - Predicting neural data with cochleograms

$$Neuron_{n,c}(t) = \sum_{s=1}^{S} Reweight(s,n) \times [Kernel_s(f,t) \circledast Cochleogram_c(f,t)]$$

Tuning the model

The first question we wanted to answer was how many kernels yielded the best prediction accuracy. .
Based on the whole data set, we first trained our model to find the optimal set of the entire hyper parameters, except from S, the number of kernels which was set to arbitrary values and searched alone afterwards. The combination of hyper parameters which maximized our test correlation were the following :

Learning rate	Weight initialization	Batch size	Number of iterations	Early stopping
0.01	0.01	10	500	0

With the following sizes :

Cochleograms size	Kernels size	Number of neurons
18 x 300	18 x 15	70

We then fixed the found set of hyper parameters and trained our model iteratively with 1 to 30 intermediate kernels, comparing each trainings' test performances. In figure 1.5, we can see that an optimal number of 5 kernels was found.

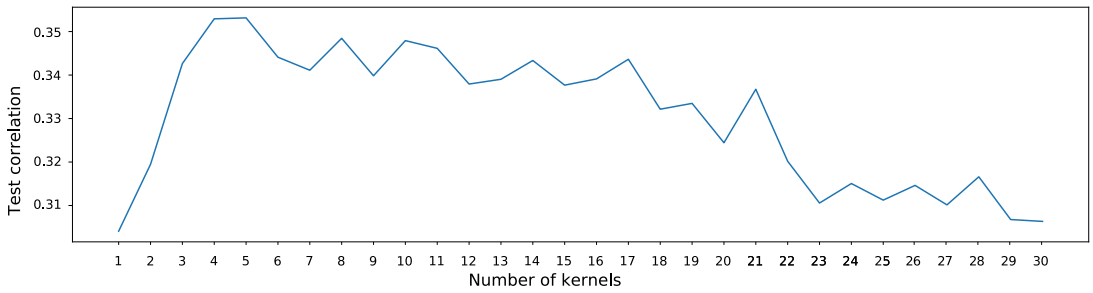

FIGURE 1.5: Finding the optimal number of kernels based on test correlation

After moderately fine tuning the hyper parameters based on 5 kernels, we were able to beat the STRF model fit with both our code (which is arguably the more fair comparison) and with the fine-tuned model :

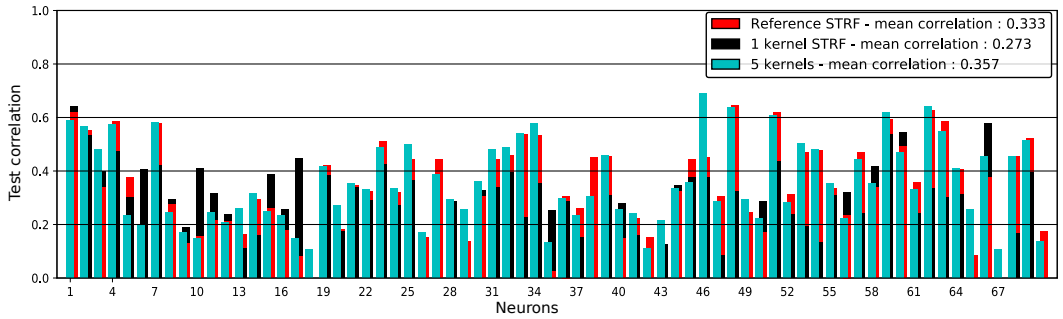

FIGURE 1.6: The STRF model was outperformed by the 5 kernel model (cyan) - Average test correlation for each neuron is shown - Red and black bars are overlapped as opposed to stacked.

On figure 1.6, the score from the fine tuned (reference) STRF model is shown in red. The score from our STRF model is shown in black, different from the one plotted in figure 1.2 as it was trained on more neurons. One must be careful while reading the figure values : the red and black bars are overlapped as opposed to stacked. Finally, the shared 5 STRF model is plotted in cyan. Its mean correlation outperforms slightly the STRF model. The increase in performance is small but stable. In figure 1.8, we can see that for 100 various network initializations, 96 beat the fine tuned STRF model (i.e. mean test correlation is above 0.333).
We can also see that in the 5 kernels model, no neurons are left unpredictable. The minimum test correlation for this model is 0.1, whereas for the reference STRF and 1 kernel model, their test correlations reach negative values. As we can see on the figure, the correlations for neurons 18 and 67 are below 0. This is likely due to the regularizing effect of the subspace for noisy neurons.

This discovery confirms our hypothesis that neurons share computation in a subspace which is represented by a limited number of components. 5 shared kernels describe better the neurons' responses than 70 STRFs fit independently.

Regularizing the re-weighting

As an attempt to improve the model, we have added a regularization term to improve the re-weighting sparsity. As we can see in figure 1.9, the 3^d recording had a distribution of weights which was sparser. In the performance figure (1.6), we can see that the 3^d recording, i.e. from neuron 42 to 60, has the best average test correlation score over the other 3 recordings. Therefore trying to force sparsity is sensible.
The following expression was added to the loss function :

$$\lambda \sum_{s=1}^{S} \sum_{n=1}^{N} |Reweight(s,n)|$$

with $||$: the absolute value,
and λ : the sparsity factor, which is constant, i.e. unlearned during training.

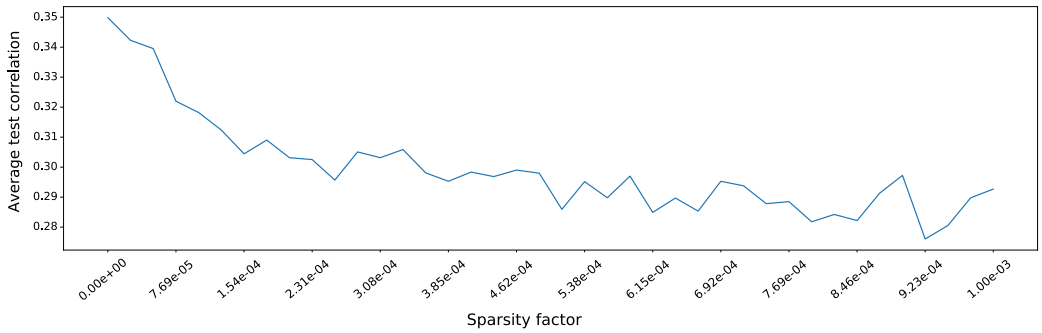

FIGURE 1.7: The influence of re-weighting sparsity on predictive performance

As we can see on figure 1.7, performance is maximum when no sparsity is set. It then decreases regularly when we increase the amount of sparsity. The curve continues to reveal decrease in performance even with the sparsity factor being 4 orders of magnitude higher. This suggests that sparsity is not useful here, possibly because of the small number of kernels. This could also be due to the fact that λ was unlearned, we decided to try a set of constant values for λ so that we could easily try the method with no implementation complexity.

1.3 Properties of the subspace discovered

Patterns are similar across various initializations

An interesting property of the learned subspace is that the time frequency patterns from the 5 kernels are similar across various random initializations.
As an example, were are showing 2 kernels from one of the random initialization (Figure 1.1). These types of kernels appear very frequently. They reveal a high frequency tuning of the neurons based on both a long and short time scale. The advantage of the shared subspace paradigm is that the re-weighting function can then fine tune for each neuron the combination of the kernels that predicts their response the best. The excitatory zones also tend to have a symmetrical inhibitory zone at lower frequencies. Because these 5 STRFs explain responses from many neurons, these properties might reflect an important and general feature of neural coding in A1.

Correlation of kernels is opposed to performance

An experiment we have tried is to assess the stability of our neural network. We initialized it randomly 100 times and compared its overall predictive accuracy for each initialization. Figure 1.8 shows the mean test correlation for each initialization. We can see that the overall test correlation is bound between a relatively small range (0.32 and 0.36), suggesting that the training procedure is not getting caught in local optima (or if it is, the optima have similar test accuracies).
On the x-axis we have also plotted the average correlation between 5 kernel re-weights. If they were all identical, the value would be 1. We can see that their value is relatively small (<0.5), and that when the average correlation increases, the predictive accuracy decreases. This pattern suggests that the subspace is most effective when the distribution of weights across neurons for each kernel is uncorrelated. This observation provides a potential way to improve upon our model in future work by encouraging the subspace to learn kernels with uncorrelated weights (e.g. by including a correlation term in the cost function).

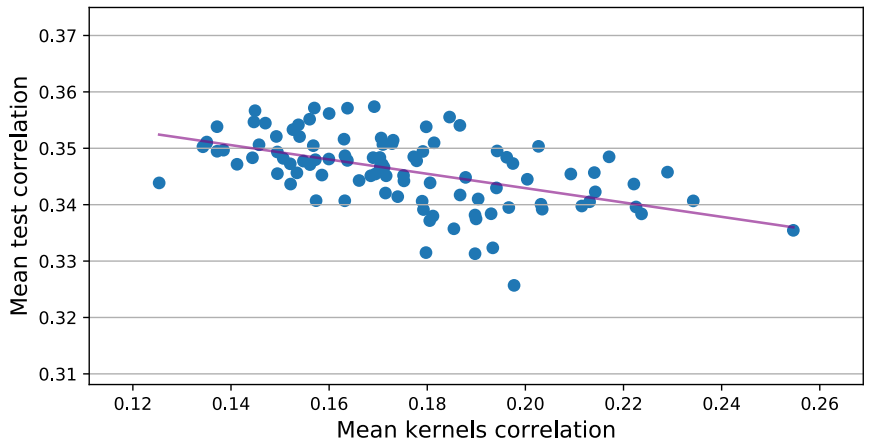

FIGURE 1.8: 100 initializations of the same network - Studying the relationship between mean test correlation and mean kernels correlation

Expressiveness of the 5 kernel model

As explained in the Data Recording section, we can see the in figure 1.9, the 4 various recordings. This plot represents the values of the weights from the re-weighting layer (i.e. that maps kernels/STRFs to the neurons). For example, cell 46 is highly activated by kernel 5 whereas cell 70 is activated by none of the kernels.
The main evidence is that the third recording, from a different ferret, exploits in a greater range the expressiveness of the 5 kernels. Whereas recordings 2 and 4 are mostly tuned to kernel 1. This involves that the various neurons from the ferrets recording zones live in a relatively different subspace. Nevertheless, the common subspace that was learned is expressive and constrained enough to gain prediction accuracy in comparison to the initial STRF model.

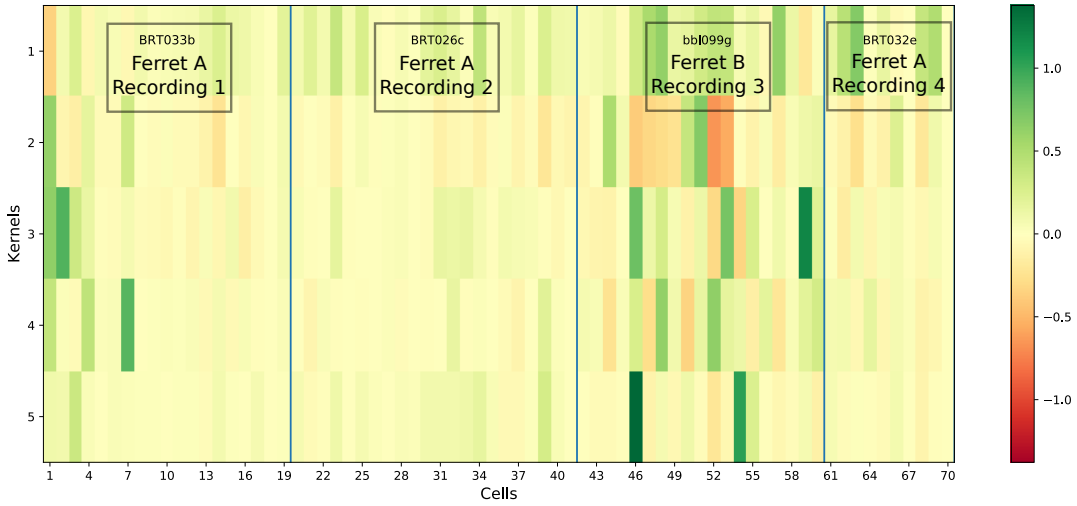

FIGURE 1.9: The learned re-weighting map - The 4 different recordings are split

1.4 Discussion

Complexity reduction is essential to understanding neural encoding, allowing a small number of parameters to represent the activities of many neurons. Our results provide a proof-of-concept that a standard encoding model can be improved and simplified using low-dimensional subspaces. Nevertheless, the performances gains were modest. We would expect a larger improvement in performance as the number of neurons recorded increases. One way to test this hypothesis would be to study the variation in performance when increasing the amount of neurons used by the network.

However, working with a shared subspace consumes less hyper parameter search time. As optimization is run only once versus as many times as the number of neurons for the STRF model. The space of hyper-parameters can therefore be searched much more precisely. The gain in performance increasing again when the number of neurons recorded rises.

When looking at the kernels time frequency patterns, we can see that they are similar to dilated and shifted versions of a common high frequency tuned kernel. This type of structure can be captured by wavelet models, where each kernel is dilated version of a basis kernel. Modeling STRFs using wavelets may thus provide another avenue with which to improve and simplify the STRF model.

Finally, another interesting question is whether the low dimensional subspaces are columnar, i.e. whether neurons that are part of the same "cortical column" live in the same subspace? Knowing that the probes are inserted vertically and in a variety of brain zones, one could train a network on neurons from one recording, and test if it better predicts responses from the same vs. a different recording. If true, this would provide evidence that neurons in the same column share a similar subspace.

Chapter 2

Towards deep nonlinear subspace models

2.1 The limits of the linear model

All the analyses described in the previous chapter implicitly assumed that the encoding of sound in the auditory cortex is primarily linear (except for a point wise non-linearity in the activations). However, given the complexity of the auditory pathway from the cochlea to the auditory cortex, it is natural to hypothesize that cortical responses are highly non-linear [Sahani and Linden, 2003].
After sound is transduced into electrical signals in the cochlea, responses are passed through a series of bilateral subcortical nuclei: the cochlear nucleus, the superior olive, the inferior colliculus and the medial geniculate before reaching primary auditory cortex. Information is initially segregated by ear, but then combined in the superior olivary complex, where binaural processing for sound localization is thought to first occur. In comparison, the visual subcortical pathway is much simpler as there is only a single mandatory relay between the retina and the cortex (the lateral geniculate nucleus). Thus, here we explore whether there is a shared subspace of neural activity that is nonlinear with respect to a cochleogram representation.

How can we explore this idea? Here, we begin by learning a low-dimensional linear subspace directly from the neural data itself (using GPFA and DSS, described below), and then we test the extent to which the basis functions of that subspace can be predicted from the input using a linear STRF.
There are also behavioral and computational reasons to expect cortical responses

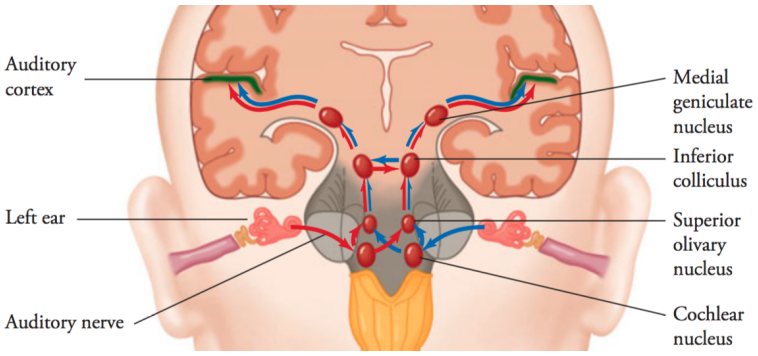

FIGURE 2.1: The auditory path - [McDermott, 2013]

to be highly nonlinear. Many of the tasks that humans excel at (e.g. recognizing phonemes in speech) have only been replicated by highly nonlinear systems with many parameters tuned to accomplish the desired task (e.g. deep networks). Such

models often transform the input representation using a series of nonlinear transformations. The final layers of the network (which are nonlinear with respect to the inputs) can often be linearly retrained for a variety of different tasks, suggesting that they have learned a nonlinear subspace that makes abstract information explicit. Particularly, relevant are recent studies showing that later layer of deep neural networks can be linearly mapped to cortical responses, in some cases dramatically outperforming simpler models [Yamins et al., 2014].

One way to test the hypothesis that neurons live in a nonlinear subspace would be to investigate prediction of the activity of one left-out neuron using a linear mapping of the activity from the remaining neurons. If better predictive accuracy was demonstrated relative to a linear encoding computed from a spectrogram-like representation, this would imply a shared nonlinear subspace. As a first step along these lines, we decided to attempt to learn a low-dimensional subspace directly from the population data we recorded (described next). This subspace could then be used to test the predictive accuracy of the model in left-out neurons in future work.

2.2 The Gaussian Process Factor Analysis transformation

The most standard way to learn a low-dimensional subspace is with PCA. However, PCA has several limitations which are suboptimal from the standpoint of extracting a sensory subspace from neural spiking data. Instead, we begin by applying Gaussian Process Factor Analysis (GPFA) [Byron et al., 2009]. GPFA is a transformation which has two goals : finding an optimal amount of temporal smoothing, and finding a low dimensional subspace. The subspace and smoothing kernels are inferred simultaneously. GPFA typically performs better than first smoothing the data and then applying classical dimensional reduction techniques such as PCA. Gaussian process is used as a probabilistic technique for fitting a multi-variate Gaussian to a set of data. The covariance matrix used for defining the Gaussian process (squared exponential), can be used to model and infer the similarity of spiking data across neurons and time. Factor analysis like PCA finds a low dimensional subspace which captures the variance that the neurons share. But unlike PCA, the model can account for variable amounts of noise across neurons.

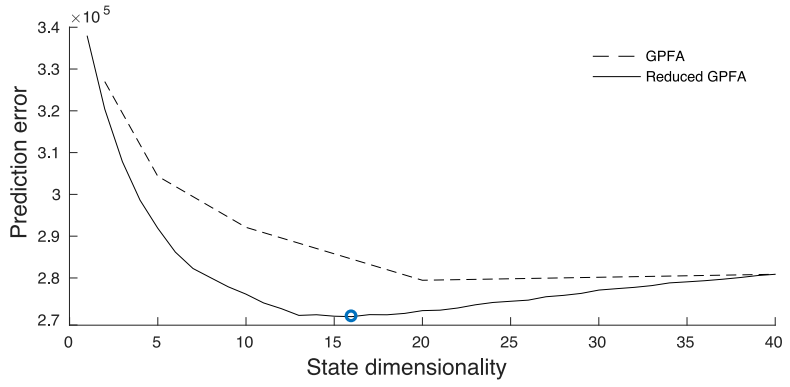

FIGURE 2.2: GPFA search of the optimal number of components based on prediction error

First, we have applied the GPFA transformation to our data, and found the optimal number of components which yield the maximum cross-validated prediction accuracy (as measured by the squared error for left out trials) (see figure 2.2). The results of this analysis suggest that 16 components was optimal, but we decided to use 20 components to be conservative, since these components were going to be further compressed by a subsequent transformation described next.

As the aim of our analysis was to find a low-dimensional subspace which is stimulus driven. To accomplish this goal, we have chosen to apply a second transformation to enhance this property. We accomplished this objective using a second technique : Denoising Source Separation (DSS) (the name is somewhat misleading in this context) [Cheveigné and Parra, 2014]. We use DSS to find components that are maximally reliable across repetitions of the same stimulus. The approach works by (1) whitening the input subspace (here from GPFA) (2) averaging across repetitions and (3) applying PCA again to sort components by their reliability. Below we plot the reliability of the components found. We note that in principle we could have applied DSS to the raw data without GPFA, but we expect this to lead to poorer results because there is no way to infer an optimal smoothing kernel using DSS.

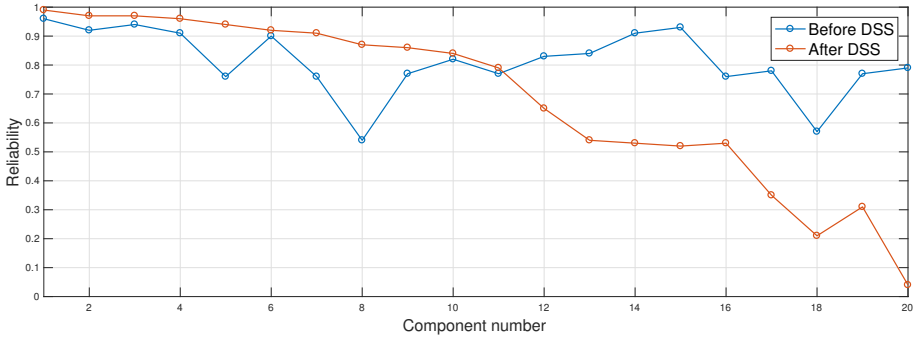

FIGURE 2.3: Comparison of the reliability of the components before and after DSS transformation

Figure 2.3 shows the reliability for each component after GPFA transformation. Reliability was computed per component, by averaging across two independent splits of data (i.e. different repetitions of the same stimuli), and correlating responses across the two splits. The responses for each split are shown in appendix B as stimulus by time images (one image per split and component) before and after DSS. We can see in figure 2.3 that the top DSS components are substantially more reliable than the top GPFA components. In particular, the top 11 components are highly reliable (they produce a very similar response across multiple repetitions of the same stimulus). We have thus focused on trying to understand these components.

We used the STRF model to predict the 20 DSS components directly, based on the cochleograms previously computed. Since these components are highly reliable and account for a substantial fraction of the neural response variance, the ability of the STRF model to account for these component responses provides a measure of how nonlinear the neural subspace is. After finding an optimal set of hyper-parameters, average prediction accuracies on the test data were computed for each component (figure 2.4)

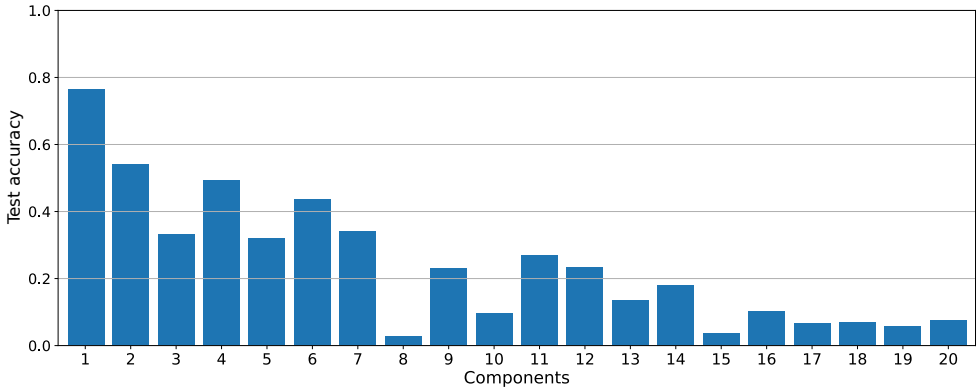

FIGURE 2.4: GPFA-DSS components predicted by the STRF model - Studying the test accuracy for each component

As in standard PCA methods, components explain less variance as they increase. With an average test accuracy of 0.24 on the 20 components, we can say that the components from the low dimensional GPFA-DSS subspace can't relatively well be predicted by the linear model. As the transformation enhances explained variance of the data (at least for the first components), poor performance in linearly predicting them provides further evidence on how non linear the subspace is.

For better understanding, we represented the subspaces from which we have performed transformations, trained our model and from which we we will compare prediction accuracies (figure 2.5).

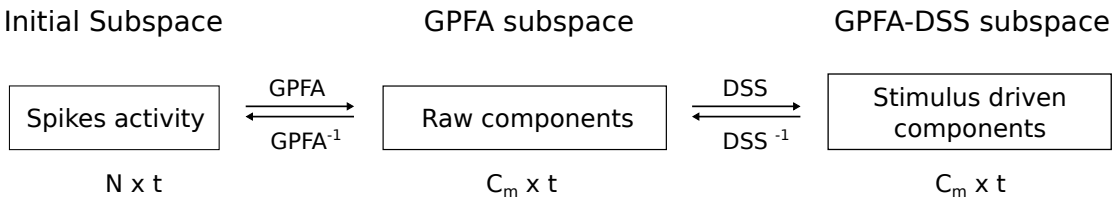

FIGURE 2.5: Visual description of the GPFA-DSS transformations

With $N = 70$ being the number of neurons and $C_m = 11$ the number of low dimensional components.
This primary result showed how linear the subspace is, the question being to know if we could predict the raw temporal spiking data better by first predicting these components and then applying the inverse transform before training our data. We have tested this method and the global results are shown in figure 2.6.

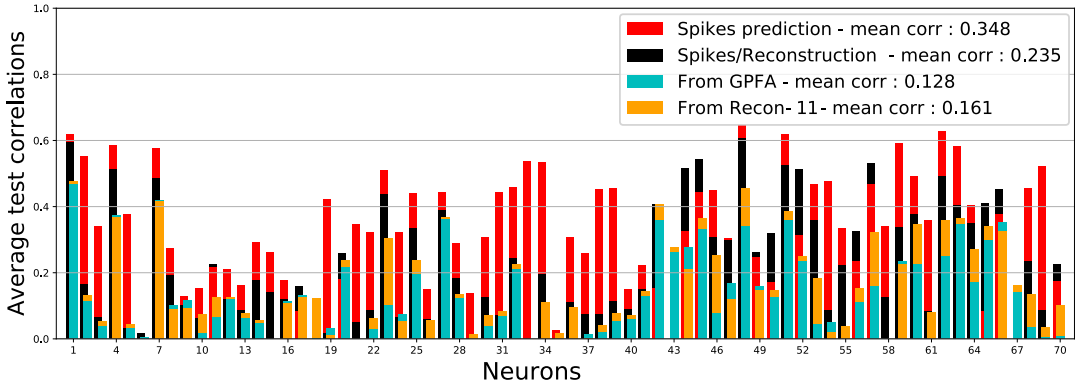

FIGURE 2.6: Comparison of 3 various predictions of the 70 neurons. 2 various GPFA-DSS predictions vs. direct spikes prediction vs spikes. reconstruction accuracy - Detailed explanation is found below

As in figure 1.5, red and black bars are overlapped as opposed to stacked. The same for the cyan and orange bars. The 4 types of bars represent the average test correlation across the 18 test stimuli of various predictions with the spiking data, in the initial subspace :

- The red bars represent accuracy when directly predicting the spiking activity, from the initial subspace. It is the same one to one STRF method as in part 1.1.
- The orange bars represent the accuracy of predicting the spikes which were first transformed in the GPFA-DSS subspace, then inverse transformed back to the initial subspace and then predicted. Using 11 components.
- The black bars show the correlation between the initial spikes and the spikes which were reconstructed after being GPFA-DSS transformed and inverse transformed to the initial subspace.
- Finally, the cyan bars show the accuracy of predicting the components from the GPFA-DSS subspace and then inverse transforming these predictions into the initial subspace.

As shown in figure 2.6, the direct prediction of the spikes in the initial subspace has the best performance (red bars). The performances of our attempts haven't reached our expectations. Nevertheless, we have similarity in performance with the methods shown in cyan and in orange. As they are the same computations achieved in a different order, their difference in average correlation is only 0.033. We decided to show the black bars : the correlation between the spikes and their reconstruction, to show that poor performance with the two GPFA-DSS methods (cyan and orange) were often correlated with a bad reconstruction of the spikes. For example, neuron 65 is much better reconstructed than it is fitted by the STRF (0.41 correlation for the GPFA-DSS reconstructed spikes against 0.08 correlation for the STRF fitted spikes), so the two GPFA-DSS methods beat the standard STRF method with a factor of 3.6 and 4.1, for the cyan and the orange methods respectfully. Whereas when the black bars are low compared to the red, the GPFA-DSS methods perform badly.
Ultimately, no improvements were achieved in predicting the spiking data with this method.

2.3 Multilayer network performance analysis

The analyses from section 2.2 suggest that neurons in A1 live in a common subspace that is nonlinear with respect to a spectrogram representation. How is this subspace computed? Here we made a first attempt to directly learn this subspace using recent advances in neural network training. Specifically we attempted to adapt a previously published method [Klindt et al., 2017] for training multi-layered CNNs to predict spiking activity from visual cortex. Our main hypothesis is similar to the one we used in section 1.2 : nearby neurons share computation, so a shared subspace should predict responses better, but here we made a first attempt to learn a shared subspace that is nonlinear. The main issue in adapting the Klindt et al., 2017 model was that the algorithm was designed to predict the spike rate of visual cortical neurons to static natural images (the model was tested on data from mouse visual cortex). To adapt the model, we considered our auditory neural time series as a sequence of average responses to images that were shifted in time, where the images were slices of cochleograms. This provides a natural way to build in the notion of temporal convolution into a model that does not have a notion of time. However, this approach led to an increase in memory usage which was a challenge for GPU training (e.g. for a 3 second recording at 100Hz, 300 cochleograms were fed to the network compared to feeding 1 image).

The model is the following (figure 2.7) : We used a model with 3 layers. The in-

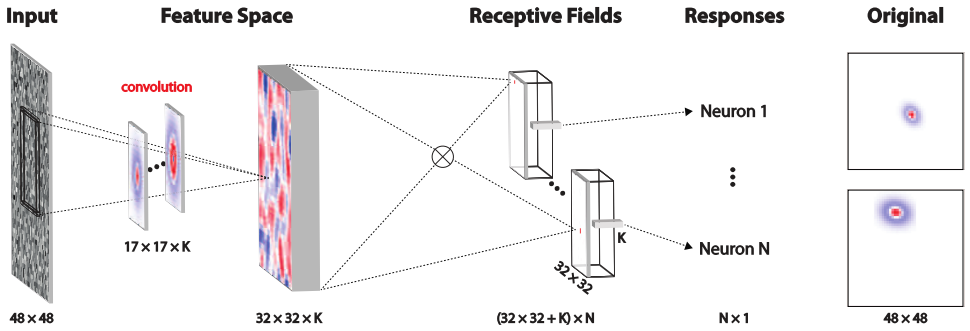

FIGURE 2.7: Vision CNN model that we adapted to audition data [Klindt et al., 2017]

put "image" was 48 x 48. Then K kernels of size 17 x 17 were "spatially" convolved with the image (i.e. convolved in time and frequency), resulting in a 32 x 32 x K feature space. The final read out layer is actually double as there is both a reweighting of the various kernels and of the various kernel zones. With our auditory data, this implies that the network trains for each neuron, to find a set of kernels, and a time frequency zone that it prefers.
In addition to this shared feature space model, 3 regularization terms were implemented by Klindt et al., 2017. First an $L1$ sparsity penalty on the kernel re-weighting, just as we used in part 1.2, to encourage the model neurons in the penultimate layer to resemble neural activity.. Then a second $L1$ sparsity penalty, but this time for the "per kernel based" re-weighting. And finally a $L2$ smoothness penalty applied to the different kernels.
Our first issue was due to the sparsity of the data. Whatever combination of hyperparameters we used, the network was predicting the average of the data, with no variance. Over fitting wasn't the issue as it was happening during the first iterations

with very low learning rates. By slightly low pass filtering the training data, the issue was fixed. The network should have been able to learn this filtering but no visible cause was found.

Getting the network to predict reliable responses was relatively difficult during our first attempts. So we decided to start by maximizing training accuracy, by attempting to over-fit the data. Our best set of hyper-parameters led us to the scores in figure 2.8, where we compare with the training correlation of the STRF model.

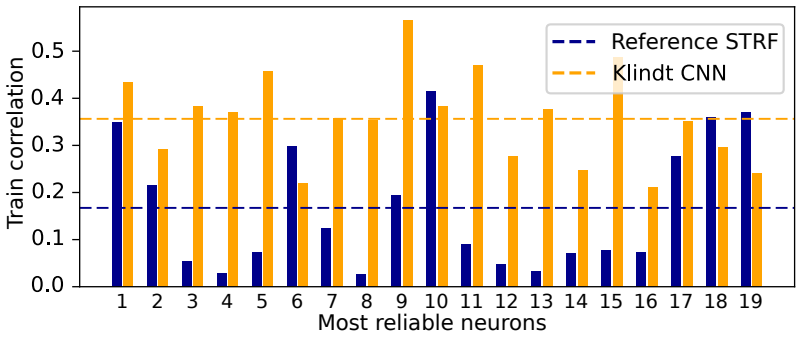

FIGURE 2.8: Over fitting the data (sanity check) with the Klindt CNN model

During our first attempts, we also selected the 19 out of 70 most reliable neurons to reduce fitting difficulty. Reliability was measured in the same way as in part 2.2. This result was obtained with no regularization, no early stopping, and the following parameters :

Learning rate	Weight initialization	Batch size	Number of iterations	Filter sizes
0.005	0.27 - 0.27 - 0.27	600	30000	30 - 30 - 30

After a random search over 200 combinations of parameters and a grid search on the 3 regularization parameters. Our best predictive model for test data was found and figure 2.9 shows how it compares to the STRF model.

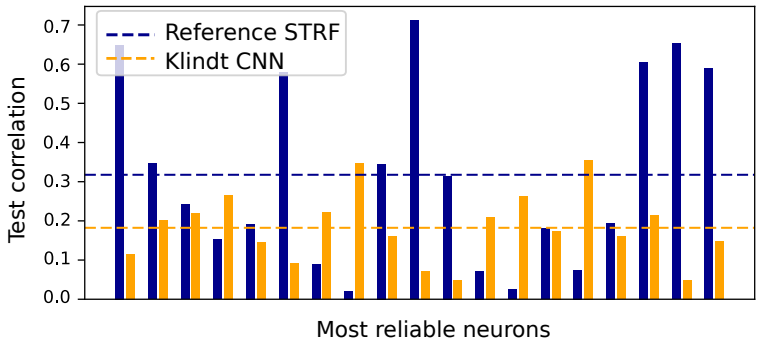

FIGURE 2.9: Maximum test correlation obtained after largely searching the hyper parameter space with the CNN model

Our model's maximum performance was in average 40% below the STRF model. However, some cells gain from a better performance of the model.

2.4 Discussion

As a global consideration, the same lack of data could have been limiting the power of our models, probably more than in the 1^{st} section. Especially for the deep convolutional network where it has been proven to outperform vision neurons predictions, with a much larger amount of data than what we have been using.
Nevertheless, a way of improvement would have been to investigate further the set of neurons which perform the best. By finding properties which improve prediction, such as reliability, these properties could have been used to tailor a specific transformation that leads to a subspace which is better linearly predictable.
The assumptions made by our GPFA-DSS transformation may have been wrong in relation to our data. We could have compared prediction performance of our GPFA components with a standard PCA dimensionality reduction. As GPFA emphasizes more on both covariance and the modeling of spiking noise.

Applying DSS was however sensible as many issues while training the CNN were due to the sparsity of the data. We had to manually low-pass filter the data as the network was not able to achieve the smoothing. Some attention was drawn to sparsity, while creating increasingly complex synthetic data during our first training attempts. A deeper focus on this aspect could have been undertaken.

Finally, the neural data has been proven to be fairly non linear with respect to the input. But no improvement was gained in predicting the spiking data, while using the methods experimented.

Conclusion

We have built, trained and tested models predicting neural activity in response to sound. The models' transfer functions range from purely linear to moderately non-linear. We confirmed that the responses are highly non linear with respect to the stimulus. We could have used considerably deep neural networks to model the computation performed in the brain but we decided to use simpler models. The reason is partly due to the little amount of data we disposed of, but was mainly in the interest of being able to interpret the results obtained.

However, all our models were relatively general and considered the neural data as any type of data. An exception can be made for the GPFA transformation which accounted for spiking noise. This noise however could be similar to another type of data noise. Therefore, we carried out a short experiment, not referred to in the report, to try and model a known neurological property : gain control. It is performed by neurons to adapt to extreme differences in intensity of stimuli. We created a model that computes this property but no improvements in predictive accuracy were found. It showed that even simple neural functions are challenging to model computationally.

However, we have found a low-dimensional linear subspace which performs better than the linear STRF model. Proving that nearby neurons share computation. Data from two different ferrets were used and the combination of both spiking activity also improved the predictive accuracy. This suggests that neurons from the primary auditory cortex share computation even across various brains.

Even though ferret cognition, and more generally, the neural network from any auditory brain is not perfectly understood, we are able to model a part of the computation and generate accurate spiking predictions (see appendice A). The shared subspace concept seems promising and further effort should be made to capture non linearities from the neural data.

Appendix A

Temporal predictions

FIGURE A.1: An example of some accurate predictions with the 5 kernel subspace model

Appendix B

Reliability before and after DSS

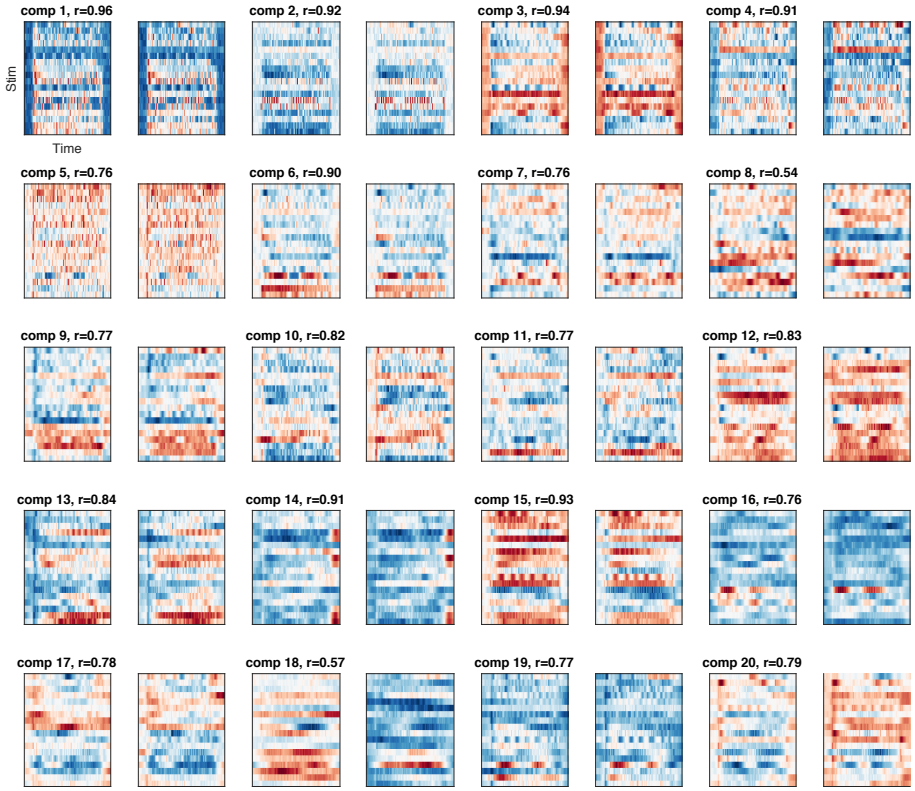

FIGURE B.1: Reliability before DSS

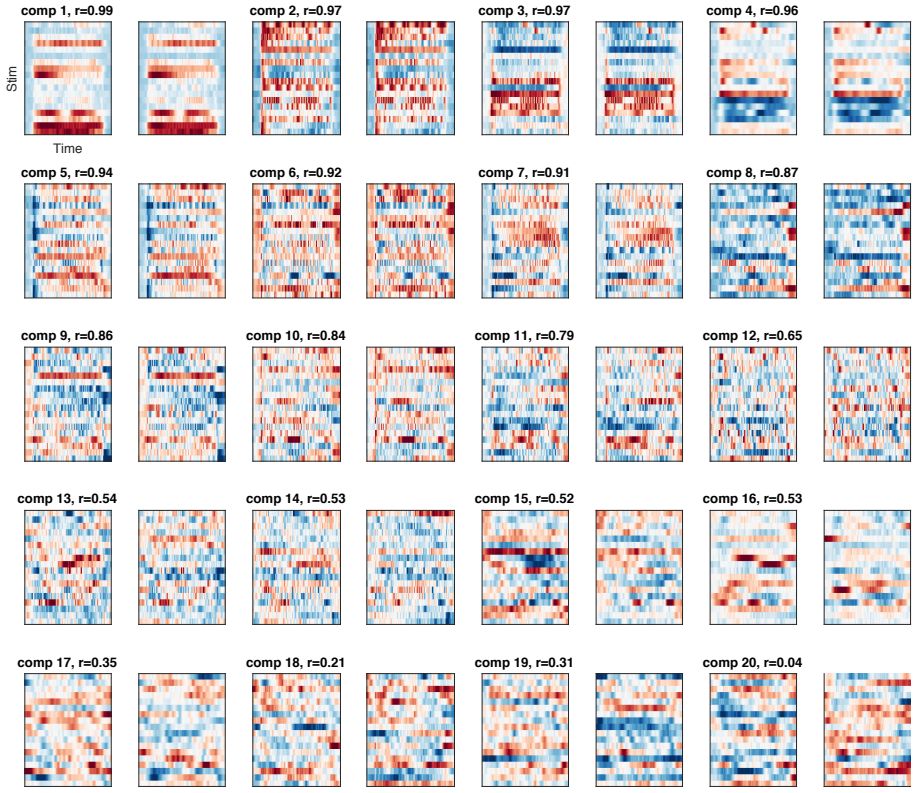

FIGURE B.2: Reliability after DSS

Bibliography

Alves-Pinto, Ana et al. (2016). "Behavioural estimates of auditory filter widths in ferrets using notched-noise maskers". In: *The Journal of the Acoustical Society of America* 139.2, EL19–EL24.

Atencio, Craig A, Tatyana O Sharpee, and Christoph E Schreiner (2008). "Cooperative nonlinearities in auditory cortical neurons". In: *Neuron* 58.6, pp. 956–966.

Byron, M Yu et al. (2009). "Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity". In: *Advances in neural information processing systems*, pp. 1881–1888.

Calabrese, Ana et al. (2011). "A generalized linear model for estimating spectrotemporal receptive fields from responses to natural sounds". In: *PloS one* 6.1, e16104.

Cheveigné, Alain de and Lucas C Parra (2014). "Joint decorrelation, a versatile tool for multichannel data analysis". In: *Neuroimage* 98, pp. 487–505.

Cunningham, John P and M Yu Byron (2014). "Dimensionality reduction for large-scale neural recordings". In: *Nature neuroscience* 17.11, p. 1500.

Freeman, Jeremy and Eero P Simoncelli (2011). "Metamers of the ventral stream". In: *Nature neuroscience* 14.9, p. 1195.

Hong, Ha et al. (2016). "Explicit information for category-orthogonal object properties increases along the ventral stream". In: *Nature neuroscience* 19.4, p. 613.

Hung, Chou P et al. (2005). "Fast readout of object identity from macaque inferior temporal cortex". In: *Science* 310.5749, pp. 863–866.

Klindt, David et al. (2017). "Neural system identification for large populations separating "what" and "where"". In: *Advances in Neural Information Processing Systems*, pp. 3506–3516.

Kozlov, Andrei S and Timothy Q Gentner (2016). "Central auditory neurons have composite receptive fields". In: *Proceedings of the National Academy of Sciences* 113.5, pp. 1441–1446.

McDermott, Josh H (2013). "Chapter 8 - The Oxford Handbook of Cognitive Neuroscience". In: 1, pp. 135–170.

Nguyen, Anh et al. (2016). "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks". In: *Advances in Neural Information Processing Systems*, pp. 3387–3395.

Norman-Haignere and McDermott (2018). "Neural responses to natural and model-matched stimuli reveal distinct computations in primary and non-primary auditory cortex". In: *Under review*.

Romani, Gian Luca, Samuel J Williamson, and Lloyd Kaufman (1982). "Tonotopic organization of the human auditory cortex". In: *Science* 216.4552, pp. 1339–1340.

Sahani, Maneesh and Jennifer F Linden (2003). "How linear are auditory cortical responses?" In: *Advances in neural information processing systems*, pp. 125–132.

Simoncelli, Eero P and Bruno A Olshausen (2001). "Natural image statistics and neural representation". In: *Annual review of neuroscience* 24.1, pp. 1193–1216.

Slaney, Malcolm (1998). "Auditory toolbox". In: *Interval Research Corporation, Tech. Rep* 10, p. 1998.

Stosiek, Christoph et al. (2003). "In vivo two-photon calcium imaging of neuronal networks". In: *Proceedings of the National Academy of Sciences* 100.12, pp. 7319–7324.

Thorson, Ivar L, Jean Liénard, and Stephen V David (2015). "The essential complexity of auditory receptive fields". In: *PLoS computational biology* 11.12, e1004628.

Yamins, Daniel LK et al. (2014). "Performance-optimized hierarchical models predict neural responses in higher visual cortex". In: *Proceedings of the National Academy of Sciences* 111.23, pp. 8619–8624.