

DISS. ETH NO. 25010

# A Generalist Approach to Learning and to Biological Neuron Networks

A thesis submitted to attain the degree of  
Doctor of Sciences of ETH Zurich

presented by  
Tycho-Cosimo True Charity Riday

Msc ETH  
Computational Science and Engineering

born 24.09.1985

citizen of the USA

accepted on the recommendation of

Prof. Dr. Kevan A. C. Martin  
Prof. Dr. Joachim M. Buhmann  
Dr. Graham Knott

2018

# Contents

|   |           |
|---|-----------|
| <b>Summary</b>  | <b>5</b>  |
| <b>Zusammenfassung</b>                                  | <b>7</b>  |
| <b>Declaration</b>                                      | <b>9</b>  |
| <b>Acknowledgements</b>                                 | <b>10</b> |
| <b>1 Introduction</b>                                   | <b>11</b> |
| 1.1 Motivation . . . . .                                | 11        |
| 1.2 A history in neuroscience . . . . .                 | 12        |
| 1.3 Cortical Microcircuits . . . . .                    | 14        |
| 1.4 Mapping the Matrix . . . . .                        | 16        |
| 1.5 Connectomics . . . . .                              | 17        |
| 1.6 Neural Networks - Cycling Data and Models . . . . . | 19        |
| <b>2 Synapse Densities</b>                              | <b>22</b> |
| 2.1 Tissue Preparation . . . . .                        | 23        |
| 2.2 Electron Microscopy . . . . .                       | 23        |
| 2.3 Results . . . . .                                   | 25        |
| <b>3 Random Forests and Classification</b>              | <b>32</b> |
| 3.1 Introduction of Random Forests . . . . .            | 32        |
| 3.2 Definitions and Algorithms . . . . .                | 33        |
| 3.3 Decision Tree Split Criteria . . . . .              | 34        |
| 3.4 Alternative coordinate systems . . . . .            | 36        |
| 3.5 Ensemble weighting . . . . .                        | 37        |
| 3.6 Out of bag Error . . . . .                          | 38        |
| 3.7 Classification Experiments . . . . .                | 38        |
| 3.8 Random forests for regression . . . . .             | 43        |
| 3.9 Random forest proximities and clustering . . . . .  | 44        |
| 3.10 Regression Experiments . . . . .                   | 46        |
| 3.11 Remarks . . . . .                                  | 48        |

|          |   |           |
|----------|---|-----------|
| <b>4</b> | <b>Image Processing</b>                             | <b>52</b> |
| 4.1      | Spatial frequency filters . . . . .                 | 52        |
| 4.2      | Watershed to super pixels . . . . .                 | 55        |
| 4.3      | Region Geometry . . . . .                           | 55        |
| 4.4      | Scaled Region PCA . . . . .                         | 56        |
| 4.5      | Spiral neighborhoods . . . . .                      | 56        |
| <b>5</b> | <b>Reclassification</b>                             | <b>67</b> |
| 5.1      | Boosting . . . . .                                  | 67        |
| 5.2      | Classification feedback . . . . .                   | 68        |
| 5.3      | Automaton Forest . . . . .                          | 68        |
| <b>6</b> | <b>EM Segmentation</b>                              | <b>70</b> |
| 6.1      | Classification Pipeline . . . . .                   | 70        |
| 6.1.1    | Image Decomposition . . . . .                       | 70        |
| 6.1.2    | Region Analysis . . . . .                           | 71        |
| 6.1.3    | Iterative Classification . . . . .                  | 71        |
| 6.2      | Implementation . . . . .                            | 72        |
| 6.3      | Results . . . . .                                   | 74        |
| 6.4      | Discussion . . . . .                                | 76        |
| <b>7</b> | <b>Random Forest Source Separation</b>              | <b>77</b> |
| 7.1      | Introduction . . . . .                              | 77        |
| 7.2      | Dynamic Range Compression . . . . .                 | 78        |
| 7.3      | Spectrograms and Binary Masks . . . . .             | 78        |
| 7.4      | Classification of Time Frequency Bins . . . . .     | 78        |
| 7.5      | Sampling of Training T-F Bin Locations . . . . .    | 79        |
| 7.6      | Feature Extraction . . . . .                        | 79        |
| 7.7      | Reclassification . . . . .                          | 80        |
| 7.8      | Experimental Setup . . . . .                        | 80        |
| 7.9      | Results . . . . .                                   | 82        |
| 7.9.1    | Parameters . . . . .                                | 82        |
| 7.9.2    | Evaluation Metrics . . . . .                        | 82        |
| 7.9.3    | Evaluation . . . . .                                | 83        |
| 7.9.4    | Frequency-wise Analysis . . . . .                   | 85        |
| 7.9.5    | Discussion . . . . .                                | 86        |
| <b>8</b> | <b>Rate-based Neural Networks for Amplification</b> | <b>91</b> |
| 8.1      | Network Constraints . . . . .                       | 91        |
| 8.2      | Neuron Model . . . . .                              | 92        |
| 8.3      | Learning Model . . . . .                            | 92        |
| 8.4      | Evaluation Measures . . . . .                       | 94        |
| 8.4.1    | Equilibrium Firing Statistics . . . . .             | 94        |

|          |  |            |
|----------|--|------------|
| 8.4.2    | Equilibrium Self-Similarity . . . . .                        | 95         |
| 8.4.3    | Reassignment Similarity . . . . .                            | 96         |
| 8.5      | Network Examples . . . . .                                   | 96         |
| 8.5.1    | Linear Excitatory Network . . . . .                          | 97         |
| 8.5.2    | Rectified Linear Network with Inhibition . . . . .           | 98         |
| 8.5.3    | Root Network . . . . .                                       | 101        |
| 8.5.4    | Learning Root Network . . . . .                              | 104        |
| 8.6      | Discussion . . . . .   | 106        |
| <b>9</b> | <b>Discussion</b>  | <b>122</b> |
| <b>A</b> | <b>Appendix A - Density Reviews</b>                          | <b>125</b> |
| A.1      | Review of Synapse and Neuron Density Papers . . . . .        | 125        |
| <b>B</b> | <b>Appendix B - Random Forest Result Tables</b>              | <b>131</b> |
| B.1      | Random Forest Classification Results . . . . .               | 131        |
| B.2      | Random Forest Regression Results . . . . .                   | 144        |
| <b>C</b> | <b>Appendix C - Image Segmentation Examples</b>              | <b>150</b> |
| C.1      | EM Segmentation Results . . . . .                            | 150        |
| <b>D</b> | <b>Appendix D - Random Forest Source Separation</b>          | <b>169</b> |
| D.1      | Random Forest Source Separation Frequency Analysis . . . . . | 169        |
|          | <b>Bibliography</b>  | <b>200</b> |
|          | <b>Curriculum Vitae</b>                                      | <b>216</b> |

# Summary

To understand fundamental similarities within the mammalian neocortex, or in other words to find a cortical canonical microcircuit, we require the connectivity mapping of the layers of the neocortex. Such a mapping can be acquired by extrapolating bouton counts of single neuron reconstructions throughout all cortical layers and distributing these among extrapolated dendrite segments of the same reconstructions. To validate the completeness of such an extrapolation, we can compare the resulting bouton density estimates against actual synapse density estimates in the same tissue.

One goal of this thesis was to provide such synapse density estimates for the mouse neocortex, in particular for the primary auditory cortex as well as the primary motor cortex. Synapse density estimates were obtained by manually counting individual synapses on ultra-thin brain sections under the electron microscope using the disector method. This manner of counting requires arduous work, imaging and investigating many hundreds of sampling locations. To this purpose, additionally to manual synapse counts, a bulk effort was put into developing automated methods of synapse recognition. When a pixel-wise supervised classification approach is taken, training data can be extracted from the sectioned images. A pixel-wise classification can preserve structural information contained in the pixel connectivity, such as object shape and size. The random forest classifier was used for the classification task due to its algorithmic simplicity and adaptability, the opportunities it offers to investigate input variable interactions, its robustness to noisy inputs, its ability to ignore spurious inputs as well as its invariance towards monotonic transformations of the input dimensions.

A set of scalable spatial frequency bandpass filters was developed which captures general spectral image statistics at multiple scales and provides a first set of input features to the random forest classifier. These filters at various scales are a favorable basis for a novel two-sided watershed segmentation based approach which partitions the images into scale specific super pixels. These super pixels allow for the computation of localized pixel statistics such as mean and variance within a region. Moreover, given a preliminary classification, we have the possibility to investigate the distribution of the classes within such super pixels at various scales. This proves useful in the reclassification of previously misclassified single pixels.

Further, preliminary classifications provide pixel connectivity information of the classes. The geometry of the resulting connected regions can also be used in the reclassification of misclassified pixels. The process of classifying individual pixels, estimating the class statistics of the super pixels as well as the geometrical information of connected class regions and setting up a new classifier which utilizes this information can be repeated multiple times,

if provided a sufficient amount of training data. Multiple such classification tasks can be addressed at once and may exchange information to improve classifications.

An advantage of this approach is that while the intention is for electron microscopy data and the recognition of synapses, the exact same approach can be applied to any pixel classification task in 2D images (expansions to 3D data are conceptually straightforward, but still computationally prohibitive).

Further this type of classification approach has been modified to solve an audio source separation task and implies the generality of the classification setup.

This thesis is rounded off with the presentation of a simple neuron network model which links findings from the synapse density estimates to minimal requirements for network stability under strong input amplification using a strongly recurrently connected network.

This thesis emphasizes the interplay and cross relating of data analysis, development of automatization tools as well as the development of simple descriptive models to deepen the understanding of neuron networks in biology as well as in machine learning.

# Zusammenfassung

Um die grundlegenden Ähnlichkeiten, oder anders ausgedrückt, ein kanonisches Vernetzungsmuster, in der Hirnrinde von Säugetieren zu finden, müssen wir die Vernetzung zwischen den Schichten der Rinde kennen. Wir können eine Tabelle von Vernetzungsstärken erhalten, in dem wir Boutons von einzelnen Nervenzellenrekonstruktionen in den Schichten der Hirnrinde zählen, diese mit der Anzahl solcher Nervenzellen hochrechnen und gleichmässig auf eine ähnliche Hochrechnung für Dendritensegmente der gleichen Nervenzellen in allen Schichten verteilen. Um die Vollständigkeit einer solchen Hochrechnung zu beurteilen, werden solche Schätzer für die Boutondichte mit Messwerten von expliziten Synapsendichte Messungen verglichen.

Ein Ziel dieser Arbeit war es solche Synapsendichten in der Hirnrinde der Maus, insbesondere in der primären Hör- sowie der primären Motorrinde zu messen. Synapsendichten wurden auf ultra-dünnen Schnitten unter dem Elektronenmikroskop durch das zählen einzelner Synapsen mit der Disector Methode gemessen. Dieses Zählen ist mit langwieriger Arbeit verbunden, und es werden hunderte von Messstellen fotografiert und ausgewertet. Aus diesem Grund wurde, zusätzlich zur manuellen Auswertung, erhebliche Arbeit investiert um automatisierte Verfahren für diesen Vorgang zu entwickeln. Die vorhandenen Bilder können als Trainingsdaten für einen Klassifikationsalgorithmus benutzt werden, welcher einzelne Pixel klassifiziert. Bei der Klassifizierung von einzelnen Pixeln kann geometrische Information von verbundenen Pixeln, wie Formen und Grösse, berücksichtigt werden. Hier wurde der Random Forest Algorithmus für die Klassifikation benutzt. Dieser wurde wegen seiner algorithmischen Einfachheit und Anpassungsfähigkeit gewählt, für die Möglichkeit Variableninteraktionen zu betrachten, seine Robustheit in Angesicht von Ausreißern und der Invarianz einzelner Eingabedimensionen bezüglich monotonen Transformationen.

Eine Serie von skalierbaren Ortsfrequenz-Bandfiltern wurde entwickelt, welche allgemeine Frequenzmuster auf mehreren Grössenskalen festhält und einen Teil der Eingabedimensionen für den Random Forest Algorithmus bietet. Diese Filter bilden die Basis für eine neue doppelseitige Wasserscheidetransformation, welche die Bilder auf ausgewählten Skalen in Super-Pixel aufteilt. Die Super Pixel können benutzt werden um die lokale Verteilung von Pixelintensität zu beschreiben. Wenn eine Erstklassifizierung vorgenommen wurde, kann in diesen Super Pixeln die Verteilung der einzelnen Klassen auf verschiedenen Skalen untersucht werden. Diese kann für Zweit- und weitere Klassifizierungen nützlich sein um zuvor falsch klassifizierte Pixel zu korrigieren.

Für jede Klassifizierung kann die Verbundenheit der einzelnen Pixel untersucht werden.

Die Geometrie solcher verknüpften Regionen kann benutzt werden um falsch klassifizierte Pixel zu erkennen. Wenn genug Trainingsdaten vorhanden sind, kann der Prozess von der Analyse der geometrischen Verbundenheit auf verschiedenen Skalen sowie der Formen von verbundenen Klassenregionen und Korrektur einzelner Pixel beliebig wiederholt werden. Weiter können mehrere Klassifikationsaufgaben gleichzeitig gelöst werden und können sich gegenseitig durch Korrelationen mit den anderen Klassifikationen verbessern.

Zusätzlich zur beabsichtigten Erkennung von Synapsen in Elektronenmikroskop Daten können die gleichen Ansätze für beliebige Pixel Klassifikationen von 2D Bildern gebraucht werden (Erweiterungen auf 3D Daten sind konzeptuell gut möglich, erfordern aber erheblich mehr Ressourcen zur Berechnung)

Ebenfalls kann dieser Klassifikationsansatz angepasst werden um Audio Source Separation in Angriff zu nehmen. Dies weist auf die Allgemeinheit des Klassifikationsansatzes hin.

Diese Arbeit endet mit der Vorstellung eines einfachen Modells eines neuronalen Netzwerkes, bei der eine Beziehung zwischen Resultaten von den Synapsendichtemessungen und minimale Voraussetzungen für Netzwerkstabilität für ein System mit sehr starker Eingabeverstärkung durch rekurrente Verbindungen im Netzwerk hergestellt wird.

Diese Arbeit hebt das Zusammenspiel und das fachübergreifende Verknüpfen von Datenanalyse, der Entwicklung von Automatisierungsmethoden, so wie dem Entwickeln von einfachen, beschreibenden Modellen um das Verständnis von Neuronen Netzwerken in der Biologie sowohl als auch im Bereich Machine Learning zu erweitern hervor.



# Declaration

I declare that the work described in this thesis including analysis of the biological data as well as the implementation of the proposed algorithms was performed by myself, the author of this thesis, with consideration of the following:

The mice for the synapse density estimates were worked with and prepared by Nuno da Costa and Marco Perrella. Histological preparations and cutting of the ultrathin section for the electron microscope were performed by Simone Rickauer and Rita Bopp.

The work on audio source separation in chapter 7 was conceived, implemented and evaluated as a joint effort with Saurabh Bhargava.

# Acknowledgements

Here I would like to express some of the thanks which should have been expressed more over the course of this thesis. I was offered wonderful opportunities to grow and learn but also experience one of the most awesome work environments I could imagine.

I am grateful for my initial tutor and guide Nuno da Costa. It is through his example and love for the “wet sciences”, that I was able to develop a genuine interest in biology. He was always willing to engage in scientific discussions and was always open for new approaches and ideas, and was willing to redirect me when I might drift off the wrong direction. Definitely a great example of what it means to do interdisciplinary science.

I am grateful my senior supervisor Kevan Martin who let me have the possibility to learn and test so much throughout this thesis. I’ll remember his patience and will remain impressed by the consistent scientific integrity I always observed in him.

I’d like to thank John Anderson for all his anecdotes on synapses. I’d like to thank Rita Bopp for her technical assistance. We had a lot of good discussions in and outside of science. I’d like to thank German Koestinger for his help with electron microscopy as well as many a good lunch break. I’d like to thank Simone Rickauer for histological background support.

INI is such a great environment for interdisciplinary discussions, and I wish to thank everyone who helped me broaden my horizon in different fields. My fondest memories are of the many discussions I was able to have. You know I enjoyed discussions.

Long live Wednesday morning breakfast meetings!

I’d like to Saurabh Bhargava for our collaboration on the audio source separation work. But much more than that, I would like to thank him for his ongoing example of what it means to be a friend. You set very high standards, Sir!

There are many more I should express gratitude towards: My initial physicist neighbors, turn clockwise 90 and 180 degrees, the old INI Fussball group, someone who emphasized the importance of adding color to your work and also many others.

Finally, I want to thank everyone who bore with me these recent years also outside of the field of science. Even if chances are high you’ll never read this...

# Chapter 1

## Introduction

### 1.1 Motivation

Every day humans solve a plentitude of, at the very least from a current computational perspective, interesting tasks varying from face and expression recognition tasks, to speech processing and empathic interpretation, all which are amazingly contained even in simple conversations between humans, to coordinated motor tasks such as bipedal walking, playing the piano or swimming. This only marks the tip of the iceberg and the list of tasks is essentially endless. What makes the matter even more impressive is that these tasks can be combined and solved in seemingly parallel and that many of the tasks can still be solved when various sensory input is missing or when external conditions are changed. Also, all this is performed using a relatively low energy consumption. Many of these tasks are not only solved by humans, but by other mammals, vertebrates and even invertebrates. A question would be: How?

As humans, understanding how all these tasks are solved can has various advantages. An understanding can be helpful if we wish to repair a broken system, i.e. provide medical methods to heal those who are sick, infirm or else wise handicapped. An understanding can be helpful in providing new tools to extend human task solving abilities. In ancient times this went from using pulleys for construction to written knowledge storage in books and in modern time goes on to interacting with multimedia systems, using telescopes to investigate the depths of space, and to the creation of commercials to influence the behavior of economic markets. An understanding is useful when we wish to create artificial systems which imitate these task solving behaviors. Prominent examples of this are found in image and speech recognition systems, but there are many more popping up on a regular basis. To acquire the desired understanding the typical approach is to recognize a system as a whole and try to decouple individual components as well as possible and observe their effect on the system. Throughout history much work has been done in regard to human task solving and we choose a specific aspect of the human anatomy to specialize on. The nervous system is essential for the steering of various embodied biological agents, in particular the centralized processing unit, the brain. We shift our focus to the neocortex, the rind or coating of mammalian

brains. It is within this structure that this work attempts to link basic anatomical, structural connectivity, the automatic acquisition of this data as well as functional implications which can be found in simplistic models within some observed structural constraints.

## 1.2 A history in neuroscience

We start with the history of neuroscience and go back to the ancients to discover the origins of the understanding of brain function. In the Encyclopedia of Neuroscience Charles G. Gross provides a good overview of the early history of neuroscience [57]. A cynic might be surprised to learn that generally in ancient Egypt and Greece, despite all the warfare and knocking each others heads humanity has gone through, that at the time, the brain was not commonly considered the seat of intelligence, but more often the heart. This could stem from the fact that injuries to the heart are more frequently lethal than injuries to the head, and that humans tend to link their intelligence to their livelihood. At least one practical Egyptian investigation exists but it seems this was not sufficient to dissuade alternative beliefs. The Ancient Greek physicians also established some relations between neurological disorders and the brain which through observations of Alcmaeon over Hippocrates to Galen established the brain at least as the seat of sensory processing, were it was hypothesized that the ventricles in the brain played a fundamental role in brain function. With time there arose an awareness of the role of the peripheral nervous system in controlling muscle movements. The proposed mechanisms for brain function would remain as medical doctrine in the upcoming Middle Ages until the early Renaissance where public dissections were conducted by Vesalius and there was a gradual shift to more mechanistic views established in investigations of reflexes by Descartes. Some two hundred years later first investigations applying electricity to nerve-muscle preparations by Galvani led to the notion of bioelectricity in animal nerve fibers. A more established principle of regional cortical specialization was supported by Broca and shortly after by Wernicke. It would take more than another hundred years after Galvani's experiments for the concept of nerve cells and their manner of connectivity to arise, making use of light microscopy and the highlighting of single neurons using Golgi's silver stain, extensively applied for the neuron depictions skillfully drawn by Cajal.

It could be said that the modern era of neuroscience was initiated by Camillo Golgi's discovery of a silver staining method which completely filled a random subset of individual neurons when injected into the brain [54]. Santiago Ramón y Cajal recognized the value of Golgi's staining method, expanded it in his own right, and provided his own meticulous studies and images of the central nervous system [185] (an example is provided in fig. 1.1). The anatomical structure of various neuron types can be associated to both authors and their individual works justify their receiving the Nobel Prize in 1906, even though both vehemently disagreed on the network structure, with Cajal supporting the notion of neurons as individual, separate units and Golgi in favor of the reticular theory, where neurons were considered part of a continuous cell network. Golgi definitely provided the spark which kindled Cajal's scientific fire that would allow him to become the father of modern neuroscience. Something else that can be appreciated about Cajal's work, is that despite the technical limitations of

the time in investigating neural tissue at the cellular level, is his gift at interpreting the data he found. Over time, many of the hypotheses would be confirmed in experiments.

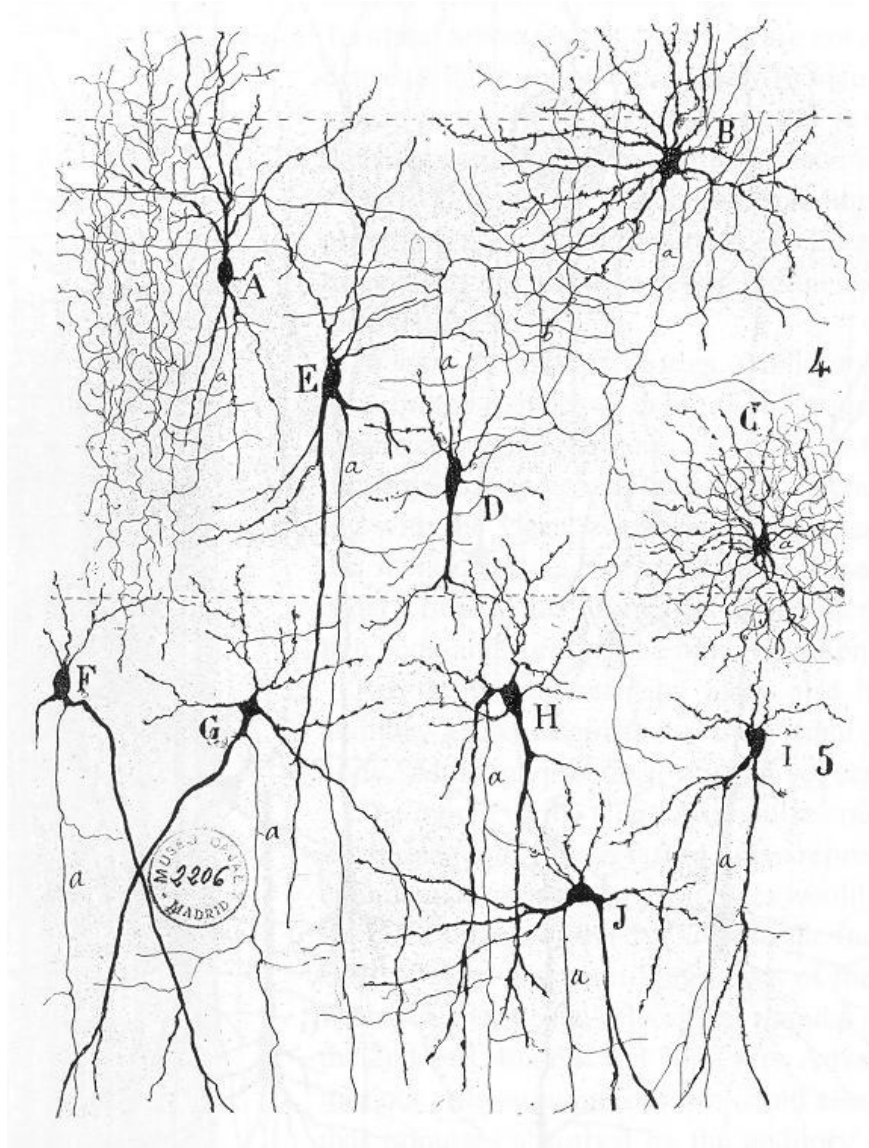


Figure 1.1: Drawings of different neurons by Cajal

What followed was a progression which would gather together the research in regards to nervous tissue and establish neuroscience as its own field of science. A selection of noteworthy contributions can be assigned to:

- Meynert who first noticed differences in the layering of the neocortex in different brain regions [110]
- Brodman who mapped the cerebral cortex into regions based on cytoarchitecture (distribution and form of cells in the cortex) [19]

- Sherrington who investigated integrated neuron activity in regard to reflexes and promoted new ideas of synaptic connectivity [153]
- Kennard who introduced the principle of critical periods of increased neural plasticity during development [91]
- Pitts and McCulloch who offered a first formal mathematical model for neurons within a network [105]
- Hebb who proposed a theory for synaptic learning [64]
- Hodgkin and Huxley who investigated the propagation of action potentials in the giant squid axon and would provide a mathematical model for these action potentials [71]
- Hubel and Wiesel who for the first time mapped the receptive fields of neurons from the cat visual cortex [79]

### 1.3 Cortical Microcircuits

At this point we choose to focus on a feature particular to mammalian brains: The neocortex (Latin for new rind or bark). The neocortex is a form of cerebral cortex, a typically layered strip of gray matter (nervous tissue with a high density of somata and glia cells) enveloping the cerebrum. The neocortex classically contains six layers, though there is some ambiguity in the distinction of these layers in different brain areas like motor cortex or species like cetaceans [72], or considering the proisocortex between neocortex and periallocortex. This layering was observable in early histological studies and Meynert [110] noticed that different aspects of this layering varied based on location in the cortex, as well based on species investigated, as highlighted by the Nissl stained tissue in fig. 1.2. Brodmann made a mapping of the cerebral cortex which distinguished areas based on their cellular composition and layer thickness [19]. Brodmann believed these anatomically based areas were simultaneously functionally distinct entities.

Remembering Cajal's drawings of individual neurons, and considering the differences of axon branching patterns depending on cortical depth and region, the question of the form of local neural circuits arises. Is there some unifying computation within the neocortex which requires some prespecified connectivity pattern of neurons within the layers? In recording studies in the somatosensory cortex of the cat, Mountcastle [116] observed that recordings within the same vertical column shared functional properties. This led to the idea of an elementary functional unit within the neocortex [133], and the resulting concept of cortical microcolumns. In a later study Powell measured the number of neurons under a unit surface of cortex and made the intriguing observation (which would remain an item of controversy to the current day and age), that all observed species and areas, except for primate visual cortex had a roughly constant number of neurons [143], which might hint at a very uniform production of neurons in the neocortex during development.

At a similar time to Mountcastle's and Powell's microcolumn measurements, Hubel and

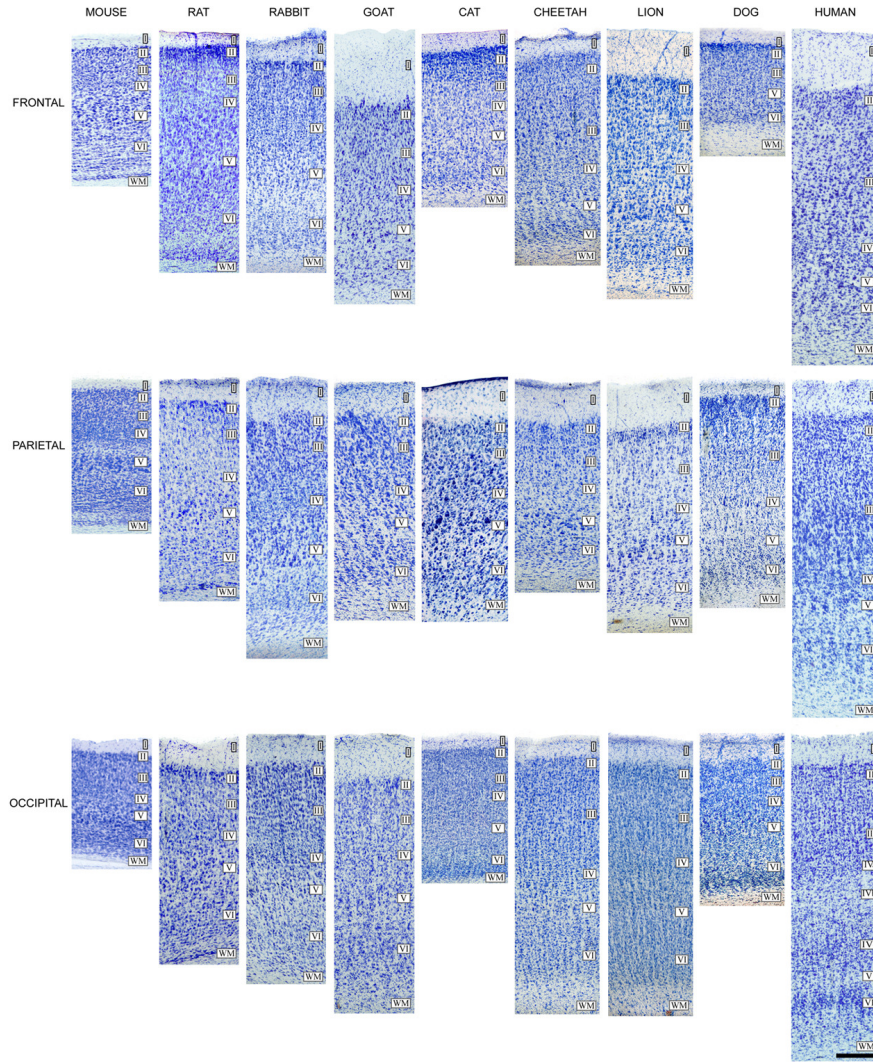


Figure 1.2: Layering of the neocortex as seen with Nissl staining. Images are from different mammals and different brain regions. Individual images share the same scale. Image from [33]

Wiesel were creating a functional mapping of the orientation responses of neurons in cat visual cortex [79]. It would appear that each location in the visual cortex had the same functional potential, but the spread of recurring orientations was in the range of hundreds of microns, similar to ocular dominance columns, which was a much greater expanse than the 25-30 microns of the microcolumns of Mountcastle and Powell. It would seem that a functional primitive for the neocortex would have to be able to reconcile these two observations. A very basic model of functionality, or more a model of population activity, was given in the canonical microcircuit model by Douglas et al. [38]. This model was used to describe average firing responses in cat visual cortex, and combines two recurrently connected excitatory neuron populations and an inhibitory population which is recurrently connected to

both. The main distinction between the two excitatory populations, is that one population receives input from an external source, in this case the visual thalamus (lateral geniculate nucleus - LGN). To expand upon this very simplistic model, great effort was invested by Binzegger et al. to quantify a connectivity mapping of the various neuron populations in all cortical layers of the cat visual cortex [12]. In essence, it was a return to measuring the stereotyped circuits proposed by Cajal [186], except this time with upgraded experimental methods and materials. Such a connectivity mapping was one more step towards a canonical description of neocortex.

## 1.4 Mapping the Matrix

Quantitative mapping of cortical circuits could be said to have started with the mapping of Binzegger et al. [12]. The connectivity diagram for the cat visual cortex was built upon assumptions and measurements presented in the following:

The connectivity mapping provided the average number of incoming and outgoing synaptic connections between the most prominent neuron types in each cortical layer in the cat visual cortex. This accounted for all synapses of the local circuit, but did not consider synapses from external areas (except for synapses coming from the LGN). To estimate the number of synapses formed by each neuron, the axonal arbors along with boutons and dendritic arbors of multiple example neurons which had been previously filled with Horseradish Peroxidase (HRP) were reconstructed in 3D (see fig. 1.3). This provides an estimate of the average number of outgoing synapses a specific type of neuron forms within all cortical layers, as well as an estimate of the potential number of synapses this neuron type might receive in all layers. Given the total number of neurons in each layer with the relative frequencies of each neuron type, and assuming, where not known better, that outgoing synapses are distributed according to the presence of dendritic arbors (also known as Peters Rule [130], [15]), an estimate on the average number of synapses formed between neuron types for each layer can be calculated. As a cross reference, estimates of the total number of synapses of in each cortical layer can be made using electron microscopy (EM).

What were the requirements for such a mapping, and what methods were required, which had not been present in Cajal's time? During the 1950's the electron microscope was being used to investigate neural tissue and it was Gray [56] who provided a deeper description of the axon-dendrite junction, the synapse (See fig. 1.4). He also made a distinction of synapse types which would be distinguished as inhibitory and excitatory synapses.

While Golgi's stain highlights single neurons, it does not do so in a predictable way, in particular it is not clear which neurons will be labeled. Injecting single neurons with a tracer during physiological recording as in [90], [92] allowed for targeted reconstruction of single neurons, while also preserving information on functional properties of that neuron.

The connectivity mapping of cat primary visual cortex was the starting point of this thesis. When speaking about canonical microcircuits, the principle of a canon needs to be established. What are the shared principles of connectivity between cat V1 and other cortical areas in other mammals? In this context mouse primary auditory cortex (a sensory/input



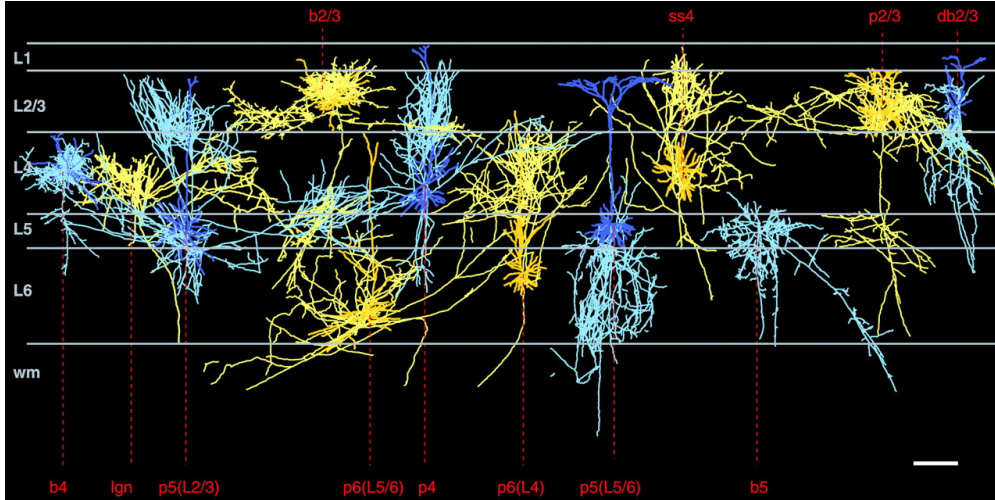


Figure 1.3: Various 3D neuron reconstructions from cat visual cortex, in relation to the layer boundaries. Dark blue and dark yellow highlight dendritic arbors. Light blue and yellow highlight axon arbors. Scale bar corresponds to  $300\mu\text{m}$ . Image from [12]

region) and mouse primary motor cortex (a motor/output) region were investigated in regards to their total number of synapses per cortical layer, to be used as a cross-reference for similar connectivity diagrams for these two regions in the mouse. In this regard, it is useful to look at more recent developments in processing and analyzing neocortical tissue.

## 1.5 Connectomics

The concept of a canonical microcircuit has a sibling who goes by the name of connectomics. Connectomics is the study of the complete connections of an organism's nervous system, in particular of its central nervous system. Modern connectomics is the conceptual offspring of two scientific achievements. The first is the complete connectivity mapping of the nematode (roundworm) *Caenorhabditis elegans* (*C. elegans*) [184]. This small invertebrate possesses 302 stereotyped neurons (for the hermaphrodite) with around 5000 chemical synapses, 2000 neuromuscular junctions and 600 gap junctions, of which most were mapped in this initial undertaking. The second parent would be the Human Genome Project [98]. This project was a massive, collaborative project with the goal to sequence the whole human genome. It was also the endeavor into a new way of conducting science termed discovery science [1] with the idea that measuring a system will result in later insights as opposed to the old-established method of hypothesis driven science where measurements are made in support or refutation of a previously formulated hypothesis.

The primary goal of connectomics as the offspring of these entities is to provide a description of the complete synaptic connectivity of the human central nervous system. The belief is that understanding, insights and discoveries will follow naturally from such an approach. That this ideal has its caveats is known [101], [115], [100]. How well these caveats may or

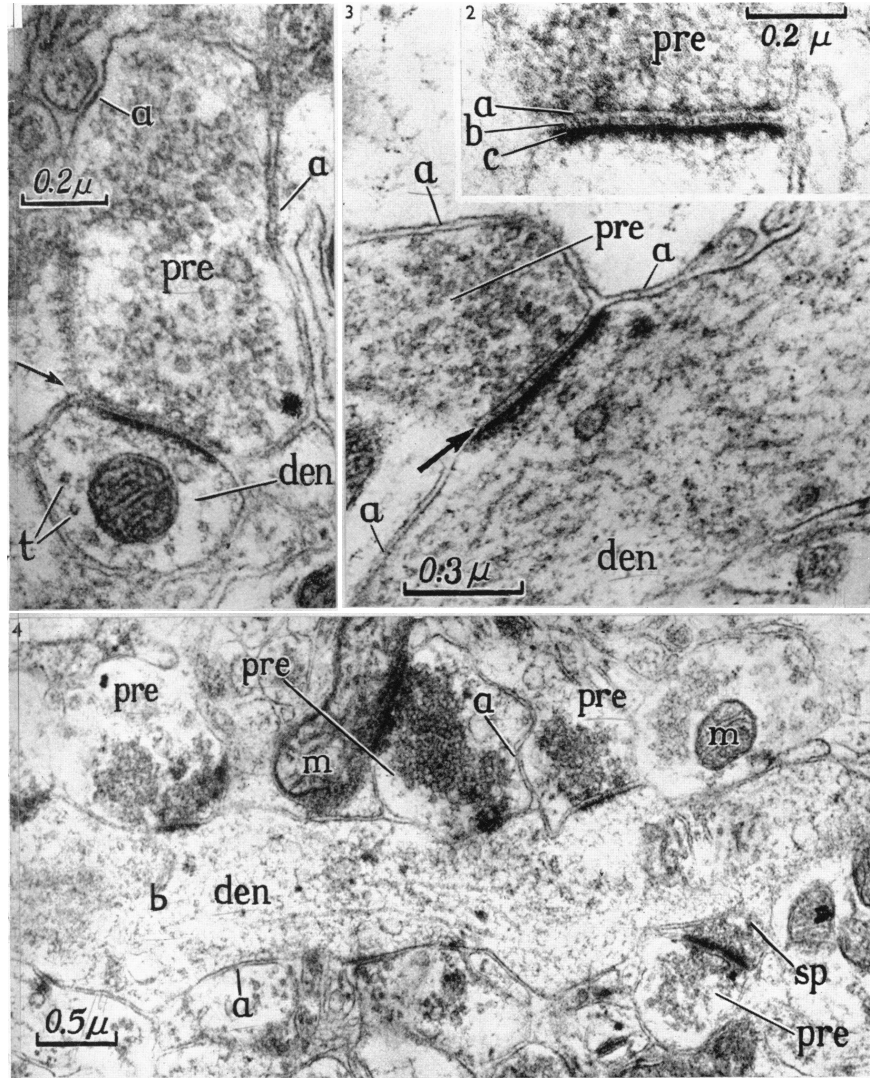


Figure 1.4: Micrographs containing examples of synapses. Image taken from [56]

may not be addressed is issue of another discussion. While connectomics and the classical measurement of cortical microcircuits are related, we should establish in which ways, and determine what connectomics needs to provide for us to gain a better understanding of cortical circuits.

A full connectome would provide for a species at least one example of intra-layer connectivity, from which coarser statistics could be procured for many brain areas at once within the same individual, which would greatly reduce measurement differences or errors based purely on methods of data acquisition. This would be a good basis to search and discover canonical elements of cortical connectivity, as long as we do not forget that synaptic connectivity changes massively during development ([188]) as well as significantly during further maturation, but even during the diurnal cycle ([129],[82], [83]). And how much variance can

we expect between individuals? While the mean connectivity statistics might not fluctuate as much, this still needs to be established. Either we need many connectomes for various conditions, or it needs to be established that general variability is lower than could be feared. A full connectome definitely will have its uses. While waiting, however short or long that may be, for said connectome we can consider ways to investigate stated variabilities.

With large scale collaborative projects like the Human Brain Project (HBP) [104] or the Brain Research through Advancing Innovative Neurotechnologies (BRAIN) initiative ([85]) we can expect upcoming advances in tissue preparation and staining, imaging techniques and automated data analysis. For example genetic markers for light microscopy like Brainbow ([182]), or markers which are visible at the EM level like miniSOG [154],[120] have made advances in recent years and can prove invaluable for counting studies once the methodology becomes more stable and established. An alternative to EM for sparsely labeled synapses would be to use super resolution imaging ([155]) Other examples would be the use of cryo-fixation [95] to better preserve the tissue's ultrastructure. Serial Blockface Scanning Electron Microscopy (SBSEM) and related approaches have become the established methods for imaging vast amounts of neuropil with the goal of obtaining dense reconstructions of these [179]. Higher throughputs can become possible when combined with approaches like Multiple Beam Scanning Electron Microscopy [39]. These approaches have laid the groundwork for various reconstruction studies for various brain regions and species such as various regions of the *Platynereis* [135], the *Drosophila* nervous system [164], [150], [109], the zebra fish olfactory bulb [178], [177], the mouse retina [65], mouse somatosensory cortex [88], rat hippocampus [114] and others. Another approach towards acquiring the mouse connectome, without the use of EM, is provided by the Allen Institute for Brain Research [122].

The computational hurdles to be overcome for connectomics, such as the necessity for improved imaging techniques, image/data alignment and storage, or automated image segmentation and more, are already very welcome to current approaches for circuit analysis. An array of automated and semi-automated neurite reconstruction tools and frameworks has arisen in the recent years ([89], [93], [40], [11], [136], [166], [14], [189]) as well as specialized architectures for synapse detection ([113], [158], [3], [119], [10], [131], [96], [77], [37]). Despite all these advances, the path towards an automated connectome construction is a rocky one. A second part of this thesis aims towards providing new building blocks for image segmentation which can be integrated into such EM segmentation frameworks.

## 1.6 Neural Networks - Cycling Data and Models

The common approach in physics to gaining a better understanding of complex, dynamic systems, is to create a descriptive, formal model of the system, which can be used to make predictive statements of real implementations of the system modeled. It would seem that the time required for the first models of neuron networks and descriptive models of single neuron firing behavior to arise in the 1940s-1950s is an indicator for the sheer complexity of neural networks and their elusiveness towards simple, analytically tractable, yet reasonably powerful descriptions.

With the dawn of the digital computers during World War II in the early 1940s, initial work regarding modeling and description of neural networks started. First prominent work in this direction is found in a mathematical paper from McCulloch and Pitts [105] using an all-or-nothing, state-based firing model with threshold neurons. In this model neurons are described as being in the binary state of active or non-active. The neuron contains an internal threshold value which is compared against the sum of the synaptic weights of all other neurons in the system to determine if the neuron will be active or not. The state of such a network is updated in discrete time steps. This is the basis for the perceptron ([145]), which, with an appropriate learning rule, becomes a simple classifier for data sets which are linearly separable in regard to a given labeling. Research on perceptron networks receded when limitations of the solving capabilities of single layer perceptron networks were shown ([112]), in particular by the incapability to solve the exclusive-or (XOR) and related problems. A few years later the discovery of the backpropagation algorithm ([183]), which allowed for the training of multi-layer perceptron networks, rekindled interest in artificial neural networks. Later research finally took a step in the direction of recurrently (self-) connected networks with Hopfield networks which were popularized by Hinton ([74]). A later relative is found in restricted Boltzmann machines ([159]), which probabilistically update the binary states of neurons found in Hopfield networks. Many of these algorithms remained greatly unused until the early 2000s, mostly for the lack of computational power and lack of large enough training sets to get state-of-the-art results. In the 2010s deep neural networks and recurrent neural networks received prominence due to competitive efforts of Schmidhuber and Hochreiter ([70],[118],[26]). These networks would be improved upon and give rise to convolutional neural networks and long short-term memory (LSTM) networks.

Investigations using linearly rectified units regarding whole network stability were performed by Hahnloser et al ([58],[59]). Using linear rectified units, the resulting network can be analyzed theoretically by investigating linear dynamics of subpopulations of neurons. Here it was shown that certain networks can have multiple stable steady states, depending on the input, which can be accessed from different sets of input signals to the network. This is a good basis for the deciphering of networks with multiple attractor states.

A neuron model which retains more of the temporal firing properties of biological neurons is found in spiking neuron models. The simple case of integrate and fire neurons, in which a model neuron has an internal integrator variable which adds together incoming inputs over time and emits a pulse event when the variable reaches a certain threshold and is then reset to zero, allows the coupling of neurons to move the system toward a cyclic attractor after a short amount of time ([75]). It has been observed in physiological recording, as well as simulations ([174]) that neurons can exhibit non-periodic firing dynamics, which are related to asynchronous network states ([137]). It is also possible to observe global oscillations of the whole system based on the total strength of inhibition ([21], [22])

Already during Rosenblatt's time when formulating the perceptron, there seemed to be a discrepancy between pure network theory and mechanisms proposed based on experiments by physiologists somewhat lacking formal description. Alas, disagreements between biological measurements and models are often not small, but occasionally models provide ideas for

extended experiments, and some experiments provide closure or inspiration for models.

The final work in this thesis builds upon and expands the linear rectified units as presented by Hahnloser et al ([58]), such that a set of typical restrictions found in biological neural networks can be accounted for by this expanded model.

# Chapter 2

## Synapse Densities

### Summary

One of the initial goals of this work was to find canonical elements in the cortical connectivity patterns of mammalian neocortex. Such work had already been done for the cat primary visual cortex(V1) by Binzegger et al [12]. To look for canonical elements, the goal was to see what properties are preserved when considering a different species and a different cortical area. For this reason mouse primary auditory cortex(A1) and primary motor cortex(M1) were investigated. To be able to estimate the completeness or non-completeness of resulting connectivity diagrams in the style of [12] we measured synapses densities in both areas.

Synapse densities were measured in A1 and M1 for one mouse each. Synapse counts were obtained using the disector method with images acquired using transmission electron microscopy(TEM) and scanning transmission electron microscopy(STEM) at a magnification of 13'500. An equivalent volume of 2'500  $\mu\text{m}^3$  and 1'200  $\mu\text{m}^3$  were scanned for synapses in A1 and M1 respectively. Sampling sites came from all over A1 and were drawn from the forelimb representation in M1, covering all cortical layers in both areas.

The synapses were identified by a single human annotator by comparing electron microscopy micrographs of subsequent tissue slices. A small random subset of these micrographs were proofread by other annotators for cross-reference. Criteria to recognize synapses are given below.

The following sources of error need to be considered for correction of the measured densities: tissue shrinkage of the cortex due to chemical fixation, blade compression during the ultra-thin section cutting, uncertainty of the precise section thickness and uncertainty due to only having a finite number of samples.

For both areas higher synapse densities were observed in the superficial layers, particularly in layer 1. Synapse density decreases in layers 4-6 in particular in M1 where densities are only half of those of the superficial layers. We compared the synapse densities with the neuron densities in the same mice in the same areas and found that there are on average 8'900 synapses per neuron in A1 and 7'700 synapses per neuron in M1, which highlights the vast amount of synaptic integration of individual in the mouse neocortex.

## 2.1 Tissue Preparation

Two adult B6/C57 Mice were used to acquire the tissue for the synapse density estimates. The animals were anesthetized with pentobarbital and perfused transcardially with saline, followed by a solution of paraformaldehyde (4%), picric acid (15%) and glutaraldehyde (2.5%) for tissue fixation. For both mice the whole brain was cut into 80  $\mu\text{m}$  slices which were alternately stained with Cresyl Violet and osmium tetroxide.

Nissl stained sections used for the identification of boundary layers as well as cell counts were put in a solution of acetic acid (1 min) and then treated with 0.1% Cresyl Violet (5-10min). The sections were dehydrated using an ascending series of ethanol and cleared with Chloroform and later with Xylene.

Osmicated sections for EM were treated with 1% osmium tetroxide (40 min) and washed in a 0.1M PB buffer. The sections were then dehydrated using an ascending series of ethanol and propylene oxide (including treatment in 1% uranyl-acetate in 70% ethanol) and then flat mounted in a Durcupan ACM resin.

The Nissl stained sections were compared with a mouse atlas [127] to localize the cortical areas of interest. Comparing the Nissl stained sections with this mouse atlas, A1 and M1 were localized and area and layer boundaries for two sections in A1 and for one section in M1 were drawn according to the cytoarchitecture described by [24]. For each section the neighboring osmium stained section was cut perpendicular to the pial surface into ultra-thin 40 nm sections which contained all cortical layers and were collected on pioloform-coated copper grids in groups of four. Starting at a random offset, pairs of sections at a distance of 80 nm were selected at regular intervals of at least 2  $\mu\text{m}$  to be used for physical disector counts. These sections were overlaid with a regular grid of sampling points with a grid spacing between 30  $\mu\text{m}$  and 50 $\mu\text{m}$  with a random initial offset and orientation.

## 2.2 Electron Microscopy

Using TEM we imaged the sampling grid points at a magnification of 13'500 and overlaid 5  $\mu\text{m}$   $\times$  5  $\mu\text{m}$  boxes for counting. Alternatively we fully imaged both sections at 2.5 nm per pixel using STEM and were able to pinpoint and investigate the grid locations directly. It took longer to fully image a full section using STEM than individually imaging grid points and required more storage, but human work time can be saved, as no individual images need to be taken any more. It is also no longer necessary to revisit the electron microscopy(EM) in the case it would be necessary to look at nearby locations within the tissue. No systemic difference was noticed between the synapse counts obtained using TEM or STEM, and differences in density estimates due to the imaging technique will be expected to be very small in comparison to uncertainties such tissue shrinkage and tissue compression due to cutting. If a sampling location showed any damage such as tears or folds, it was disregarded for the counting. However, opposed to [151] or [51], locations with blood vessels and cell bodies were considered, so that the complete synapse density statistics are preserved, also in regards to ratios of inhibitory and excitatory synapses. The physical disector method [163],

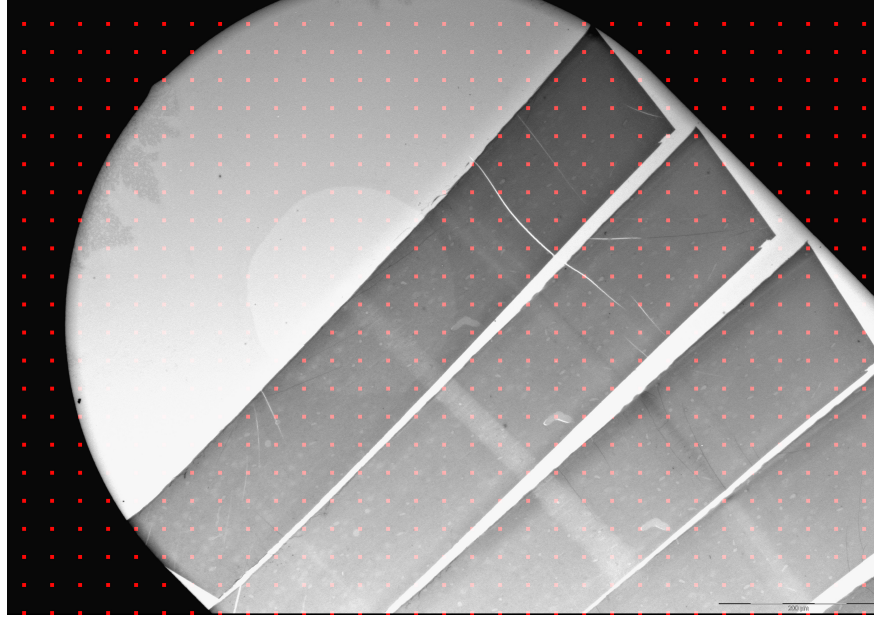


Figure 2.1: A light microscope image of a copper grid containing ultra-thin 40 nm tissue slices from mouse A1. The red dots which lay over the section used for sampling correspond to sampling locations. The scale bar is 200  $\mu\text{m}$

as opposed to a volume counting method such as FIBSEM, was used for counting to provide a greater number of varying sampling sites which also better captures synapse statistics in the proximity of myelinated axons, blood vessels as well cell bodies. Fig. 2.2 shows an example of a typical disector pair.

Synapses were identified using standard criterion [27]. Fig. 2.3 shows an example of three excitatory synapses in a single section.

Synapses were classified according to [27] by the presence of presynaptic vesicles and a darkening of the membrane on both sides of the synaptic cleft. Asymmetric synapses are characterized by their electron dense post-synaptic density. The visibility of the synaptic cleft is a criterion according to [27]. However, to avoid a bias in the counting and to avoid missing synapses which are at an oblique angle towards the cutting plane, classification of a synapse was considered, also if the synapse cleft was not clearly visible.

Synapses which appeared in only the first section, but not in the second were counted. Synapses which touched the right or the top boundary of the grid box were counted and disregarded if they touched the left or the bottom boundary. In accordance to [32] the counting procedure was repeated from second section to first, to double the number of sampling sites.



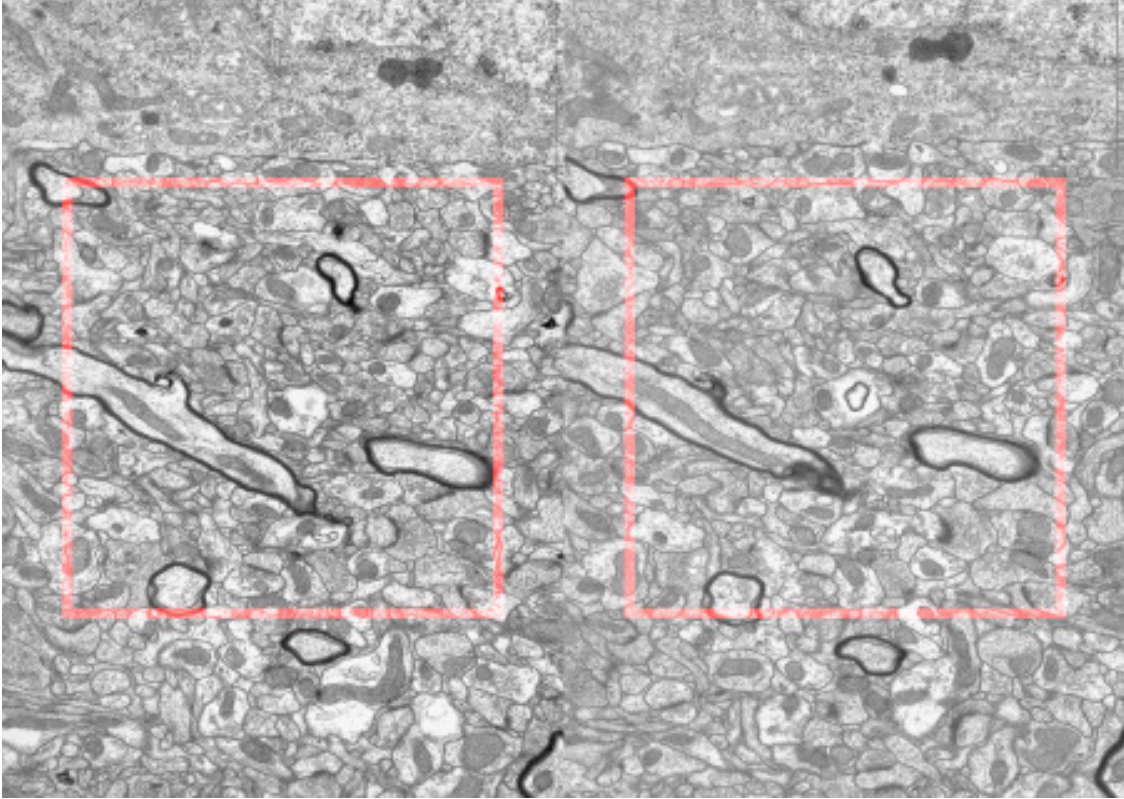


Figure 2.2: Cortical ultrastructure (mouse A1, layer 4). Myelin sheaths and mitochondria can be observed to slightly change their structure between the two sections. The red square corresponds to an area of  $5 \mu\text{m} \times 5 \mu\text{m}$ . The two images are 80 nm apart.

## 2.3 Results

In total we sampled counting sites from eight grids in A1 and from four grids in M1. Each counting site has an equivalent volume of  $2 \mu\text{m}^3$ . The distribution of counting sites in both areas and in all layers can be found in the table 2.1.

| Layer | A1 (8 grids) | M1 (4 grids) |
|-------|--------------|--------------|
| 1     | 112          | 72           |
| 2/3   | 250          | 102          |
| 4     | 184          | 20           |
| 5     | 478          | 286          |
| 6     | 252          | 110          |
| Total | 1276         | 586          |

Table 2.1: Total number of sampled disectors throughout the cortical layers of A1 and M1

In total an equivalent volume of  $2'552 \mu\text{m}^3$  was sampled in A1 and of  $1'172 \mu\text{m}^3$  was sampled in M1. We measured the thickness of the cortical layers at the boundaries and

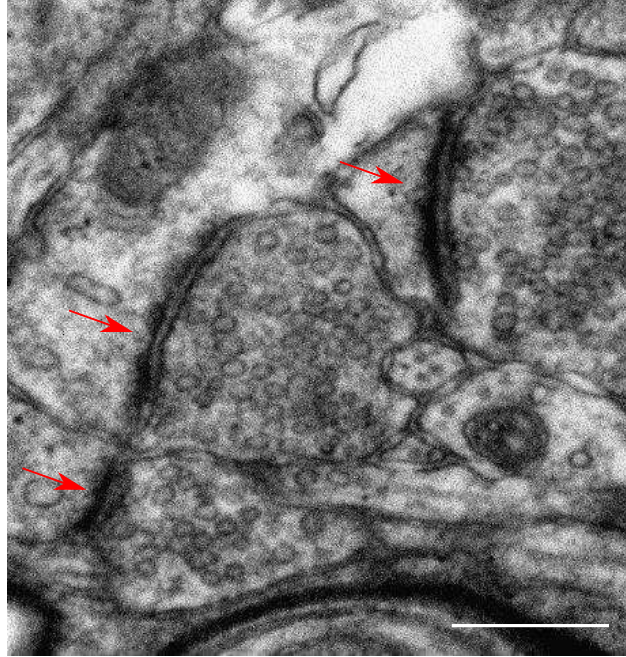


Figure 2.3: Red arrows indicate asymmetric synapses. The white bar corresponds to 250 nm

center of the osmium sections used for the synapse counts. Layer thicknesses in  $\mu\text{m}$  of A1 and M1 are presented in fig. 2.4.

Layer L1 is comparably thick for both areas. The combined thickness of layers L2/3 and L4 is also similar for both areas. The ratio of L5 to L6 is similar for both areas

For both areas we observed an average compression of the ultra-thin sections of 25% perpendicular to the cutting direction. This compression is factored out in the synapse densities presented in fig. 2.5.

Interestingly, the synapse densities show a very similar pattern throughout the cortical layers for both areas. Densities are highest in layer 1, where there are few neuron cell bodies and decrease slightly towards layers 2 and 3. Density drops noticeably for layer 4 and deeper layers, possibly due to the increased presence of myelinated axons as well as larger cell bodies. Total average synapse densities are determined more strongly by the deep layers, due to the proportions of layer thickness. For all layers we observe a trend of A1 having a higher synapse density.

Also presented here in fig. 2.6 are neuron density estimates provided by Rita Bopp using the disector method on semi thin sections from the same mice.

As with the synapse densities, neuron densities in A1 and M1 show a similar pattern throughout all cortical layers. Again, there is a trend for higher density counts in A1 for all cortical layers. Layer 1 is characterized by a very low neuron density. Layers 2 and 3 have high cell densities with small cells. Density decreases for layer 5 where there are fewer but larger cells. Combining the neuron densities and the cortical thickness we can estimate the number of neurons under  $1\text{mm}^2$ . There are  $1.09 \cdot 10^5$  neurons under  $1\text{mm}^2$  in A1 and  $1.17 \cdot 10^5$

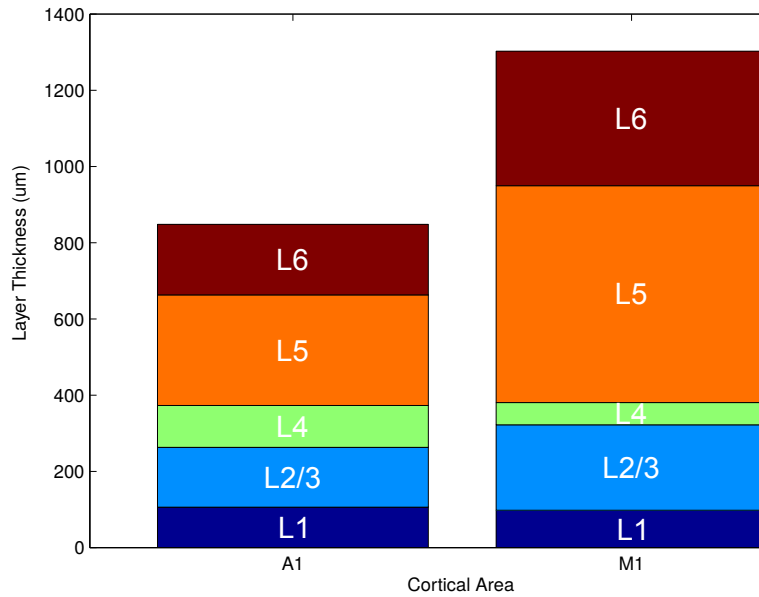


Figure 2.4: Layer thicknesses in mouse A1 and M1. The vertical axis corresponds to thickness in  $\mu\text{m}$ . Layers are indicated by L and the layer number

neurons under  $1\text{mm}^2$  in M1.

These values are close to one another and well within the range of values given by [143]. It should be remarked that it is unclear if and what effect there is on these numbers due to shrinkage during tissue fixation, especially since cortical shrinkage can be non-uniform [95]

Combining synapse and neuron densities allows for the evaluation of the number of synapses per neuron in an area, which can be used as an indicator for the integrative power of cortical neurons. In A1 we find 8'930 synapses per neuron in A1 and 7'680 synapses per neuron in M1. These values are well within the range of those found for the mouse in areas 6, 8 and 17 [151]. The high number of synapses per neuron will be important when considering input amplification in model neuron networks as in chapter 5. A review of synapse and neuron densities for different species and areas is provided in appendix A

Symmetric and asymmetric synapses, which are believed to correspond to excitatory and inhibitory synapses were also measured. It is observed that long range projecting neurons are generally excitatory. Since almost no long range inhibitory projections are measured, the large majority of inhibitory synapses are considered to be formed by the local circuitry. Therefore inhibitory density estimates provided by neuron reconstruction should be the same as symmetric synapse densities measured using the disector method. The fraction of symmetric synapses observed in all layers is presented below. There is no obvious pattern for the fraction of symmetric synapses within the layers of between the areas (fig 2.7). We find an average of 14.1% symmetric synapses in A1 and 11.8% symmetric synapses in M1. These values are not far apart, but there is still a statistically noticeable difference. It should be

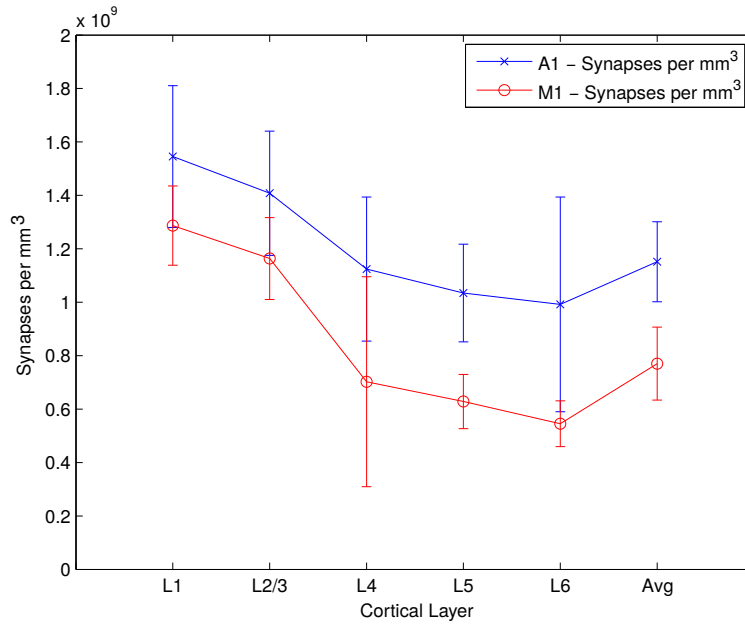


Figure 2.5: Synapse densities with standard error bars in mouse A1 and M1. The vertical axis corresponds to number of synapses per  $\text{mm}^3$ . Layers are indicated by L and the layer number on the horizontal axis. Avg is for the average synapse density within the whole cortical area

noted at this point, that both mice had good ultrastructure, but that, in particular asymmetric synapses had a slightly different appearance in both tissues, independent of cortical area (fig. 2.8).

Fig. 2.8 shows five typical examples of asymmetric synapses. The top row shows examples from A1 in the first mouse. The bottom row shows synapses from A1 from the second mouse where densities measurements for M1 were made. Both mice had been fixed with the same protocol, but some differences can be noticed between synapses. The second mouse has more prominent post-synaptic density(PSD), more irregularly shaped vesicles and irregularly shaped synaptic clefts. The difference in PSDs could be a reason for the classification threshold between symmetric and asymmetric synapses to shift slightly in the favor of one type or another.

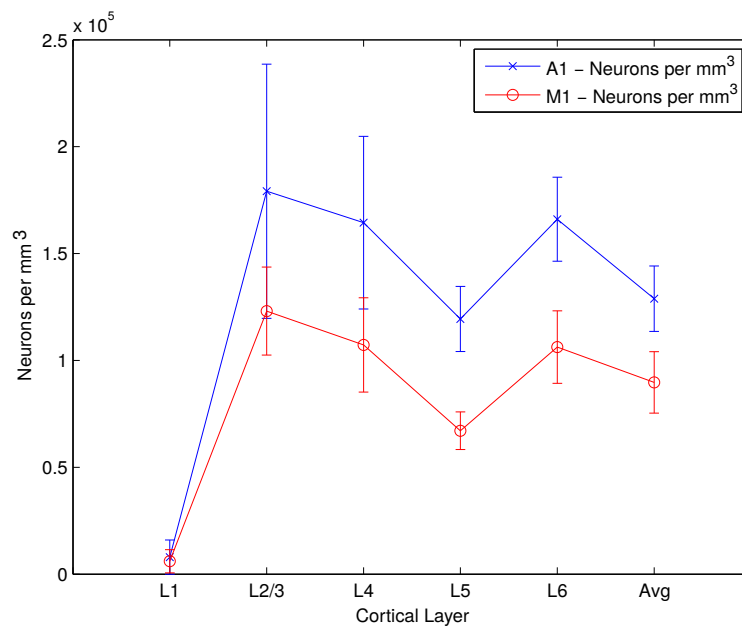


Figure 2.6: Neuron densities with standard error bars in mouse A1 and M1. The vertical axis corresponds to number of neurons per mm<sup>3</sup>. Layers are indicated by L and the layer number on the horizontal axis. Avg is for the average neuron density in the whole cortical area

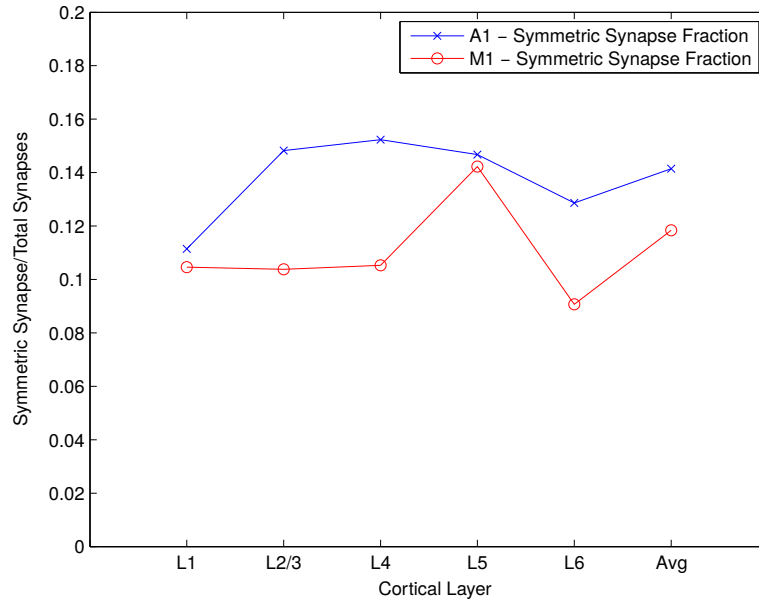


Figure 2.7: Fraction of symmetric synapses in mouse A1 and M1. Layers are indicated by L and the layer number on the horizontal axis. Avg is for the average symmetric synapse fraction within the whole cortical area

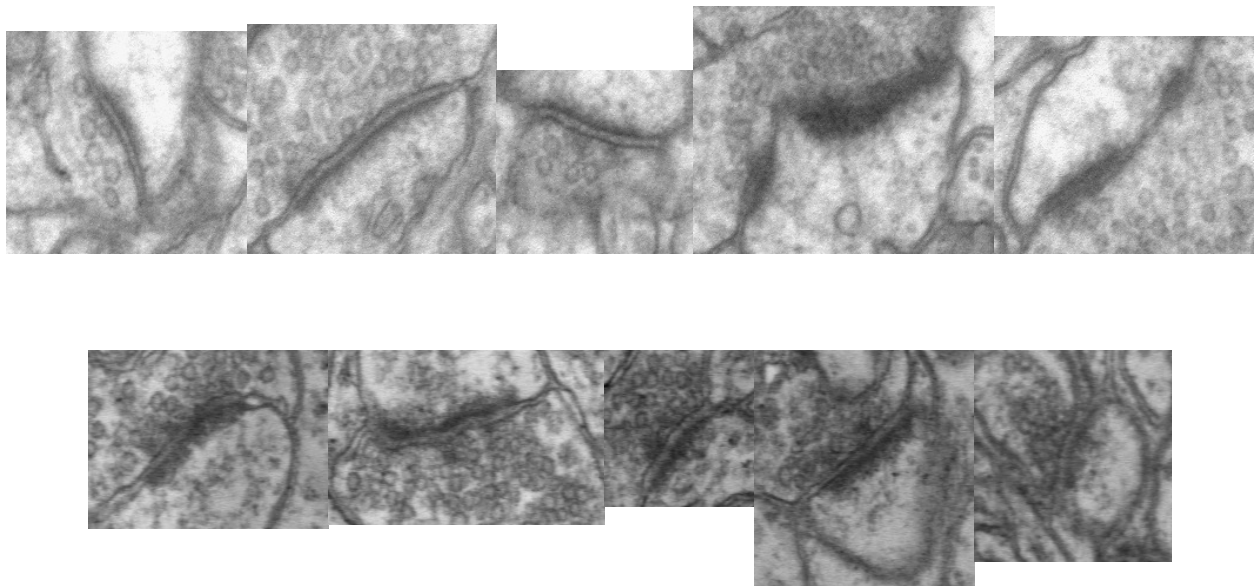


Figure 2.8: Examples of asymmetric synapses in mouse A1. Top row: from mouse 1. Bottom row: from mouse 2

# Glossary

**A1** primary auditory cortex. 19–26

**EM** electron microscopy. 20

**FIBSEM** focus ion beam scanning electron microscopy. 20

**M1** primary motor cortex. 19–26

**PSD** post-synaptic density. 25

**STEM** scanning transmission electron microscopy. 19, 20

**TEM** transmission electron microscopy. 19, 20

# Chapter 3

## Random Forests and Classification

### Summary

Identifying large quantities of synapses under the electron microscope is a laborious task. A human synapse counter commonly requires one or two weeks of training to become initiated in this task, and even then some counters will remain challenged by the task. It would be useful if the time invested in training a human counter could be used to train an automatized agent with the sole purpose of counting synapses. Although great advances have been made in automated image recognition in the recent years, this recognition power does not come without caveats. The following will hopefully provide some insights gained from the perspective of a human counter as well as a "teacher" of automated learning systems. It might be a combination of the understanding of human relation forming as well as the consistency and efficiency provided by automatic systems that leads in the direction of reliably solving tasks such as the synapse counting task at hand.

The form of automatic recognition used for this thesis was based on pixel-wise classification. More specifically for synapse counting, the goal of the initial classifier is to specify for each pixel of an EM image if that pixel is an asymmetric, a symmetric or no synapse. For the sake of simplicity and understanding, a random forest classifier [16] was used and partially adapted for this classification task.

This chapter will provide an overview of the random forest algorithm and some potential adaptations to the decision tree splitting procedure, as well as some clustering and compression algorithms that make use of the structure of the random forest algorithm.

### 3.1 Introduction of Random Forests

The random forest algorithm [16] was developed as a classification and regression tool. The goal of classification is to assign a label to an object based on the object's descriptive attributes, features, dimensions or however we may want to term them. Therefore, to understand the classification task, it is necessary to understand what objects are to be considered, what attributes are useful for the description of these objects and what approaches there



are to set up rules or guidelines which allow these objects to be distinguished based on their descriptive attributes. The purpose of the random forest algorithm is to find such a set of rules, based on a collection of examples for each class we chose to define.

Specifically, a random forest is an ensemble, or collection, of binary decision tree classifiers. At the root of a single the decision tree a yes or no question is asked about one of the attributes of the object. If the question is answered in the affirmative, then a new question about one of the attributes of the object must be answered. If the answer is negative a different question about one of the objects attributes must to be answered. The sequence, stored in a tree structure, of posed questions and corresponding answers determine the estimate of the class of an object which is specified in each leaf node of the tree. The decision tree approach for classification has an intuitive understanding as a series of yes and no questions to specify a particular object. Random forests expand this approach by letting a collection of decision trees ask their questions about an object and then vote on the decision of the final class for each object. The difficulty of this approach doesn't lie in the classification procedure per se, but in determining the set of questions that each decision tree should be asking, given a finite number of examples to base these questions on. This is the key of the random forest algorithm and will determine the performance of the classification.

## 3.2 Definitions and Algorithms

Given a set of training examples a random forest can be trained. Each example “object”  $\mathbf{x}_i \in \mathbb{R}^d$  comes with a classification label  $y_i \in \{1, \dots, c\}$ , where  $d$  is the number of descriptive attributes, here termed features (or dimensions, or feature dimensions), and  $c$  is the number of possible object classes. It should be noted that any of the feature dimensions could be exchanged with a non-ordinal, categorical dimension.

Decision trees are built up of nodes of the following structure:  $o_{i,s} = o_{i,s}\{\theta, j, \mathbf{p}\}$ , where  $i \in \{1, \dots, t\}$  is the tree identifier with  $t$  the number of decision trees,  $s \in \mathbb{N}$  the node's index in the decision tree,  $\theta \in \mathbb{R} \cup \emptyset$  the decision threshold on the feature  $j \in \{1, \dots, d\}$  and  $\mathbf{p} \in [0, 1]^c \cup \emptyset$ ,  $|\mathbf{p}|_1 = 1$  the class label probability distribution for all objects which fall into this node (only specified in leaf nodes).

For an object or point  $\mathbf{x} \in \mathbb{R}^d$  each decision tree can be parsed in a recursive manner, checking if a final partitioning of the input space has been reached (see alg. 3.1).

Intuitively, a decision tree partitions the input space into axis-aligned boxes, and provides the class probability distribution for each box.

The combined(ensemble) decision tree prediction of the forest, is the mean value of the class probabilities of each leaf node the object to be classified ends up in for each tree, as given by Alg. 3.2. Decision trees are trained in Alg. 3.3. This is similar to the standard setup for decision tree learning in random forest, when the weight of each tree is equal for the classification vote. Also, in contrast to the standard procedure, direct probability values instead of the majority class are used in the leaf nodes. It has been observed that  $d_{trial}$  and the split function  $G$  can have a relevant effect on prediction accuracy [81], [142], [52], [176]). Common as well as new split criterion are presented in the next section.

---

```

1: function DTREE( $o_{i,s}, \mathbf{x}$ )
2:   if  $o_{i,s} \rightarrow \theta = \emptyset$  then                                ▷ check if a leaf node has been reached
3:     return  $\{o_{i,s} \rightarrow \mathbf{p}, s\}$                                 ▷ return class probabilities and the tree node index
4:   else if  $o_{i,s} \rightarrow \theta \geq x_{o_{i,s} \rightarrow j}$  then      ▷ else if below threshold: branch left and repeat
5:     return DTREE( $o_{i,2,s}, \mathbf{x}$ )
6:   else                                                                ▷ else: branch right and repeat
7:     return DTREE( $o_{i,2,s+1}, \mathbf{x}$ )
8:   end if
9: end function

```

---

Algorithm 3.1: Parse the decision tree

---

```

1: function VOTE( $\mathbf{o}, \mathbf{w}, \mathbf{x}$ )
2:   for  $i \in \{1, \dots, t\}$  do                                          ▷ go through all trees
3:      $\{\mathbf{p}_i, l\} = \text{DTREE}(o_{i,1}, \mathbf{x})$                                 ▷ get the class probabilities in the tree's leaf node
4:   end for
5:   return  $\sum_{i=1}^t w_i \cdot \mathbf{p}_i$                                        ▷ weighted sum of tree votes
6: end function

```

---

Algorithm 3.2: Ensemble voting for the decision tree.  $\mathbf{o}$  is the collection of decision trees and  $\mathbf{w} \in [0, 1]^t$ ,  $|\mathbf{w}| = 1$  a tree weight vector for the voting.

---

Each decision tree is trained with a randomly drawn subset of all the available training data, which makes the ensemble vote more robust. We can measure the out of bag error as the prediction error of all training examples based on all decision trees, where the example had not been used in the training of the decision tree.

A single decision tree in a random forest classifier is trained using only a subset of the available training data points. Conversely, for each training point a subset of the decision trees were trained without using that training point. We can obtain an unbiased estimate of the classification of each training point by having all trees which were trained without this training point vote to predict the training point's class. Applying this approach to all training points and averaging the resulting classification errors over all training points offers an unbiased estimate of the performance of the random forest classifier. This is basis of the out of bag error (oobe) of the random forest classifier [16].

### 3.3 Decision Tree Split Criteria

Most splitting criteria used try to minimize some measure of dissimilarity of the labels on both sides of the decision boundary. The two most commonly used measures of dissimilarity are the gini impurity and information gain.

The gini impurity is a measure of the frequency of mislabelling a randomly chosen element according to the subset label distribution (eq. 3.1)

---

```

1: function TRAIN( $\mathbf{X}, \mathbf{Y}, o_{i,s}$ )
2:   if ( $\max_{j \in \{1, \dots, c\}} \sum_{k=1}^{|\mathbf{Y}|} \mathbf{1}_j(y_k) = |\mathbf{Y}| \vee (|\mathbf{Y}| \leq r_{node})$ ) then    ▷ single class node, or too small
3:     for  $j \in \{1, \dots, c\}$  do                                          ▷ go through all classes
4:        $o_{i,s} \rightarrow \mathbf{p} = \sum_{k=1}^{|\mathbf{Y}|} \mathbf{1}_j(y_k) / |\mathbf{Y}|$                 ▷ store class probabilities
5:     end for
6:   else                                                                    ▷ non-terminal node
7:      $\mathbf{m} = \text{SAMPLE}(\{1, \dots, d\}, d_{trial})$                             ▷ random subset of the features
8:     for  $k \in \mathbf{m}$  do
9:        $\{v_k, \theta_k\} = \text{G}(\mathbf{X} \cdot \hat{\mathbf{e}}_k, \mathbf{Y})$                         ▷ find best split value and location
10:    end for
11:     $o_{i,s} \rightarrow j = \underset{k \in \mathbf{m}}{\text{argmin}}(v_k)$                             ▷ feature with minimum split
12:     $o_{i,s} \rightarrow \theta = \theta_{o_{i,s} \rightarrow j}$                         ▷ store the split location
13:     $L = (\bigcup_{k=1}^{|\mathbf{Y}|} \mathbf{1}_{\mathbf{X}_{o_{i,s} \rightarrow j, k} < o_{i,s} \rightarrow \theta} \cdot k) \setminus \{0\}$     ▷ examples on left side of split
14:     $R = \{1, \dots, k\} \setminus L$                                           ▷ examples on right side of split
15:    TRAIN( $\underset{k \in L}{\text{C}} \mathbf{X} \cdot \hat{\mathbf{e}}_k, \underset{k \in L}{\text{C}} y_k, o_{i,2 \cdot s}$ )                ▷ continue training on both subsets
16:    TRAIN( $\underset{k \in R}{\text{C}} \mathbf{X} \cdot \hat{\mathbf{e}}_k, \underset{k \in R}{\text{C}} y_k, o_{i,2 \cdot s+1}$ )
17:   end if
18: end function

```

---

Algorithm 3.3: Training of individual decision trees.  $\mathbf{X}$  is the complete set of training vectors and  $\mathbf{Y}$  the corresponding labels,  $r_{node}$  the minimal number of points required for a node split,  $\mathbf{1}_j$  the indicator function for the value  $j$ ,  $d_{trial}$  the number of random feature dimensions used in each node to choose a decision threshold,  $\text{G}$  the split function,  $v_j$  the value of the split function for feature  $j$  and  $\text{C}_{k \in L} a_k$  is used as a vertical concatenation operator for all  $a_k$  with  $k \in L$

---

$$G_{gini}(\mathbf{p}) = \sum_{i=1}^c p_i \cdot (1 - p_i) = 1 - \sum_{i=1}^c p_i^2 \quad (3.1)$$

The information gain criterion is related to the entropy contained in the probability distribution of the subset labels (eq. 3.2)

$$G_{inf}(\mathbf{p}) = - \sum_{i=1}^c p_i \cdot \log(p_i) \quad (3.2)$$

Here we present another measure of dissimilarity, which is not specific to a split point, but depends on the total ordering of the labels  $\mathbf{Y}$  sorted according to their rank in the input dimension, and how many class transitions are present along the dimension (eq. 3.3)

$$G_{flip}(\mathbf{Y}) = - \sum_{i=2}^{|\mathbf{Y}|} \frac{1 - \mathbf{1}_{y_i}(y_{i-1})}{|\mathbf{Y}|} \quad (3.3)$$

It can be shown that this measure is related to the local probability distribution in each point. In a local neighborhood according to the input axis the probability of a class flip between two neighboring points is (eq. 3.4)

$$p_{flip}(z_0) = \sum_{i=1}^c p_i(z_0) \cdot (1 - p_i(z_0)) = 1 - \sum_{i=1}^c p_i^2(z_0), \quad (3.4)$$

where  $p_i(z_0) = f_i(z_0) / (\sum_{j=1}^c f_j(z_0))$ , with  $f_i(z_0)$  the probability density of class  $i$  at  $z_0$ .

From this it becomes apparent that  $G_{flip}$  is an approximation of (eq. 3.5)

$$G_{flip}(\mathbf{Y}) = 1 - \sum_{i=1}^c \int_{-\infty}^{\infty} p_i^2(z) \cdot \sum_{j=1}^c f_j(z) dz = 1 - \sum_{i=1}^c \int_{-\infty}^{\infty} \frac{f_i^2(z)}{\sum_{j=1}^c f_j(z)} dz \quad (3.5)$$

Commonly for the gini impurity and information gain criterion, all possible splits along a dimension are considered and the dissimilarities on both sides are weighted by the number of points on each side of the split. Alternatively to considering all possible splits along the dimension, extremely randomized forests [52] only consider one splitting location for each dimension and compares these and can be used to provide a split location for eq. 3.3.

### 3.4 Alternative coordinate systems

Only being able to have axis aligned decision boundaries can make classification tasks where the data is aligned with a rotated coordinate system more difficult. Principle component analysis(PCA) is a common dimensionality reduction technique, which creates a coordinate

system with dimensions which are ranked by the amount of variance in the data which they can explain.

Here we present new method which aims to minimize the ratio of the variances of individual classes  $j$  against the total variance of the data, which will be called covariance ratio projection(CVRP) (covariance ratio projection). More formally, we look for the dimension which maximizes the ratio in eq. 3.6

$$\mathbf{u}_j = \operatorname{argmax}_{\mathbf{u}_j} \left( \frac{\operatorname{var}(\mathbf{u}_j \cdot \mathbf{X})}{\operatorname{var}(\mathbf{u}_j \cdot \mathbf{X}|_j)} \right), \quad (3.6)$$

where  $\mathbf{X}$  is the input data distribution and  $\mathbf{X}|_j$  the input data distribution given the data having class label  $j$ .

We can reformulate the description of  $\mathbf{u}_j$  (eq.3.7)

$$\begin{aligned} \mathbf{u}_j &= \operatorname{argmax}_{\mathbf{u}_j} \left( \frac{\mathbb{E}(\mathbf{u}_j^T \mathbf{X} - \mathbb{E} \mathbf{u}_j^T \mathbf{X})^T (\mathbf{u}_j^T \mathbf{X} - \mathbb{E} \mathbf{u}_j^T \mathbf{X})}{\mathbb{E}(\mathbf{u}_j^T \mathbf{X}|_j - \mathbb{E} \mathbf{u}_j^T \mathbf{X}|_j)^T (\mathbf{u}_j^T \mathbf{X}|_j - \mathbb{E} \mathbf{u}_j^T \mathbf{X}|_j)} \right) \\ &= \operatorname{argmax}_{\mathbf{u}_j} \left( \frac{\mathbf{u}_j^T \Sigma(\mathbf{X}) \mathbf{u}_j}{\mathbf{u}_j^T \Sigma(\mathbf{X}|_j) \mathbf{u}_j} \right), \end{aligned} \quad (3.7)$$

where  $\Sigma(\mathbf{X})$  is the covariance matrix of  $\mathbf{X}$ . This can be recognized as the eigenvector corresponding to the largest eigenvalue  $\lambda$  of a generalized eigenvalue problem (eq. 3.8)

$$\Sigma(\mathbf{X}|_j)^{-1} \Sigma(\mathbf{X}) \cdot \mathbf{u}_j = \lambda \cdot \mathbf{u}_j \quad (3.8)$$

## 3.5 Ensemble weighting

As previously mentioned, it is possible to assign weights to the individual classifiers in an ensemble. If the individual votes for a data point for each class are considered as probabilities, we can try and find a set of weights which minimizes the probability estimation error. This is presented more formally in eq. 3.9

$$\mathbf{w} = \operatorname{argmin}_{\mathbf{w}} \left( \sum_{k=1}^c \sum_{i=1}^{|\mathbf{Y}|} \left( \sum_{j=1}^t w_j \cdot v_{ijk} - \mathbf{1}_k(y_i) \right)^2 \right), \quad (3.9)$$

where  $v_{ijk}$  is the prediction of the data point  $\mathbf{x}_i$  of the  $j^{\text{th}}$  classifier for class  $k$ .

This can be rewritten in matrix form (eq. 3.10)

$$\begin{aligned} \mathbf{w} &= \operatorname{argmin}_{\mathbf{w}} \left( \sum_{k=1}^c (\mathbf{V}_k \mathbf{w} - \mathbf{1}_k(\mathbf{Y}))^T (\mathbf{V}_k \mathbf{w} - \mathbf{1}_k(\mathbf{Y})) \right) \\ &= \operatorname{argmin}_{\mathbf{w}} \left( \mathbf{w}^T \left( \sum_{k=1}^c \mathbf{V}_k^T \mathbf{V}_k \right) \mathbf{w} - 2 \cdot \left( \sum_{k=1}^c \mathbf{1}_k(\mathbf{Y})^T \mathbf{V}_k \right) \mathbf{w} + \mathbf{1}_k(\mathbf{w})^T \mathbf{1}_k(\mathbf{Y}) \right), \end{aligned} \quad (3.10)$$

where  $\mathbf{V}_k$  contains the predictions of class  $k$  of each tree for each point. To find the optimal  $\mathbf{w}$  we need to solve the positive definite quadratic programming problem presented above with the constraints  $|\mathbf{w}|_1 = 1, 0 \leq w_j \leq 1, \forall j \in \{1, \dots, t\}$ .

The classifiers used here can be complete random forests or individual classification trees (or even totally different types of classification functions). A correlation matrix of the classifiers is given by  $\mathbf{V}_k^T \mathbf{V}_k$ . Alternatively, we can also readjust the probabilities in the tree leaf nodes (eq. 3.11)

$$\mathbf{p}_{\text{leaves}} = \underset{\mathbf{P}}{\operatorname{argmin}} \left( \mathbf{p}^T \left( \sum_{k=1}^c \mathbf{v}_k^T \mathbf{v}_k \right) \mathbf{p} - 2 \cdot \left( \sum_{k=1}^c \mathbf{1}_k(\mathbf{Y})^T \mathbf{v}_k \right) \mathbf{p} + \mathbf{1}_k(\mathbf{Y})^T \mathbf{1}_k(\mathbf{Y}) \right), \quad (3.11)$$

where  $\mathbf{p}_{\text{leaves}}$  contains the class probability distribution in each leaf in each tree and  $\mathbf{v}_k^T \mathbf{v}_k$  are the correlations between all leaves for the  $k^{\text{th}}$  data point. Constraints are needed to guaranty that in each leaf  $|\mathbf{p}_{\text{leaf}}|_1 = 1$ . For weakly overlapping leaves between trees  $\sum_{k=1}^c \mathbf{v}_k^T \mathbf{v}_k$  is more likely to be sparse, keeping the quadratic programming solvable for many trees with many leaves.

An alternative weighting approach for classifiers would be to choose classifier weights so that the majority vote class is most often correct. A greedy optimization was developed for this (Alg. 3.4)

For both weighting approaches no improvement was observed in regards to the classification performance of the random forest algorithm. However, the first approach allowed for a ranking of all trees by their weight, and for various classification tasks a large fraction of the tree weights were focused on a relatively small subset of all decision trees.

## 3.6 Out of bag Error

## 3.7 Classification Experiments

The algorithms from this chapter were implemented as Java classes and called in Jython scripts using FIJI [148]. For a first comparison data sets were taken from the UCI machine learning database which had been previously used to evaluate the performance of random forests in their original presentation [16], as well as for some additional sets. Simulations were run for a different number of trial dimensions:  $d_{\text{trial}} = \{1, 2, \lfloor \log_2(d) + 1 \rfloor, \lfloor \sqrt{d} \rfloor\}$ , four different split setups  $G = \{G_{\text{gini}}, G_{\text{inf}}, G_{\text{flip}}, G_{\text{rs}}\}$ , where  $G_{\text{rs}}$  randomly selects one of the splitting rules  $\{G_{\text{gini}}, G_{\text{inf}}, G_{\text{flip}}\}$  for each node, and three coordinate systems: original input, PCA of the z-score normalized input and CVRP of the z-score normalized input, both concatenated with the original input dimensions. It is important to note that for the gini split the optimal split location was used, while for the information gain split a random split location was used. The flip split method has no optimal split location and also used a random split location. For each of the coordinate systems the forest with the lowest out of bag error was selected and retrained for classification and compared with the other forests

---

```

1: function GDTW(o, X, Y)
2:   while  $it < \text{maxit}$  do
3:      $it = it + 1$ 
4:     for  $i \in \{1, \dots, |\mathbf{w}|\}$  do ▷ go through trees in random order
5:       for  $j \in \{1, \dots, |\mathbf{Y}|\}$  do ▷ go through points
6:         for  $k \in \{1, \dots, c\} \setminus \{y_i\}$  do ▷ go through non-label classes
7:            $\lambda = (\tilde{v}_k - \tilde{v}_{y_i}) / (v_{ik} - \tilde{v}_k - v_{iy_i} + \tilde{v}_{y_i})$  ▷ intersection  $w_i$  with  $y_i$  the majority class
8:           if  $v_{ik} - \tilde{v}_{y_i} > v_{iy_i} + \tilde{v}_{y_i}$  then ▷ right sided interval
9:              $q_{jk} = [\Lambda, 1]$ 
10:          else ▷ left sided interval
11:             $q_{jk} = [0, \Lambda]$ 
12:          end if
13:        end for
14:         $q_j = \bigcap_{k=1}^c q_{jk}$  ▷ intersection interval of all intervals
15:      end for
16:       $w_i = \text{RAND}(\text{MAXOVERLAP}(\mathbf{q}))$  ▷ find interval with the most overlapping intervals  $q_j$ 
17:       $\text{REWEIGHT}(\mathbf{w}, i, w_i)$  ▷ reweight  $\mathbf{w}$  using new  $w_i$ 
18:    end for
19:  end while
20:  return  $\mathbf{w}$ 
21: end function

```

---

Algorithm 3.4: Greedy weight optimization.  $\tilde{v}_k$  is the  $k^{\text{th}}$  class vote of point  $j$  without tree  $i$ .  $v_{ik}$  is the  $k^{\text{th}}$  class vote of point  $j$  of tree  $i$ .

---

as well. For each coordinate system, it was checked which of the test sets were significantly outperformed by the forest with the lowest test error (highlighted in bold). Forests with \*,\*\* or \*\*\* had p-values higher than 0.95, 0.99 and 0.999 respectively. Results can be found in Appendix B.1 in the dataset tables (performance table for each dataset). Left figures show how well out of bag error predicts the actual test set error. Figures of the right side show how the random forest out of bag error decreases with the number of decision trees for the best performing classifier for each coordinate system.

The data sets were the following (results in Appendix B.1):

## **Banknote Authentication**

See Table B.1 for results and [102] for the dataset origin. The dataset contains 1372 examples with two class labels and four input features. The test set was comprised of 138 randomly selected examples. This dataset contains examples of genuine and counterfeit banknotes in the form of four wavelet transform attributes. The attributes were chosen to allow for a quick separation of genuine and counterfeit examples. Therefore classification accuracy is high (perfect classification accuracy for 100 permutations of training and test data). For all parameter settings the mapping of out of bag error and test error is close to unity. The dataset has a low input dimensionality and might therefore profit from the additional input dimensions provided by zPCA and zCVRP which have higher classification accuracies (in particular zCVRP). For this dataset the information gain and flip split criteria perform better than the gini split.

## **Forest Cover type**

See Table B.2 for results and [13] for the dataset origin. The dataset contains 581012 examples with seven class labels and 54 input features, 40 of which are binary valued. The test set was comprised of 565892 randomly selected examples. This dataset contains examples with cartographic features which are used to determine the type of forest cover type of 30m  $\times$  30m patches on satellite images. Typical classification accuracy is in the range of 80% - 85% . The dimensionality of the dataset is comparably low against the number of training examples and for all parameter settings the mapping between out of bag error and test error is close to unity. The performance is noticeably higher for zPCA and in particular zCVRP. From the out of bag error vs number of trees plot we can observe that zPCA and zCVRP are able to make better use of additional trees, all the way up to the 100 trees used for classification. The gini and the flip method have the best performances for this data set.

## **German Credit**

See Table B.3 for results and [73] for the dataset origin. The dataset contains 1000 examples with two class labels and 24 input features (which were expanded from 20 with categorical dimensions). The test set was comprised of 100 randomly selected examples. This dataset contains personal attributes of credit takers together with their labeling as good or as bad



clients. Typical classification accuracy is in the range of 70% - 77% . The mapping between out of bag error and test error deviates from unity, and more so for zCVRP. Performance is highest for zPCA, followed by standard dimensions and finally zCVRP. In all cases, gini is clearly the best splitting method.

## **Glass Identification**

See Table B.4 for results and [42] for the dataset origin. The dataset contains 214 examples with seven class labels and 9 input features. The test set was comprised of 22 randomly selected examples. This dataset contains refractory values for different glass types intended for the use in forensic scene analysis. Typical classification accuracy is in the range of 70% - 79% . For this dataset the initial dimensions yield the best results. The mapping of out of bag error and test error is close to unity except for zCVRP. Typically, the best split is provided by the gini split method, but the overall best accuracy is given by mixed split approach.

## **Ionosphere**

See Table B.5 for results and [156] for the dataset origin. The dataset contains 351 examples with two class labels and 34 input features. The test set was comprised of 36 randomly selected examples. This dataset contains sonar measurements of the ionosphere tracking free electrons. Typical classification accuracy is in the range of 90% - 95% . For this dataset zPCA yields the best results, closely followed by zCVRP. The mapping of out of bag error and test error is close to unity except for CVRP. CVRP seems to even have out of bag errors which are anti-correlated with the test error. This could be due to the fact that the number of input dimensions of CVRP is comparable to the number of samples given for training. Typically, the best split is provided by the gini split method or the mixed split approach in the case of CVRP.

## **Iris**

See Table B.6 for results and [44] for the dataset origin. The dataset contains 150 examples with three class labels and 4 input features. The test set was comprised of 15 randomly selected examples. This dataset contains stem and petal properties of three different types of iris flowers. Typical classification accuracy is high and in the range of 95% - 98% . For this dataset zCVRP yields the best results by a margin. This could be due the low dimensionality of the input which might be aided The mapping of out of bag error and test error is close to unity. The split method makes little difference on accuracy, gini split has a tendency to be a bit better.

## Liver

See Table B.7 for results and [46] for the dataset origin. The dataset contains 345 examples with two class labels and 6 input features. The test set was comprised of 35 randomly selected examples. This dataset labels contains blood values used to determine if a patient has a liver disorder. Typical classification accuracy is in the fairly tight range of 72% - 74% . For this dataset the standard input provides the highest accuracy, and that while using only one dimension for random dimension selection. The mapping of out of bag error to test error is jittered. The gini split method has a slight edge on the other split methods.

## Sonar

See Table B.8 for results and [55] for the dataset origin. The dataset contains 208 examples with two class labels and 60 input features. The test set was comprised of 21 randomly selected examples. This dataset contains sonar data to determine the material of a cylinder. Typical classification accuracy is in the range of 76% - 85% . For this dataset zPCA provides the highest accuracy followed by the standard input. The mapping of out of bag error and test error is close to unity except for CVRP. CVRP seems to even have out of bag errors which are anti-correlated with the test error. Typically, the best split is provided by the gini split method.

## Twonorm

See Table B.9 for results and [138] for the dataset origin. The dataset contains 7400 examples with two class labels and 20 input features. The test set was comprised of 7100 randomly selected examples. This dataset is an artificial dataset which separates points into classes using their distance according to a  $L_2$ -norm. Typical classification accuracy is in the range of 95% - 98% . For this dataset zPCA provides the highest accuracy trailed by zCVRP. The mapping of out of bag error and test error is close to unity only for the standard input. The out of bag error tends to predict a too low test error for both zPCA and zCVRP. Best split method seems to depend on the actual feature dimensions chosen.

## Vowel

See Table B.10 for results and [35] for the dataset origin. The dataset contains 990 examples with eleven class labels and 10 input features. The test set was comprised of 99 randomly selected examples. This dataset contains audio attributes of different speakers which should be assigned to particular vowels. Typical classification accuracy is high and in the range of 97% - 99% . For this dataset zCVRP provides the highest accuracy trailed by zPCA. The out of bag error tends to overpredict the test error for zPCA and zCVRP. The flip split provides the highest accuracies in this case.

## Waveform

See Table B.11 for results and [17] for the dataset origin. The dataset contains 5000 examples with three class labels and 21 input features. The test set was comprised of 4700 randomly selected examples. This dataset contains artificial data for a waveform based separation. Typical classification accuracy is in the range of 78% - 85% . For this dataset zPCA and zCVRP are tied for the highest accuracy. The out of bag error tends to overpredict the test error for zPCA and zCVRP. The mixed split provides the highest accuracies improving upon the gini split.

## Wine

See Table B.12 for results and [45] for the dataset origin. The dataset contains 178 examples with three class labels and 13 input features. The test set was comprised of 18 randomly selected examples. This dataset contains chemical attributes to determine wine origin. Typical classification accuracy is high and in the range of 98% - 99% . For this dataset zCVRP yields the best results by a margin. The mapping between out of bag error and test error are jittered for all inputs. The gini split provides the best accuracy in all cases.

## Discussion

The effect of input coordinates and splitting methods on the random forest algorithm were presented. The splitting methods consisted of a standardized gini impurity based splitting, an entropy based splitting using a random split location and a new splitting method based on local class transitions, as a well as a hybrid split approach. There is no general preference in choosing the best splitting location or choosing a randomized split location. There are differences but not in favor of one particular approach. Among the randomized splits, the information gain and flip approaches mostly performed similarly with no obvious preference. Generally, if it is unknown whether the out of bag error will be a good predictor for the test error it is best to use the hybrid split approach, as this has the best average case accuracies of all approaches. The usefulness of adding dimensions from zCVRP as well as zPCA depends greatly on the dataset being used. Here, datasets which are generally well separable with high classification accuracy profited more from the dimensions provided by zCVRP. For some datasets however, the zCVRP dimensions seem to allow the random forests to overfit on the training data, which is reflected in the relationships between out of bag error and test error.

## 3.8 Random forests for regression

With minimal adaption, decision trees as presented previously can also be used for regression tasks (prediction of a continuous value, instead of a class label) [16]. The leaf nodes of the decision trees contain estimates of the mean of the target regression value. To train decision trees for regression a different splitting criterion is required. The most commonly used split

criterion, variance reduction, minimizes the weighted variance contained on both sides of the split (eq. 3.12)

$$G_{varr}(\mathbf{Y}) = \text{var}(\mathbf{Y}), \quad (3.12)$$

where  $\mathbf{Y}$  contains the points' regression values on one side of the split boundary.

Similar to the flip split in classification 3.3, here a method which approximates the local variance around the data points according to the ordering of the current dimension is presented (eq. 3.13)

$$G_{varf}(\mathbf{Y}) = \sum_{i=2}^{|\mathbf{Y}|} (y_i - y_{i-1})^2 / |\mathbf{Y}|, \quad (3.13)$$

It can be shown that this measure is related to the local variance distribution in each point. In a local neighborhood of  $z_0$  along the input axis the expected squared difference between two neighboring points is (eq. 3.14)

$$\begin{aligned} v_{varf}|_{z_0} &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (y_0 - y_1)^2 \cdot f(y_1)|_{z_0} \cdot f(y_0)|_{z_0} dy_0 dy_1 = \\ &= \int_{-\infty}^{\infty} (\mathbf{E} \mathbf{y}^2|_{z_0} - 2 \mathbf{E} y|_{z_0} \cdot y_1 + y_1^2) \cdot f(y_1)|_{z_0} dy_1 \\ &= 2(\mathbf{E} \mathbf{y}^2|_{z_0} - (\mathbf{E} \mathbf{y}|_{z_0})^2) = 2 \cdot \text{var}(\mathbf{Y})|_{z_0}, \end{aligned} \quad (3.14)$$

where  $f(y)|_{z_0}$  is the local density function of the regression target distribution of  $\mathbf{y}$  at the location  $z_0$ . From this it becomes apparent that  $G_{varf}$  is an approximation of (eq. 3.15)

$$G_{varf}(\mathbf{Y}) = \int_{-\infty}^{\infty} \text{var}(\mathbf{Y})|_z \cdot f(\mathbf{Y})|_z dz \quad (3.15)$$

which is the density weighted variance of the current dimension.

For regression ensembles optimal weights can be found with eq. 3.16 (compare eq. 3.9)

$$\mathbf{w} = \underset{\mathbf{w}}{\text{argmin}} (\mathbf{w}^T \mathbf{V}^T \mathbf{V} \mathbf{w} - 2 \cdot \mathbf{Y}^T \mathbf{V} \mathbf{w} + \mathbf{Y}^T \mathbf{Y}), \quad (3.16)$$

where  $\mathbf{V}$  contains the regression prediction of each tree for each point.

### 3.9 Random forest proximities and clustering

Random forests can provide a measure of how similar two points are in regards to a target classification or regression for all trees. This proximity or similarity measure is simply the

number of leaves which contain both these points among all trees in the forest. A novel clustering method that takes advantage of this proximity measure is introduced in Alg. 3.5. This clustering is reminiscent of a k-means clustering, where there are  $|\mathbf{o}|$  input dimensions, however here each dimension is categorical (built up from a non-ordered set of leaves), therefore the cluster center calculation will differ from that of k-means. Another difference is the variable balancing factor for the number of points already contained in each cluster.

---

```

1: function RFC( $\mathbf{o}, \mathbf{X}, n_{clust}, b$ )
2:   for  $i \in \{1, \dots, |\mathbf{X}|\}$  do                                ▷ assign all points to random initial clusters
3:      $cl_i = \text{RANDINT}(n_{clust})$ 
4:     for  $j \in \{1, \dots, |\mathbf{o}|\}$  do                                ▷ update clusters for each tree
5:        $M_{cl_i, j, l_{ij}} = M_{cl_i, j, l_{ij}} + 1$                 ▷ update cluster center for visited leaf
6:     end for
7:   end for
8:   while  $it < maxit$  do
9:      $it = it + 1$ 
10:    for  $i \in \{1, \dots, |\mathbf{X}|\}$  do                                ▷ traverse points in random order
11:      for  $j \in \{1, \dots, |\mathbf{o}|\}$  do                                ▷ traverse trees in random order
12:         $M_{cl_i, j, l_{ij}} = M_{cl_i, j, l_{ij}} - 1$                 ▷ remove  $i$  from current cluster
13:         $cl_i = \emptyset$ 
14:      end for
15:       $cl_i = \underset{k}{\operatorname{argmax}} \sum_{j=1}^{|\mathbf{o}|} (M_{k, j, l_{ij}}) / (\sum_{s=1}^{|\mathbf{X}|} \mathbf{1}_k(cl_s))^b$ 
                                                                    ▷ Assign to cluster with maximal leaf overlap
16:      for  $j \in \{1, \dots, |\mathbf{o}|\}$  do                                ▷ update centers
17:         $M_{cl_i, j, l_{ij}} = M_{cl_i, j, l_{ij}} + 1$ 
18:      end for
19:    end for
20:  end while
21:  return  $\mathbf{M}$ 
22: end function

```

---

Algorithm 3.5: Random forest clustering. Returns  $n_{clust}$  cluster centers in  $\mathbf{M}$  which contain the number of samples in each cluster for each leaf in each decision tree. Cluster sizes can be balanced with the parameter  $b$ . The value  $l_{ij}$  is the leaf index of point  $i$  in tree  $j$ .

---

This clustering requires that a random forest for either classification or regression has been trained a priori. Unsupervised clustering can be achieved by classifying input data against artificially randomized versions of the same data. Points which are close to another due to highly correlated dimensions will be close to one another by forest proximity in this case as well.

An alternative approach for unsupervised decision trees has splitting nodes which partition the points into groups of minimal sum of squared inter-point distances. The sum of squared inter-point distances of a set of points is given by (eq. 3.17)

$$\text{sipd} = \sum_{i=1}^{|\mathbf{X}|} \sum_{j=1}^{|\mathbf{X}|} \sum_{k=1}^d (X_{ik} - X_{jk})^2 = 2|\mathbf{X}|^2 \sum_{k=1}^d \text{var}(\mathbf{X}_{:,k}), \quad (3.17)$$

where  $:$  denotes all possible indexes, to obtain the variance of the regression value  $\mathbf{X}$  with regards to a single dimension  $k$ . If there were an infinite number of data points and all dimensions would be independent, the optimal axis aligned split to minimize the sipd on both sides of the split would be given by the dimension with the maximal difference between sipd on both sides of the split and the previous sipd value. We can repeatedly apply this procedure to the data with correlated dimensions, to find splits corresponding to less correlated dimensions of the data.

### 3.10 Regression Experiments

For the investigation of random forest regression we used a similar setup as for the classification task. While classification errors reflect the amount of misclassified test case examples, regression errors reflect by how much a value prediction for the value of a test example deviates from the actual test value. For a first comparison, data sets were taken from the UCI machine learning database, which had been previously used to evaluate random forest performance for regression [16]. Simulations were run for a different number of trial dimensions:  $d_{\text{trial}} = \{1, 2, \lfloor \log_2(d) + 1 \rfloor, \lfloor \sqrt{d} \rfloor\}$ , four different split setups  $G = \{G_{\text{varr}}, G_{\text{varf}}, G_{\text{rs}}\}$ , where  $G_{\text{rs}}$  randomly selects one of the splitting rules  $\{G_{\text{varr}}, G_{\text{varf}}\}$  for each node, and three coordinate systems: original input, PCA of the z-score normalized input and CVRP of the z-score normalized input, both concatenated with the original input dimensions. Labels for CVRP for the regression task were obtained by creating a preliminary regression forest and creating clusters based on the forest proximities. For each of the coordinate systems the forest with the lowest out of bag error was selected and retrained for regression and compared with the other forests as well. For each coordinate system, it was checked which of the test sets were significantly outperformed by the forest with the lowest test error (highlighted in bold). Forests with \*,\*\* or \*\*\* had p-values higher than 0.95, 0.99 and 0.999 respectively. Result tables can be found in Appendix B.2. Left figures show how well out of bag error predicts the actual test set error. Right figures show how the out of bag error decreases with the number of decision trees for the best performing classifier for each coordinate system.

#### Abalone

See Table B.13 for results and [181] for the dataset origin. The dataset contains 4177 examples with eight input features. The test set was comprised of 1045 randomly selected examples. This data set contains attributes to predict the age of Tasmanian Abalone shellfish. The decrease in regression error between a single tree and a full forest is around a factor of two. Here both zPCA and zCVRP allow for improvements in accuracy, where zCVRP

slightly outperforms zPCA. Both methods have their highest accuracy when using the hybrid split.

## **Airfoil**

See Table B.14 for results and [20] for the dataset origin. The dataset contains 1503 examples with five input features. The test set was comprised of 151 randomly selected examples. This data set contains wind tunnel speeds and angles of attack on an airfoil to predict noise levels. The decrease in regression error between a single tree and a full forest is around a factor of 2.5. zPCA provides the best performance when using the standard variance split.

## **Concrete**

See Table B.15 for results and [187] for the dataset origin. The dataset contains 1030 examples with eight input features. The test set was comprised of 103 randomly selected examples. This data set contains concrete compressive strengths which should be predicted based on production attributes. The decrease in regression error between a single tree and a full forest is around a factor of 2.5. Here standard regression forests with variance split yield the best performance.

## **Housing**

See Table B.16 for results and [61] for the dataset origin. The dataset contains 506 examples with 13 input features. The test set was comprised of 51 randomly selected examples. This data set was used to determine the willingness of residents to pay for clean air based in various economic factors. The decrease in regression error between a single tree and a full forest is around a factor of three. Here standard regression forests with variance split yield the best performance.

## **Servo**

See Table B.17 for results and [134] for the dataset origin. The dataset contains 167 examples with four input features. The test set was comprised of 17 randomly selected examples. This data contains information on the response time of a servo system based on the circuit configuration. The decrease in regression error between a single tree and a full forest is around a factor of two. Here standard regression forests with variance split yield the best performance.

## **Discussion**

The results provided for the regression data sets provided here are different from the classification results in a few regards. Typical decreases in regression error for regression forests versus a single tree were in the range of a factor of two to three, which is a substantially

smaller gain than for many of the classification data sets. For regression tasks using the initial dimensions often provided the best predictions, though it is not clear if this is due to over-fitting of the initial data while providing additional dimensions. Also, for these data sets using a variance reduction split also commonly gave the best results. It would be necessary to investigate a broader array of data sets to see or data sets beyond the Abalone data set can be assigned lower regression errors when adding further dimension for regression as well as using different splitting methods..

### 3.11 Remarks

The data analysis in this chapter used a cross validation approach to assess the performance of various components of the random forest algorithm. This type of cross validation guarantees that the training and test sets have the same data statistics. For various real world applications the training and testing set may differ significantly. There might be significant correlations of the input dimensions present only in the training data which might bias classifiers during training.

A classification algorithm could be used to determine how similar training and test data are (e.g. the training set and test set are more similar when the classification accuracy to distinguish examples from both is low), and might provide insights on how to better choose training samples when working with biased training set. Trying to have only training and test set invariant dimensions is useful to keep generalized performance.



# Variables

- 1** Indicator function with  $\mathbf{1}_x(x) = 1$  and otherwise 0. 31–33, 40
- argmax** location of the maximum value. 32, 33, 40
- argmin** location of the minimum value. 31, 33, 39
- b*** elasticity parameter for random forest clustering. 40
- C** concatenation operator. 31
- c*** number of possible labels for  $y$ . 29–34
- cl*** cluster number for each point in  $\mathbf{X}$ . 40
- d*** random forest input dimensionality. 29, 31, 35, 41
- d<sub>trial</sub>*** number of randomly chosen dimension for splitting during the random forest training. 30, 31, 35, 41
- E** expected value. 33, 39
- $\hat{\mathbf{e}}$**  unit vector  $\hat{\mathbf{e}}_k$  for dimension  $k$ . 31
- G** splitting function for tree nodes during the random forest training. 30, 31, 35, 41
- $G_{flip}$**  local gini splitting criterion. 32, 35
- $G_{gini}$**  gini index based splitting criterion. 30, 35
- $G_{inf}$**  information gain based splitting criterion. 30, 35
- $G_{rs}$**  randomly selected splitting rule for each node. 35, 41
- $G_{varf}$**  local variance reduction splitting rule for regression trees. 39, 41
- $G_{varr}$**  variance reduction splitting rule for regression trees. 39, 41

$L$  union of all points present at the current split with are smaller than the threshold. 31

$M$  cluster centers over all decision trees and leaf nodes. 40

$\mathbf{m}$  randomly sampled splitting dimensions. 31

max maximum function. 31

**maxoverlap** find the location where the maximum interval segments overlap. 34

$n_{clust}$  number of clusters for random forest clustering. 40

$o$  compound  $o_{i,s} = o_{i,s}\{\theta, j, \mathbf{p}\}$  of the  $s^{th}$  node and  $i^{th}$  tree.  $\mathbf{o}$  is the collection of all trees. 29–31, 34, 40

$\mathbf{p}$  class probability vector of a leaf node. 29–31, 33

$R$  union of all points present at the current split with are greater or equal than the threshold. 31

$r_{node}$  minimum number of points in a node required for further splitting. 31

**randint** draw a random integer between 1 and  $n_{clust}$ . 40

**sample** sample  $d_{trial}$  dimensions from  $d$  with replacement. 31

sipd summed squared inter-point distance function. 41

$t$  number of trees in the random forest. 29, 30, 33

var variance. 32, 39, 41

$\mathbf{w}$  vector of ensemble voting weights for each tree. 30, 33, 34, 39

$\mathbf{X}$  collection of random forest input vectors. 31–34, 40, 41

$\mathbf{x}$  random forest input vector. 29, 30

$\mathbf{Y}$  collection of random forest input labels. 30–34, 39

$y$  random forest input labels. 29

$\theta$  splitting threshold  $\theta = \theta_j$  upon the  $j^{th}$  input dimension. 29, 31

$\lambda$  eigenvalue of matrix. 33

$\Sigma$  covariance matrix. 33

# Glossary

**CVRP** covariance ratio projection dimensions. 32, 35–37, 41

**PCA** principle component analysis. 32, 35, 41

**zCVRP** z-score normalized covariance ratio projection dimensions. 35–38, 42

**zPCA** z-score normalized principle component analysis. 35–38, 42

# Chapter 4

## Image Processing

### Summary

The classification of images or individual image pixels is a commonly encountered classification task. The choice or generation of the input features has a significant impact on classification tasks, and this is true for image classification as well. This chapter focuses on image processing steps which should be of use to general pixel-wise image classifiers. First, a simple set of filters to spectrally decompose the pixel neighborhood of a single pixel is presented. After follows the presentation of a watershed based method to create scale dependent super pixels. Finally, scalar measures to describe the geometry of connected pixel components are presented.

### 4.1 Spatial frequency filters

Our first focus goes towards frequency filters, which are commonly used as tools to decompose and interpret the image space (pixels and their x-y coordinates in the image) and shared properties of pixels with fixed spatial relations. The Fourier transform is a common starting point to understand the relationship of image (spatial) and spectral space (frequency domain).

The Fourier transform provides a frequency decomposition of a function using sine and cosine functions. A function can be represented as a sum of sine and cosine functions of different frequency, if it is a bounded function  $f(x)$  which fulfills eq. 4.1:

$$\int_{-\infty}^{\infty} \|f(x)\| dx < \infty \quad (4.1)$$

An attribute of the Fourier transform is that a translation in the image domain of the form  $f(x-x_0)$  leads to a phase shift of the corresponding frequency signal  $\hat{f}(\omega) \cdot e^{-i2\pi x_0 \omega}$ . Inversely, multiplying a masking function with the transformed signal in the frequency domain will highlight all function components corresponding to masked frequencies, irrespective of their

offset in the spatial domain. This is commonly the motivation to convolve image templates with an image, to highlight the prevalence of a masking template within the image. Such observations hint at the usefulness of investigating signals in the spatial as well as in frequency domain. Signals can not be well simultaneously localized in the spatial and in the frequency domain, a principle which is known as the uncertainty principle of the Fourier transform. This uncertainty is minimal for Gaussian functions and helps explain the usefulness of Gabor and related filters [48]. Natural images often contain an amount of self-similarity or scale invariance [43], [169]. This motivates the choice of a filter with the same shape irrespective of absolute frequency (scale invariance), which has a balanced multiplicative frequency coverage around the filter's center frequency for higher and lower frequencies and provides a function basis which can potentially cover all frequencies in the frequency domain. These properties are found in the circularly symmetric Log-Gabor filter defined by eq. 4.2 derived from [43].

$$\text{LG}(\boldsymbol{\omega}, k, \sigma) = \exp\left(-\pi^2 \frac{\log(k\|\boldsymbol{\omega}\|)^2}{2(\log(2)\sigma)^2}\right), \quad (4.2)$$

where  $\boldsymbol{\omega}$  is a multidimensional frequency vector,  $k$  roughly corresponds to the first zero crossing of the filter in the spatial domain and  $\sigma$  is the filter's bandwidth coverage in octaves. The Log-Gabor filter can be used to generate a more general bandpass filter. A very simple bandpass filter is given by focusing on a frequency interval of interest and setting all other frequency values to zero. A smooth approximation to such a bandpass filter is given by the Weierstrass transform of a rectangular bandpass filter (eq. 4.3)

$$\begin{aligned} \text{GPB}(\boldsymbol{\omega}, a, b, t) &= \frac{1}{\sqrt{4\pi}} \int_a^b 1 \cdot \exp\left(-\frac{(\|\boldsymbol{\omega}\| - y)^2}{4t}\right) dy \\ &= \frac{1}{2} \left( \text{erf}\left(\frac{\|\boldsymbol{\omega}\| - b}{\sqrt{4t}}\right) - \text{erf}\left(\frac{\|\boldsymbol{\omega}\| - a}{\sqrt{4t}}\right) \right), \end{aligned} \quad (4.3)$$

where  $t$  determines the smoothing of the bandpass, and  $a$  and  $b$  are the boundaries of the bandpass interval before the Weierstrass transform. Similarly, the bandpass can be approximated by a sum of exponentially spaced Log-Gabor filters (eq. 4.4)

$$\begin{aligned} \text{LGB}(\boldsymbol{\omega}, a, b, \sigma) &= \sqrt{2\pi} \int_a^b \exp\left(-\pi^2 \frac{\log(k\|\boldsymbol{\omega}\|)^2}{2(\log(2)\sigma)^2}\right) \frac{1}{k} dk \\ &= \sigma \log(2) \left( \text{erf}\left(\frac{\pi \log(\|\boldsymbol{\omega}\|b)}{\sqrt{2}\sigma \log(2)}\right) - \text{erf}\left(\frac{\pi \log(\|\boldsymbol{\omega}\|a)}{\sqrt{2}\sigma \log(2)}\right) \right), \end{aligned} \quad (4.4)$$

with  $a > 0, b > a$ .

All presented filters are dependent on the magnitude of the frequency vector, which has the same dimensionality as the image space. It is further possible to have the filter be

specialized for a certain direction and angle. For this the von Mises-Fisher distribution [36] is proposed as an angular decay function (eq. 4.5)

$$f_{\delta}(\boldsymbol{\omega}, \boldsymbol{\mu}, \kappa) = \frac{\kappa^{\delta/2-1}}{(2\pi)^{\delta/2} I_{\delta/2-1}(\kappa)} \exp\left(\kappa \frac{\boldsymbol{\mu}^T \boldsymbol{\omega}}{\|\boldsymbol{\mu}^T\| \|\boldsymbol{\omega}\|}\right), \quad (4.5)$$

where  $\delta$  is the dimensionality of the frequency vector,  $I_{\delta}$  the Bessel function of the first kind of order  $\delta$ ,  $\boldsymbol{\mu}$  is the target direction vector and  $\kappa$  is the angular dispersion. A oriented Log Gabor function would be the multiplication (eq. 4.6)

$$\text{OLG}(\boldsymbol{\omega}, \boldsymbol{\mu}, k, \sigma, \kappa) = \text{LG}(\boldsymbol{\omega}, k, \sigma) \cdot f_{\delta}(\boldsymbol{\omega}, \boldsymbol{\mu}, \kappa) \quad (4.6)$$

To guaranty that the oriented Log Gabor filter is real valued in the spatial domain it can be expanded with a phase component  $\phi$  (eq. 4.7)

$$\begin{aligned} \text{POLG}(\boldsymbol{\omega}, \boldsymbol{\mu}, k, \sigma, \kappa, \phi) = & \text{LG}(\boldsymbol{\omega}, k, \sigma) \cdot (\cos(\phi)(f_{\delta}(\boldsymbol{\omega}, \boldsymbol{\mu}, \kappa) + f_{\delta}(-\boldsymbol{\omega}, \boldsymbol{\mu}, \kappa)) \\ & + \mathbf{i} \cdot \sin(\phi)(f_{\delta}(\boldsymbol{\omega}, \boldsymbol{\mu}, \kappa) - f_{\delta}(-\boldsymbol{\omega}, \boldsymbol{\mu}, \kappa))) \end{aligned} \quad (4.7)$$

In chapter 3 it was observed that the input coordinate system may play an important role in classification performance. Simply rotating the coordinate system of the input space such as happens with principle component analysis(PCA) can have a significant impact on prediction. It was also remarked that training and testing sets should be drawn from the same probability distributions, if possible. Finally, for pixel classification there needs to be a balance of the number of input dimensions and the amount of information these contain, also for the sake of computational efficiency. Using Gabor filter banks, which are a collection of Gabor filters spanning various spatial scales and orientations, is a common method of providing a spatial neighborhood decomposition for each pixel. We propose a similar decomposition with phase oriented Log Gabor filters at different scales. Additionally, it is proposed to use a scale adapted local root mean square(rms) normalization of the filtered images in the real domain to make images more contrast invariant (eq. 4.8)

$$\text{SCN}(\hat{\mathbf{x}}, k, \sigma, \alpha) = \frac{\hat{\mathbf{x}} - \mathcal{N}(0, (\alpha k)^2) * \hat{\mathbf{x}}}{\sqrt{\mathcal{N}(0, (\alpha k)^2) * \hat{\mathbf{x}}^2 - (\mathcal{N}(0, (\alpha k)^2) * \hat{\mathbf{x}})^2}}, \quad (4.8)$$

where  $\mathcal{N}(\mu, \sigma^2)$  is the normal distribution with mean  $\mu$  and variance  $\sigma^2$ ,  $\alpha$  is a multiple of the zero crossing constant  $k$  of the Log Gabor filter,  $\hat{\mathbf{x}} = \text{LG}(\mathbf{x}, k, \alpha)(\mathbf{x})$  with  $\text{LG}(\mathbf{x}, k, \sigma)$  the convolution of the image space  $\text{im}$  and the Log Gabor filter (eq. 4.9)

$$\text{LG}(\mathbf{x}, k, \sigma) = (\mathcal{F}^{-1}\{\text{LG}(\boldsymbol{\omega}, k, \sigma)\} * \text{im})(\mathbf{x}) \quad (4.9)$$

Examples of images with Log Gabor filters and rms normalization applied to them are presented in fig. 4.1 - 4.4

## 4.2 Watershed to super pixels

In image processing the watershed algorithm provides a means to identify local minima and their corresponding basins of attraction. The watershed lines which separate the local minima are reminiscent of ridge lines which also describe local crossing properties. Taking a watershed of the negative values of an image space provides local maxima and watershed lines' corresponding basins of attraction. Log Gabor filtered images as well as their rms normalized versions have a mean intensity of zero due to the zero DC component of the log Gabor filter in the frequency domain. Thresholding the image at zero partitions the images into regions corresponding to high intensity and low intensity watershed ridge lines. Here, super pixels are created by taking the binary image after thresholding the filtered image at zero and storing the portions of the watershed regions of the positive image in pixels with intensities below the threshold and storing the portions of the watershed regions of the negative image in pixels with intensities above the threshold. This procedure can be performed at multiple scales and provides super pixels of corresponding scale. These super pixels can be used to calculate various region statistics of the contained pixels. Examples of images utilizing such super pixels to calculate the mean region intensity of the contrasted image are presented in fig. 4.5 and 4.6

## 4.3 Region Geometry

Super pixels as in the previous section are an example of connected component regions in images. If class or cluster labels are given, connected components of the same class or cluster might have shape information which could provide additional information for the classification task. Below is a list of commonly used scalar valued geometrical interpreters:

The simplest measure of a region is its size  $N$ , which is the number of pixels the region contains.

A scale independent measure of compactness and roundness is provide by eq. 4.10

$$\rho = \frac{A}{2\pi r_g^2} \approx \frac{N}{\frac{2\pi}{N} \sum_{i=1}^N (\mathbf{r}_i - \bar{\mathbf{r}})^2}, \quad (4.10)$$

where  $0 \leq \rho \leq 1$ ,  $A$  is the surface area of the region,  $r_g$  the radius of gyration,  $\mathbf{r}_i$  the coordinate of pixel  $i$  and  $\bar{\mathbf{r}}$  the center of mass of the region. The maximum value of 1 for  $\rho$  is achieved for the circle. Less round and less compact shapes have lower values of  $\rho$ .

Another measure is the elongation of the region given by (eq. 4.11)

$$\varrho = \sqrt{\frac{\lambda_2}{\lambda_1}}, \quad (4.11)$$

where  $0 \leq \varrho \leq 1$  and  $\lambda_1 \geq \lambda_2$  are the pixel coordinate variances along the principle axes of the region.

The next measure is provided by the angle between the first principle axis and the original x-axis given by (eq. 4.12)

$$\varphi = \text{atan2}\left(1, \frac{\Sigma_{12}}{\lambda_1 - \Sigma_{11}}\right), \quad (4.12)$$

where  $\Sigma_{ij}$  are components of the covariance matrix of the region point vectors  $\mathbf{r}$ .

The final measure is the isoperimetric quotient which is a measure of boundary compactness (eq. 4.13)

$$\mathcal{Q} = \frac{4\pi A}{p^2}, \quad (4.13)$$

where  $0 \leq \mathcal{Q} \leq 1$  and  $p$  is the perimeter or boundary length of the region. Again, the maximum value of 1 for  $\mathcal{Q}$  is obtained for the circle and goes towards zero for fractal objects and is low for perforated objects.

## 4.4 Scaled Region PCA

The region properties presented in section 4.3 are standardized, scale invariant measures (except for region size) to describe the geometry of a region. Here a method is introduced to capture even more varied types of geometry.

To this purpose all points in the region are mapped to a scale and partially rotation invariant coordinate system:

$$\begin{pmatrix} \xi_i \\ \eta_i \end{pmatrix} = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{pmatrix} \begin{pmatrix} x_i - \bar{\mathbf{x}} \\ y_i - \bar{\mathbf{y}} \end{pmatrix} \frac{1}{\sqrt{N}} \quad (4.14)$$

where  $\xi_i$  and  $\eta_i$  are the new coordinates of point  $(x_i, y_i)$ ,  $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$  are the region center of mass coordinates,  $\varphi$  is the orientation from 4.12 and  $N$  the region size.

These points are interpolated onto a finite rectangular grid in the new coordinate system. Typical region shapes can be acquired by looking at the principle components or covariance ratio projections of the rectangular grid coordinates of all points in the group.

Often z-score normalized coordinates are used for PCA as well as covariance ratio projection(CVRP). To avoid over-representing coordinates on the new rectangular grid where a lot of the variance comes from outliers, it is proposed here to replace intensity values outside of the  $\alpha$ -quantile of the intensity distribution of each coordinate with random values from within the  $\alpha$ -quantile.

## 4.5 Spiral neighborhoods

It is not immediately obvious how the amount of relevant information for different classifications varies with distance from the pixel to be classified, and often times it is undesirable to densely sample all points in a neighborhood for large distances. It would be good to have a method of choosing sampling locations which are spaced in a regular manner and where the density of these sampling points can be a decreasing function of the center distance.



This is the basis of the proposed spiral based sampling method presented in eq. 4.15

$$\begin{pmatrix} s_x \\ s_y \end{pmatrix} = \bigcup_{i=1}^{n_a} \bigcup_{j=0}^{\frac{f_r(r_m)}{r_s}} \left( \begin{array}{l} \lfloor \cos(2\pi \frac{i}{n_a} + f_\phi(j \cdot \phi_s)) \cdot f_r^{-1}(j \cdot r_s) + \frac{1}{2} \rfloor \\ \lfloor \sin(2\pi \frac{i}{n_a} + f_\phi(j \cdot \phi_s)) \cdot f_r^{-1}(j \cdot r_s) + \frac{1}{2} \rfloor \end{array} \right) \quad (4.15)$$

where  $s_x$  and  $s_y$  are the sets of all  $x$  and  $y$  coordinates of all sampling points,  $n_a$  are the number of spiral arms,  $f_r(r)$  is a radial distance dependent density function and  $f_\phi$  is a angular spacing function.  $r_m$  is the cutoff radius for the spiral  $r_s$  and  $\phi_s$  are radial and angular step constants respectively.

More specifically the simplified parametrization used for later application is presented in eq. 4.16:

$$\begin{pmatrix} s_x \\ s_y \end{pmatrix} = \bigcup_{i=1}^{n_a} \bigcup_{j=0}^{\frac{r_m^{1/\alpha_r}}{r_s}} \left( \begin{array}{l} \lfloor \cos(2\pi \frac{i}{n_a} + (j \cdot \phi_s)^{\alpha_\phi}) \cdot (j \cdot r_s)^{\alpha_r} + \frac{1}{2} \rfloor \\ \lfloor \sin(2\pi \frac{i}{n_a} + (j \cdot \phi_s)^{\alpha_\phi}) \cdot (j \cdot r_s)^{\alpha_r} + \frac{1}{2} \rfloor \end{array} \right) \quad (4.16)$$

where  $\alpha_\phi$  and  $\alpha_r$  are angular and radial stretching exponents respectively. This approach allows for a very simple parametrization of distance dependent sampling point distributions. Examples of “spirals” based on different input parameters are presented in fig. 4.7.

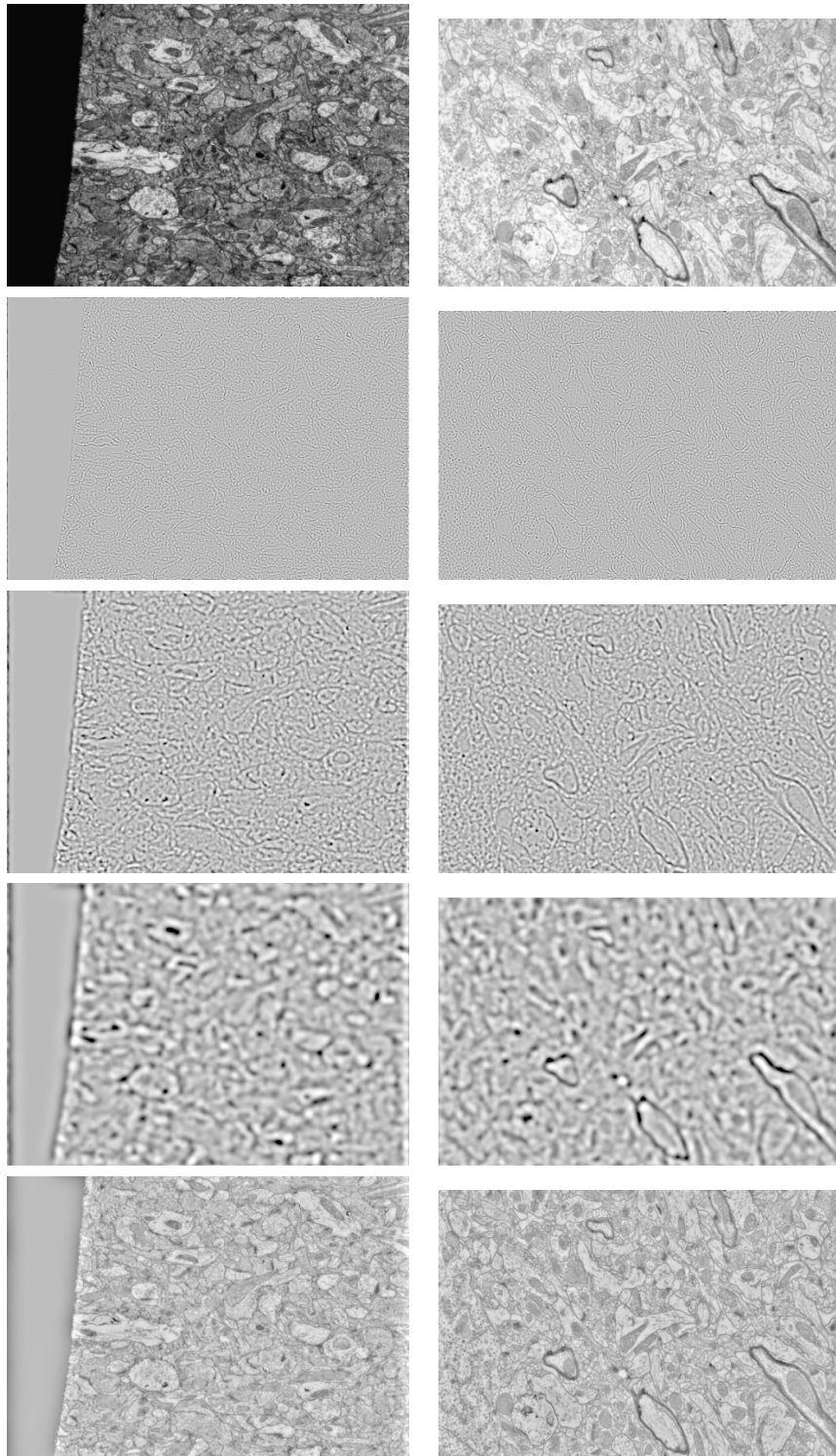


Figure 4.1: Two example EM images used for synapse identification presented at a low resolution. The first row presents raw images. The second through fourth row present selected Log Gabor filtered versions of the images. The bottom row is a contrast adapted version of the image formed by summation of multiple filtered images at different scales

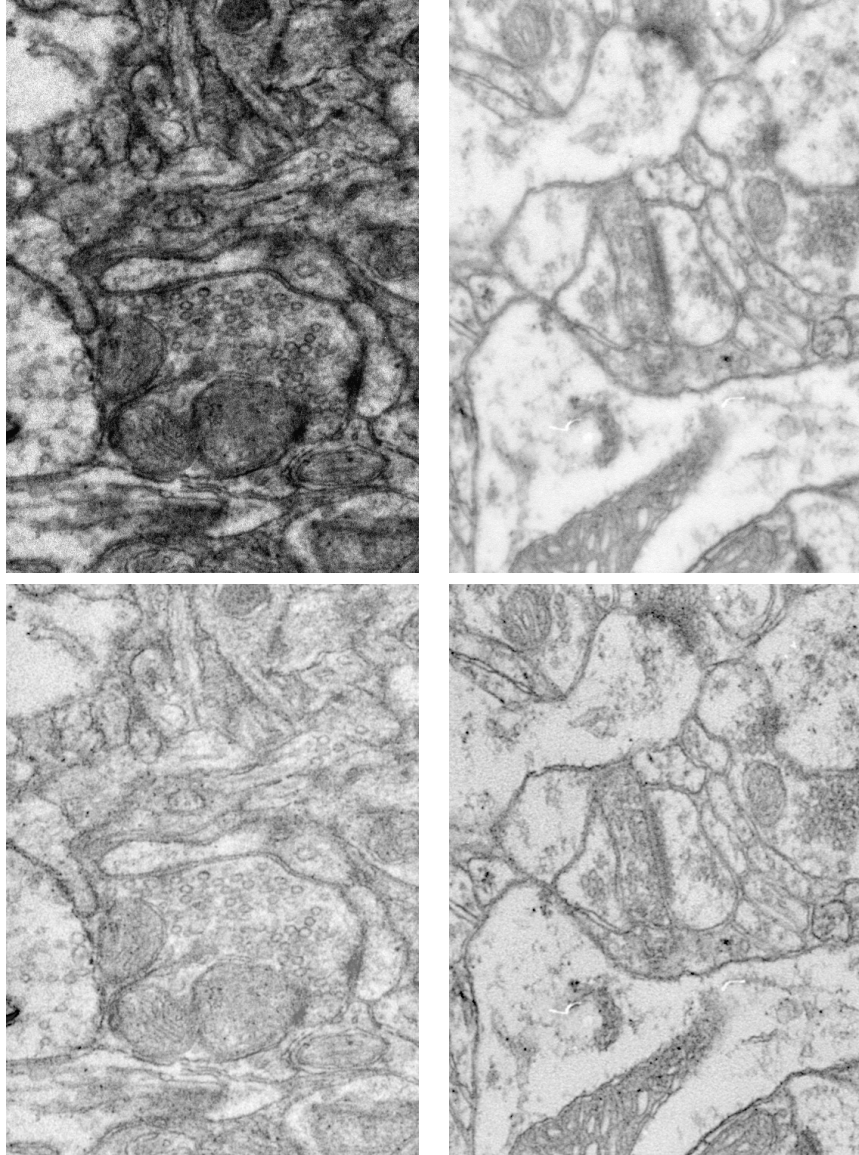


Figure 4.2: Zoomed in to fig. 4.1. Raw and contrast enhanced image via summation of multiple scales of normalized Log Gabor filtered images.

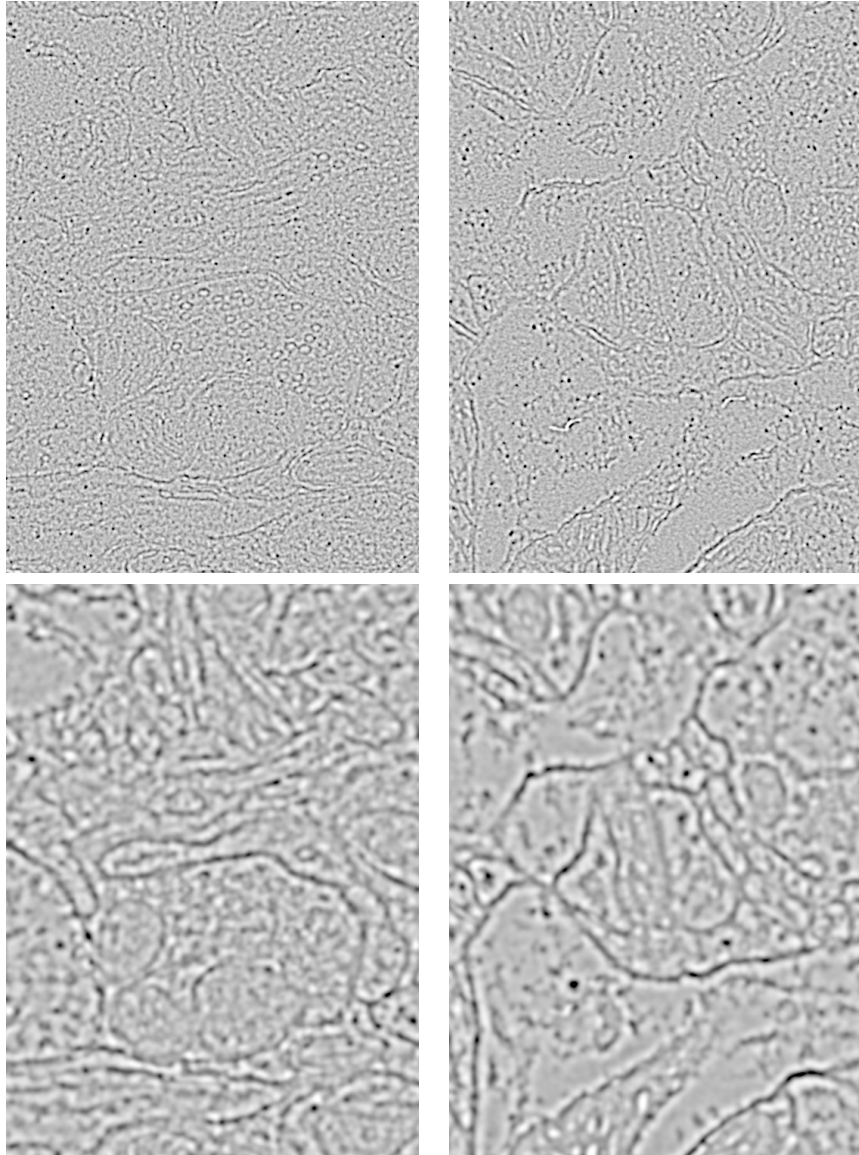


Figure 4.3: Zoomed in to fig. 4.1. Images filtered with different Log Gabor scales.

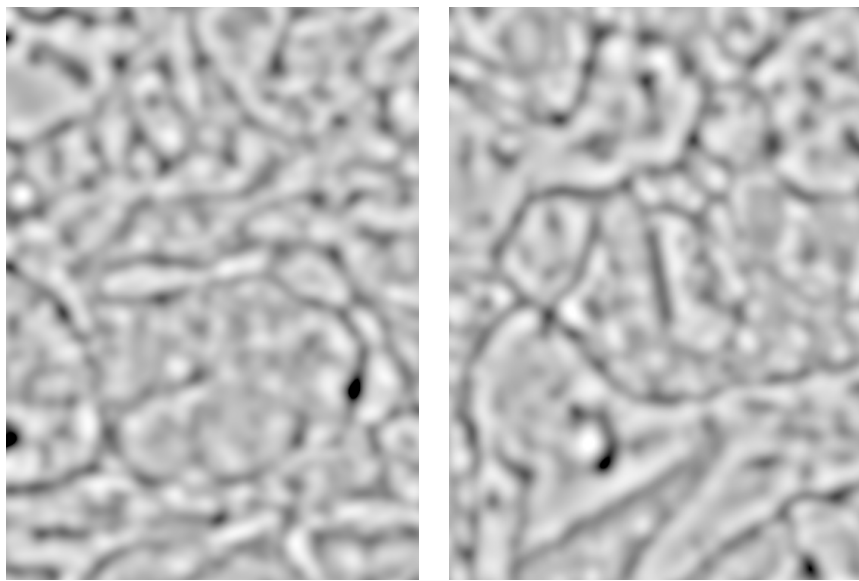


Figure 4.4: Zoomed in to fig. 4.1. Images filtered with a Log Gabor filter with a larger scale constant.

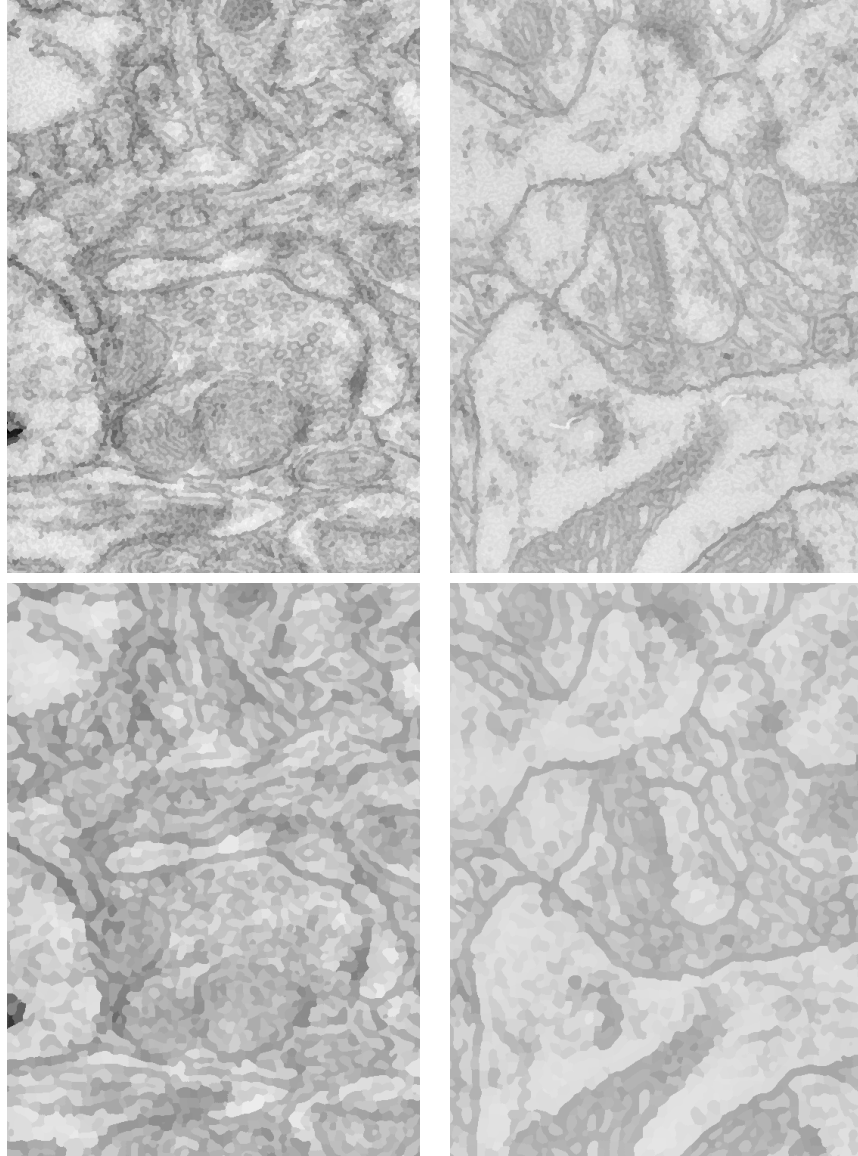


Figure 4.5: Zoomed in to fig. 4.1. Images filtered with different Log Gabor scales followed by a two sided watershed to create region pixels. The images provide mean pixel intensity of the super pixels



Figure 4.6: Zoomed in to fig. 4.1. Images filtered with are Log Gabor filter with a larger spatial constant followed by a two sided watershed to create region pixels. The image provides mean pixel intensity of the super pixels.

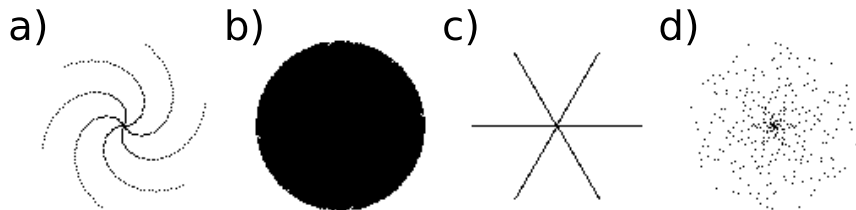


Figure 4.7: Four examples of "spirals" generated with eq. 4.16 .

- a) Simple six armed spiral with  $\alpha_r = 1$
- b) Approximation of a dense disk neighborhood with  $\alpha_r \leq 0.5$
- c) Stencil neighborhood used for line detectors
- d) Progressively sparse neighborhood produced using an angular exponent  $\alpha_\phi > 1$

# Variables

$A$  region surface area - related to  $N$ . 50, 51

$\text{erf}$  error function. 48

$\mathcal{F}^{-1}$  inverse Fourier transform. 49

$f_\delta$  von Mises-Fisher distribution with dimension  $\delta$ . 49

GPB Gabor based band pass filter in the frequency domain. 48

$I$  Bessel function  $I_\delta$  of the first kind of order  $\delta$ . 49

$k$  approximate location of the first zero crossing of a circular log Gabor filter in the spatial domain. 48, 49

LG log Gabor filter. 48, 49

LGB log Gabor based band pass filter in the frequency domain. 48

$\mathcal{N}$  normal distribution. 49–51

$n_a$  number of spiral arms. 51, 52

OLG oriented log Gabor filter. 49

$p$  region boundary length. 51

POLG oriented log Gabor filter with phase component. 49

$\mathcal{Q}$  isoperimetric quotient. 51

$r_g$  radius of gyration. 50

$r_m$  spiral angular stretching exponent. 51, 52



$r_s$  spiral radial step constant. 51, 52

SCN scale adapted normalization applied in the spatial domain. 49

$\alpha$  ratio of Gaussian half width to log Gabor zero-crossing  $k$ . 49

$\alpha_\phi$  spiral angular stretching exponent. 52, 58

$\alpha_r$  spiral radial stretching exponent. 52, 58

$\delta$  dimensionality of the frequency vector. 49

$\kappa$  angular dispersion of the von Mises-Fisher distribution. 49

$\lambda$  eigenvalues of region surface. 50

$\boldsymbol{\mu}$  direction vector of the von Mises-Fisher distribution. 49

$\rho$  region roundness. 50

$\varrho$  region elongation. 50

$\Sigma$  region covariance matrix. 50

$\phi$  phase of the log Gabor filter. 49

$\phi_s$  spiral angular step constant. 51, 52

$\varphi$  region direction angle. 50, 51

$\boldsymbol{\omega}$  spatial frequency vector. 48, 49

$\boldsymbol{\sigma}$  spatial frequency vector. 48, 49

# Glossary

**CVRP** covariance ratio projection dimensions. 51

**PCA** principle component analysis. 49, 51

**rms** root mean square. 49

# Chapter 5

## Reclassification

### Summary

The classifiers presented in previous chapters are only as good as the dimensions which are provided as input to the classifier. Often a significant classification error. In this chapter, by combining approaches related to boosting and considering the geometric relations of classified pixels in images, it can be shown that classification performance can be improved over multiple levels of classification and reclassification. This leads to the concept of an Automaton Forest which is capable of solving a simple classification task which would not be solvable with the standard random forest algorithm considering only unmodified input.

### 5.1 Boosting

In machine learning boosting describes a set of algorithms where single simple classifiers are combined which are learned iteratively by reweighting of the training samples based on misclassified examples. For the standard random forest algorithm training samples are commonly non-weighted, and chosen according to the complete sample distribution. A balanced training set-up takes an equal number of samples from all classes, which generally leads to a more balanced classification accuracy for all classes.

For random forest reclassification we commonly take an equal number of previously correctly classified and incorrectly classified samples for the next classifier which is to be trained. This essentially functions as a balanced reweighting of keeping correct classifications correct, while putting more emphasis on examples which are more difficult to classify correctly.

Here however, opposed to boosting, successive classifiers are not combined to a single classifier, but used in a dependent sequence of classifiers.

## 5.2 Classification feedback

In the context of image classification, pixels are at first often classified individually. However, after a preliminary classification, it is possible to try to improve classification based on the context of the classification of pixels in a neighborhood, or based on group connectivity which can link a non constant number of classification objects. Combining previous classifications and the geometric relations of classified objects allow for the development of second set of classifiers which can function, for example, as smoothing functions, e.g. the second classifier can learn that a pixel should have a similar classification label as its neighbor pixels. This information might not have been tractable from the initial feature set, and is provided only after a first, preliminary classification. This approach can be further expanded by considering connected regions of the same class and providing shape information as further priors for an extended classification. A secondary classifier can correct misclassified pixels based on an expected connectivity of labeled objects. Examples of reclassification forest are presented for image segmentation in chapter 6 and in chapter 7.

## 5.3 Automaton Forest

We consider the following thought experiment: we are interested in determining if a black pixel in a 2D plane is a member of a closed loop or not. The size of the plane can not be determined prior to the loop detection algorithm. Interestingly, this task cannot be solved by a finite feed-forward individual pixel classifier. Such a feed-forward classifier can only be trained on examples of finite sized loops and therefore during testing there can always exist loops which are larger than any previously observed loops (which implies that the task can not be solved by a finite automaton). The task is fairly easy to solve with the following approach: All black pixels which only have one adjacent black pixel do not belong a loop and can be turned into white pixels. This process can be repeated as necessary until only loop pixels remain. Seeking all initial ending points and going through all those, mean that complexity is linear in the number of initial line pixels and the approach is guaranteed to end in a finite amount of time. Based on examples of pixels which are updated according to this rule, a classifier can be trained which individually classifies each pixel and can update classifications to take into consideration locally dependent rules. It should be noted that this requires the way the training data is presented to be changed. It is very important to note that the input image plane is used as a storage for the classification task, and lets the whole classifier function beyond the capacity of a finite automaton.

In practice this approach may be very dependent on correct training examples. In experiments (not shown) learning a simple 8-way connectivity loop classifier requires a few hundred training examples, which all need to be correct examples, for the classifier to function reliably. It should be noted that the simple loop classifier, which uses a minimal amount of contextual information, can lead to an unbounded number of wrongly classified pixels if the classifier is not trained perfectly. Real world tasks might have more stabilizing contextual information, and may be more tolerant to less rigid rules. Automaton forests were tested

for image segmentation, but showed unstable convergence behavior. Typically two to three targeted reclassifications gave the best performance for the data sets chosen in chapter 6 and chapter 7. It can be expected that larger training data sets could make use of more reclassifications, but it is not at all clear how the size of the training data has to increase when adding reclassification layers. It could be that the number of required training samples grows exponentially with the number of neighboring objects to be considered.

# Chapter 6

## EM Segmentation

### Summary

The initial goal of investigating image classification was primarily to provide the tools to automate synapse detection in single slice electron microscopy images. This chapter combines the approaches and methods presented in chapter 3, chapter 4 and chapter 5, to create a classification pipeline for general image classification applied to specifically recognize prominent cell organelles in EM images. The setup was tested on a small training and testing set of 2D electron microscopy images. These images were taken the disectors used in chapter 2 for manual synapse counts. Synapse and two other cell organelle classification tasks were solved in parallel. It is observed that reclassification and parallel classification task solving leads to improvements of classification results using a standard random forest classification approach.

### 6.1 Classification Pipeline

A general purpose image classification pipeline was put together from the various methods presented in the previous chapters. The following are the pipeline components.

#### 6.1.1 Image Decomposition

Each image is decomposed by a set of spatially and angularly specialized Log-Gabor filters (e.g. section 4.1). Each filtered image is normalized by its scale adapted local root mean square (e.g. 4.8). Summing all normalized scale images together provides a contrast enhanced version of the raw image. The individual normalized scale images are thresholded at zero and a distance transform is performed on the resulting binary images. The normalized scale images are used to create super pixels using a two-sided watershed at each scale (e.g. section 4.2). The mean and variance of the contrast adapted image are calculated for each super pixel at each scale and can be accessed by all pixels contained in the super pixel. Additionally, the distance of each pixel in a super pixel to the local minima/maxima of that super pixel,

scaled down by the square root of the number of pixels contained in the super pixel, as well as the angle between the pixel and the local minima/maxima is provided. These steps are highlighted in fig. 6.1.

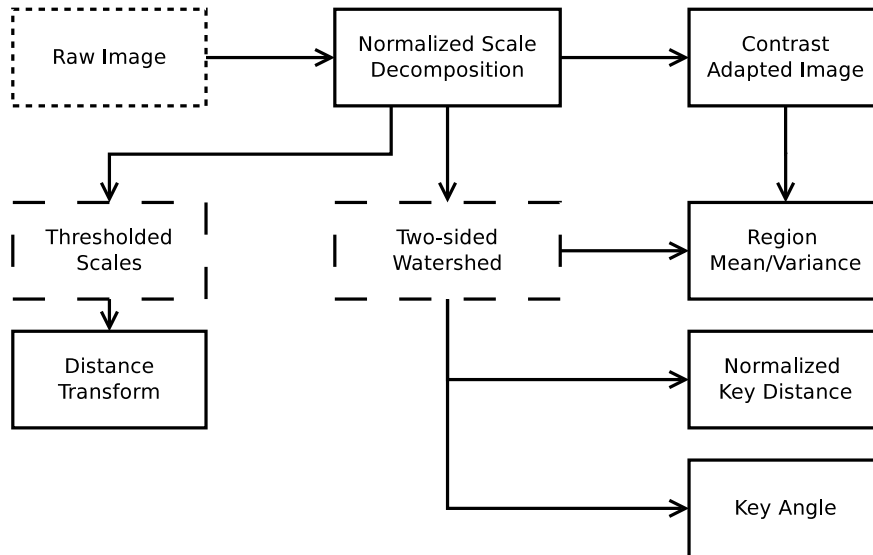


Figure 6.1: This diagram highlights the initial image processing of raw images in the classification pipeline. Dashed boxes represent intermediate processing steps. Drawn out boxes contain image data which can be used directly as input data for later classifiers.

### 6.1.2 Region Analysis

When an image is used for classification or random forest clustering (e.g. section 3.9), connected regions (similar to the watershed super pixels) can be extracted from these images by considering groups of connected pixels with the same label. Similarly, the mean and variance of the pixel intensity values of the contrast adapted image in these regions can be computed. The majority class, as well as class fractions of other labellings or of other clusterings within this region can be evaluated. Each region provides geometrical information derived from all contained pixels (see section 4.3). After finding the orientation of the region, the transformed coordinates of each pixel in the region can be provided and rotated according to orientation and scaled down by the square root of the number of pixels in the region (e.g. section 4.4). All these steps are highlighted in fig. 6.2

### 6.1.3 Iterative Classification

For the final pipeline the raw image is decomposed according to section 6.1.1. This data can be used for all classification levels and helps to stabilize later reclassifications against drifts of the classification. Different than in most standard classification set-ups, we choose to solve multiple, potentially overlapping, classification tasks simultaneously. For each level

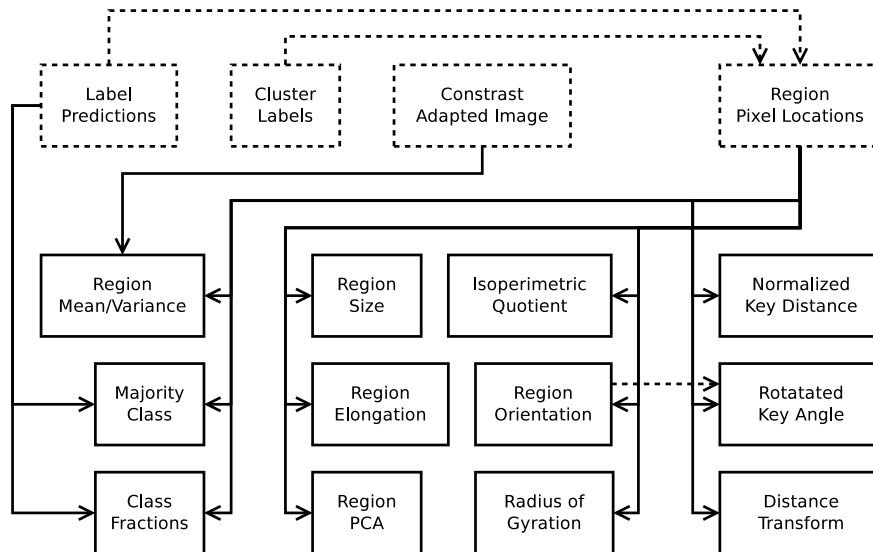


Figure 6.2: This diagram shows typical region information that is retrieved as input for image segmentation. Dotted boxes represent input images. The left group of boxes contain region statistics, the center group region geometry, and the right group provides localization information for region pixels. Dotted lines represent potential input to create regions from. The key point for the key distance and key angle is most commonly the region’s centroid, but can be based on local minima/maxima or distance transforms or other criteria

and for each collection of training labels, an individual random forest classifier is trained based on the images provided by the image decomposition as well as the region analysis of the previous classification level. The labels of the training samples used while training during reclassification contain an equal number of correctly as well as incorrectly classified examples from the previous classification for each class. Region analysis information can be provided to the next classification level, but also to subsequent levels beyond the immediately next, if required. Each random forest classifier at each level can be used for a random forest clustering (see section 3.9) to provide additional regions for analysis. The classification hierarchy is presented in fig. 6.3.

## 6.2 Implementation

All algorithms have been programmed in Java and have been integrated into the framework of ImageJ and FIJI [148], with top level scripts in Jython.

We addressed solving three separate classification tasks simultaneously: Vesicle and mitochondria detection vs everything else (Vesicle and Mitochondria set); cell interior, vesicle, mitochondria and membrane detection (Sparse Organelle); synapse detection with pre- and post synaptic regions vs everything else (Synapse set). The vesicle and mitochondria set is densely labeled, e.g. has a label for each pixel in the training images. The other two sets are



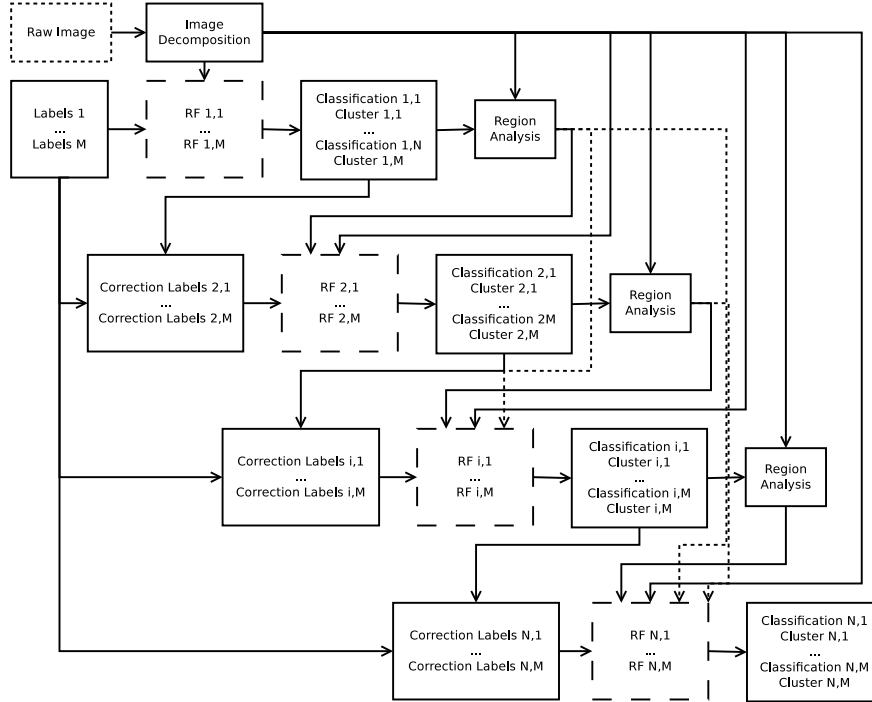


Figure 6.3: This diagram shows a simple representation of the classification pipeline used for EM segmentation. Raw images are decomposed and provided to each classification level. Classification output is used for region analysis and provided to subsequent classification levels. Dotted lines present additional optional processing pathways. RF with numbers are different random forest classifiers at different iterations of the classification process. Correction labels are provided during training only using training images.

only sparsely labeled and focus on pixels that have a subjective (e.g. labeler’s perspective) higher priority to be classified correctly. All training images and labels, as well as classification results, can be found in appendix C. The data set used here consists of ten single section EM images which come from two mice and two cortical areas A1 and M1 in different cortical layers. Examples from A1 come from two different sections at different locations in A1 (all data comes from the data set used in chapter 2). Origins of the images in regards to brain region, section and layer are given in tab. 6.1. The training and test sets for all classification tasks were prepared by a human annotator and are considered as the ground truth reference for the automated classifications presented in this chapter.

For each image each pixel corresponds to a surface area of  $2.5\text{nm} \times 2.5\text{nm} = 6.25\text{n m}^2$ . Each image roughly corresponds to a surface area of  $62.5\mu\text{m}^2$ .

Each of the ten images (except for image G5) from table 6.1 was used to validate a classification system which had been trained data from the other nine images. For the first round of classification an equal number of random pixel locations per class were chosen from the nine training images. Input features for the Random Forest classifiers were taken based on these training pixel locations. For the second and further rounds of classification, a clas-

| Image | Area-Grid | Cortical Layer |
|-------|-----------|----------------|
| B1    | A1-1      | 4              |
| D3    | A1-1      | 5              |
| J5    | A1-1      | 5              |
| C2    | A1-2      | 1              |
| G5    | A1-2      | 1              |
| K4    | A1-2      | 5              |
| C5    | M1-1      | 2/3            |
| D1    | M1-1      | 5              |
| G1    | M1-1      | 5              |
| H5    | M1-1      | 1              |

Table 6.1: Image identifiers and origins used for the classification pipeline testing

sification estimate was made for each pixel in all nine training images. An equal number of random pixel locations per class which had been correctly classified in the first round and an equal number of random pixel locations per class which had been incorrectly classified in the first round were chosen as training samples for the second classifier. Additional features were calculated based on the connectivity of the classification estimates of the training images. The same procedure was repeated for the third round of classification. The resulting classifiers were then applied in sequence to the remaining test image and compared with the ground truth labels for the test image. Training and testing data for each of the label sets as well as for each of the images can be found in appendix C. For the Mitochondria-Vesicle set 5000 training samples were randomly selected per class for each round of classification. For the other two label sets 3000 training samples per class were randomly selected for each round of classification.

## 6.3 Results

The individual, intermediate and final classifications for all labellings for each image can be found in appendix C. The measures “precision” and “recall” were used to get an additional estimate of the performance of the classification pipeline. Precision and recall were calculated only using pixels for which there is a classification label in the test image. This needs to be considered in particular for the sparsely labeled classification tasks. Precision is given as the number of correctly classified pixels of a particular class divided by the total number of pixels assigned to that particular class (only pixels, for which there is a label, are considered). Recall is given as the number of correctly classified pixels of a particular class divided by the total number of pixels that particular class has according to the original labeling (only pixels, for which there is a label, are considered). Precision and recall were calculated for all pixels from all test images combined and results are shown in table 6.2 for precision and in table 6.3 for recall. Results for individual images can be found in appendix C.

Some of these results may seem underwhelming at first glance. A few remarks are needed

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.037, 0.992, 0.294                 | 0.387, 0.665, 0.656, 0.881, 0.863       | 0.982, 0.156, 0.140, 0.151  |
| 1     | 0.067, 0.990, 0.383                 | 0.605, 0.944, 0.729, 0.902, 0.873       | 0.978, 0.293, 0.306, 0.299  |
| 2     | 0.063, 0.988, 0.360                 | 0.661, 0.956, 0.711, 0.899, 0.886       | 0.952, 0.683, 0.678, 0.641  |

Table 6.2: Collective precision values for all images. Results from left to right according to the labeling type. The groups are: vesicles, other and mitochondria; vesicles, myelin, mitochondria, membrane and cell interior; other, synaptic cleft, post-synaptic density and pre-synaptic density. Results from top to bottom show values after each round of classification.

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.682, 0.592, 0.816                 | 0.591, 0.861, 0.799, 0.700, 0.816       | 0.482, 0.926, 0.769, 0.763  |
| 1     | 0.640, 0.758, 0.887                 | 0.620, 0.883, 0.898, 0.841, 0.884       | 0.805, 0.872, 0.733, 0.719  |
| 2     | 0.602, 0.738, 0.910                 | 0.629, 0.873, 0.921, 0.846, 0.887       | 0.974, 0.637, 0.476, 0.515  |

Table 6.3: Collective recall values for all images. Results from left to right according to the labeling type. The groups are: vesicles, other and mitochondria; vesicles, myelin, mitochondria, membrane and cell interior; other, synaptic cleft, post-synaptic density and pre-synaptic density. Results from top to bottom show values after each round of classification.

to help provide insight into the precision and recall values. We start by considering the initially strangest result: the precision of the vesicle class in the Vesicle-Mitochondria set. Precision values actually start out at below 4% for the first iteration. In comparison, when we look a bit further, we see that the Sparse Organelle set also has vesicle labels, yet a much higher precision. It should be restated that the Vesicle-Mitochondria set is densely labeled and that the Sparse Organelle set is sparsely labeled. Furthermore, the vesicles in the Vesicle-Mitochondria set are labeled individually (which introduces a great deal of ambiguity when considering individual vesicles at the pixel level). The actual classification images in appendix C do not seem invalid to the degree the precision value would let on. We also restate that each class is trained with an equal number of training samples, even when one class might contain more than ten times the samples of another. The classifier marks whole regions as vesicles, even though a large fraction of the pixels in the region are not labeled as vesicles. Here we can note that measuring precision might be less useful when considering labellings, where one class has much fewer items than the other. Also, we consider that sparse labellings can provide us with a measures of precision and recall, which better reflect our perception and intuition. The observations are important when interpreting the values in tables 6.2 and 6.3.

What can be done with less reservation, is comparing precision and recall values within a label set at different stages of reclassification. Many precision values greatly increase from the first round of classification to the second and while the others remain fairly stable. Recall does not increase as much, though many recall values increase. Some values are stable,

and some decrease slightly. Although the random forest algorithm isn't the most sophisticated classification algorithm, it tends to give reasonable classifications without requiring too many training samples. However, the improvement of classification performance at the first reclassification goes beyond the effect of only adding an equivalent amount of training data. Observing the differences in the classification images in appendix C, it can also be observed that the classifications are frequently smoothed based on neighboring classifications, or even removed if a region shape is not correct, or if the classifier of another set disagrees in regards to class type.

The second reclassification does not have as strong an impact. The results are already mostly stable (possibly the classification pipeline is able to exhaust the current training set using as few training samples), but there are still a few increasing values. The Vesicle-Mitochondria and the Sparse Organelle sets are fairly stable, but the Synapse set still changes drastically. Many misclassifications are detected and removed, though also a significant number of actual synapses are removed. Despite missing synapses, the resulting images have a much stronger visual semblance of marking synapses with pre- and postsynaptic densities (e.g. appendix C).

## 6.4 Discussion

There are only around 200 example synapses, 1000 example mitochondria and less than 100 myelinated cell examples during training. Further there are only nine images which can be used to generalize to new images. These images have slightly different section thicknesses and there are slightly different directional sheering effects on the sections based on ultra-thin section cutting. While the training data and test data overlap in animal and cortical area, the sampling sights are far enough apart and are distinct single sections. There are few parameters for the setup beyond number of training samples and choosing a broad enough set of scales, and these parameters were chosen independently of the data (scales for the filters were chosen based on typical image sizes) and the same configuration could easily be used for another image segmentation task. While the precision and recall values may not be too high, the generalization capability of the segmentation pipeline is considerable given the training setup. The visual results (also the errors) reflect some similarities to human classifications of novice classifiers. Various aspects of the approach deserve at least more investigation and expansion, in particular the effects of context based reclassification and information sharing of different classification tasks for various settings.

# Chapter 7

## Random Forest Source Separation

### Summary

The classification setup which has been used for image recognition in chapter 6, in particular organelle recognition in 2D electron microscopy images can be used for various different classification tasks. Another problem that was tackled in collaboration with Saurabh Bhargava, is single channel source separation of audio signals. The reclassification methods presented in chapter 5 are useful in increasing the accuracy of reconstructions of individual sources extracted from mixed source audio signals. This work was published together with Saurabh Bhargava [141].

### 7.1 Introduction

The idea behind audio source separation, is that a listener has a limited number of recording devices (e.g. ears for humans and other mammals), but needs to distinguish multiple sound sources at once, or focus on one of multiple sound sources while ignoring all other sources. This task finds many applications in speech recognition, music processing, medical signal processing, hearing aid development, etc ([94],[172],[80],[173],[87],[63],[128]). One approach to solving source separation is Computational Auditory Scene Analysis(CASA) which uses a binary masking of the mixture spectrogram signal for demixing ([4],[23],[60],[86],[121],[140]). The Ideal Binary Mask(IBM) is the spectrogram mask applied to an audio mixture which should return an approximation of the source signal. Under certain conditions speech, signals reconstructed with the IBM provide the best possible reconstruction in regards to signal-to-noise-ratio(SNR) among all possible binary masks (BMs) [99]. Such IBMs can be used as labels for a classification architecture, such as the one presented in chapter 5.

In the following we present a Reclassification Random Forest(RRF) setup (initially called Multi-Layered Random Forest(MLRF)) and compare it with two methods commonly used in the field of audio source separation, namely: non-negative sparse coding(NNSC) and recurrent neural networks(RNN).

## 7.2 Dynamic Range Compression

For the case that long term amplitude fluctuations of the signals are not of interest, there is the option for dynamic range compression, e.g. the signal can be normalized by the average of a temporal Gaussian window (similar to the local normalization in section 4.1):

$$\tilde{s}(t) = \frac{s(t)}{\sqrt{\mathcal{N}(0, \sigma^2) * s(t)^2}}, \quad (7.1)$$

where  $\mathcal{N}(\mu, \sigma^2)$  is the normal distribution with mean  $\mu$ , variance  $\sigma^2$  and  $s(t)$  the original audio signal. This formulation implicitly assumes that the long term average of  $s(t)$  is 0 (compare eq. 4.8 for spatial normalization). The larger the temporal smoothing constant  $\sigma$  is, the more similar eq. 7.1 becomes to standard root mean squared(RMS) normalization.

## 7.3 Spectrograms and Binary Masks

Often the separation task will become more tractable in a higher dimensional space, in particular when the separating function is based on splitting planes. To this purpose a spectrogram  $S[\tau, f]$  of the signal  $\tilde{s}(t)$  is created using the absolute value of the Short Time Fourier Transform(STFT):

$$S[\tau, f] = |\text{STFT}(\tilde{s}(t))|, \quad (7.2)$$

where  $\tau$  is the temporal offset and  $f$  the spectral frequency.

The spectrogram provides the amount of power contained in a specific frequency band and a given discrete time point. The mixtures referred to here are composed of two input signals  $x(t)$  and  $y(t)$ . The IBM is constructed by comparing the spectrograms of both input signals in each time-frequency(T-F) bin and determining which of the two signals has a larger amplitude. Here, a more general form of IBM is presented:

$$IBM(\tau_1, \tau_2), f = \begin{cases} 1 & X[\tau_1, f] \leq Y[\tau_2, f] \\ 0 & X[\tau_1, f] > Y[\tau_2, f] \end{cases}, \quad (7.3)$$

where  $\tau_1 - \tau_2$  is the time shift between both signals. The commonly found definition of IBM in the literature is equivalent to this definition with  $\tau_1 = \tau_2 = \tau$ .

## 7.4 Classification of Time Frequency Bins

The previously introduced IBM can be used as a basis for training labels for a classifier such as the random forest. In chapter 3 it was required that the distribution of feature vectors underlying the classification labels differ from one another. Many audio signals have different power distributions for each frequency band, and we expect that the distributions and prevalence of the labels will differ in each frequency band. For this purpose a random forest classifier can be trained for each individual frequency band.

## 7.5 Sampling of Training T-F Bin Locations

Given training data for both audio sources, a training mixture can be produced by summing the two corresponding spectrograms and reverting them to the audio domain. A priori, the two training sources are assumed to be independent of another. From this it results that any sum of the two sources with a relative temporal shift towards another is a potential training mixture. This results in a total of  $L_x \cdot L_y$  possible training mixtures, where  $L_x$  and  $L_y$  are the number of temporal columns in the spectrograms  $X$  and  $Y$ , respectively. As the amount of training data and hence the number of temporal columns for both sources increases the number of possible mixtures becomes prohibitively large to express explicitly. We have observed that a balanced number of training examples from all classes typically leads to lower error rates for all classes simultaneously. We want to draw an equal number of samples from both sources according to the IBM label given in eq. 7.3. For both sources individually all  $L_x$  resp.  $L_y$  T-F bins at a particular frequency  $f$  are sorted according to their spectrogram value. For each value  $x_{if}$  in  $X^f$  (spectrogram values of  $X$  at frequency  $f$ ) find the number  $\hat{x}_{if}$  of T-F bins in  $Y^f$  such that:  $\hat{x}_{if} = \sum_{j=1}^{L_y} \mathbf{1}_{\geq y_{jf}}(x_{if})$ . Efficiently finding all values for  $\hat{x}_{if}$  requires  $Y^f$  to be presorted, so that the largest  $y_{jf} \leq x_{if}$  can be found via binary search. for The cumulative sum vector  $\vec{q}$  of all  $\hat{x}_{if}$  with

$$q_i = \sum_{j=1}^i \hat{x}_{jf} \quad (7.4)$$

provides a list of values which can be compared with a uniformly random value between 0 and  $q_{L_x-1}$  for the corresponding time  $\tau_1$  of  $x_{\tau_1 f}$ . A corresponding  $\tau_2$  of  $y_{\tau_2 f}$  can be sampled by choosing a random  $\tau_2$  with  $y_{\tau_2 f} \leq x_{\tau_1 f}$ . This procedure provides  $X$ -dominant T-F bin locations and can easily be adapted to acquire  $Y$ -dominant T-F bin locations. All in all, the time complexity of drawing  $n_x$   $X$ -dominant and  $n_y$   $Y$ -dominant locations has a complexity of  $O((L_x + n_y) \cdot \log(L_x) + (L_y + n_x) \cdot \log(L_y))$ .

## 7.6 Feature Extraction

Provided center locations for the training points, it is possible to create feature vectors which capture local spectral-temporal patterns of each audio source within the mixture. The T-F neighborhood of a sample location contains the vector of the sums of all T-F bin pairs in a temporal range  $\{\tau_1 - w_{\tau_1}, \tau_1 + w_{\tau_2}\}$  and spectral range  $\{f - w_{f_1}, f + w_{f_1}\}$  for  $X$  and in a temporal range  $\{\tau_2 - w_{\tau_1}, \tau_2 + w_{\tau_2}\}$  and spectral range  $\{f - w_{f_1}, f + w_{f_1}\}$  for  $Y$ , where  $w_{\tau_1}, w_{\tau_2}, w_{f_1}$  and  $w_{f_2}$  provide the temporal and spectral expanse of the neighborhood. Three types of features are extracted using such mixture neighborhoods. The first type of feature dimensions contain the immediate mixture spectrogram values of the neighborhood. The second type of feature values come from applying a linear dimension reduction similar to CVRP (see 3.4) to the neighborhood window. CVRP was modified so that eq. 3.8 was

replaced with:

$$\Sigma(\mathbf{X}) \Sigma(\mathbf{X}|_j)^{-1} \Sigma(\mathbf{X}|_j)^{-1} \Sigma(\mathbf{X}) \cdot \mathbf{u}_j = \lambda \cdot \mathbf{u}_j \quad (7.5)$$

The first few resulting eigenvectors are chosen as projection vectors for the neighborhood window. Finally, the same projection method is used, however using the z-score normalized values of the neighborhood window. Multiple, different neighborhood window sizes can be chosen for the projections mentioned in this section. The feature selection process is highlighted in fig. 7.1.

## 7.7 Reclassification

After training a preliminary classifier for the separation task, the resulting classification of the training data can be incorporated into a next classifier for reclassification. Feature extraction for the second and further classifiers is expanded by considering neighborhood windows and covariance ratio projection(CVRP)-like projections on the classification labels (see fig. 7.1). The previous method of sampling training T-F bin locations could be applied here as well, but is less useful, as T-F bin neighborhood windows have to be fully classified for feature extraction, which results in a large computational overhead. Instead, we randomly selected large blocks of the source spectrograms concatenated these along the time axis and then overlaid both concatenated spectrograms to form a training mixture spectrogram. It is important to note that an equal number of training locations for both sources were chosen for reclassification, with the additional condition of there being an equal number of previously correctly and incorrectly classified locations. The training of a reclassifier, and reclassifying the resulting training data can be performed multiple times, but can result in an exhaustion of the training data and a corresponding overfitting.

## 7.8 Experimental Setup

To evaluate performance we implemented this method in MATLAB and applied it to speech data from the GRID corpus database [28], as well as artificially generated pink noise. All audio signals have a sample rate of 16 kHz. Performance was evaluated on three groups of audio mixtures viz. Male-Female, Male-Male and Male-pink noise with five different pairs in each group. For each speech source used, we used 100 sentences from the GRID corpus database. Eighty of the sentences, corresponding to roughly two minutes and ten seconds of clean speech were used as training data, while the remaining twenty sentences were concatenated, corresponding to roughly 40 seconds of speech. The normalization  $\sigma$  from eq. 7.1 was chosen to be 1 second, which is longer than typical temporal patterns, such as phonemes, found in human speech [144]. This normalization  $\sigma$  led to higher SNR gains for all methods tested in comparison to a pure RMS normalization of the entire duration. Artificial test mixtures were generated by summing the test audio spectrograms of both speakers at an input SNR of zero dB (both sources are equally loud). To compute the spectrograms, a Fourier window size of 1024 samples (equivalent to 64ms) and 75% overlap



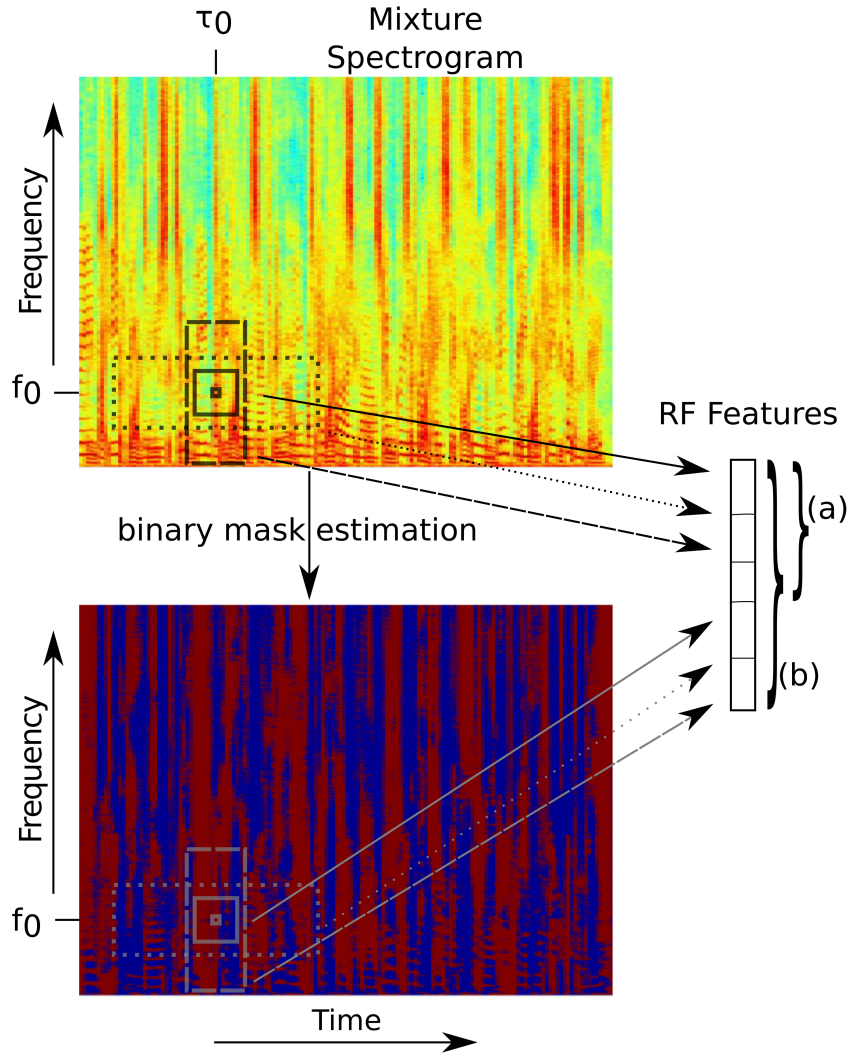


Figure 7.1: Spectral-temporal neighborhood windows used for feature extraction. Top: Multiple neighborhood windows (dotted and dashed lines) with the center T-F bin  $(\tau_0, f_0)$  in a log-power spectrogram of a speech mixture sample. Bottom: BM estimate of the first classifier with blue for mask value 0 and red mask value 1. Input vectors to the first layer (a) are formed from linear combinations of neighborhood windows of the mixture spectrogram and to the second layer (b) are formed from linear combinations of neighborhood windows of the mixture spectrogram and from the mask estimates of the first classifier (slanted arrows).

between successive Hanning windows were used. This results in 513 distinct frequency bands for the spectrogram. Optimal parameters for all methods were chosen by maximizing the output SNR of a different, single randomly selected male-female pair. For the current method the number of random splitting dimensions for each decision tree node ( $mtry$ ), the number of decision trees ( $n_{tree}$ ), the number of classification layers ( $l$ ) and the training samples per layer

( $n_{sl}$ ) were set to  $mtry = 64$ ,  $n_{tree} = 300$ ,  $l = 2$  and  $n_{s1} = 15000$  and  $n_{s2} = 7000$  respectively. Performance saturated (no decline beyond saturation) with the chosen number of decision trees as well as number of training samples per layer. Adding more layers beyond two to the classification didn't improve performance, and it is likely that the current training data was exhausted by the two layers of classification. The reclassification layer did not improve the performance for the male-pink noise mixtures. Pink noise contains less spectro-temporal patterns which allows the classifier to overspecialize on the male speaker. For the first layer, nine different neighborhood windows of various size combinations were chosen and four neighborhood windows of various sizes were chosen for the second layer. The optimal parameters for RNN were chosen by maximizing the output SNR and are in agreement with [78]. The number of layers and nodes per layer were chosen to be 3 and 1000 respectively. The optimal parameter values for NNSC were chosen as in [149] which were also obtained by maximizing the SNR.

## 7.9 Results

### 7.9.1 Parameters

The performance of this source separation approach was evaluated using 4 metrics viz. SNR, classification accuracy (Acc) as the percentage of correctly classified T-F bins between  $BM_{test}$  and  $IBM_{test}$ , Perceptual Evaluation of Speech Quality (PESQ) [76] and Short-Time Objective Intelligibility (STOI) [165]. The SNR values reflect how the distance of the reconstructed signal to the target signal in the audio domain. The PESQ score reflects the quality of the reconstructed speech while the STOI score reflects its intelligibility. For the male-female case, the average power overlap in the individual frequency bands is smaller compared to that in the male-male case. Therefore, in the male-female case a good separation based on frequency specialized classifiers can be expected, while the male-male case requires better recognition of local, source specific T-F patterns for good separation. The male-pink noise case poses a different challenge, as signals typically overlap more strongly within individual T-F bins and therefore making a separation difficult for any of the binary classification methods. Results are summarized in Table 7.1.

### 7.9.2 Evaluation Metrics

Four measures were used to evaluate the performance of the audio source separation.

#### Signal-to-Noise-Ratio

SNR is a measure that describes how similar two audio signals are in regards to squared difference of both signals. It is given by the following, for a target signal  $\tilde{x}(t)$  and the estimate of the signal  $\hat{x}(t)$ :

$$\text{SNR} = 10 \log_{10} \left( \frac{\int \hat{x}(t)^2 dt}{\int (\tilde{x}(t) - \hat{x}(t))^2 dt} \right) \quad (7.6)$$

## Accuracy

Acc is given by the number of correctly classified T-F bins when comparing the mixture's IBM and the estimated BM:

$$\text{ACC} = \frac{\sum_f \sum_\tau \text{BM}_{f, \tau} \oplus \text{BM}_{f, \tau}}{\sum_f \sum_\tau \text{IBM}_{f, \tau} \oplus \text{BM}_{f, \tau}}, \quad (7.7)$$

with  $\oplus$  is the elementwise XOR operator. Accuracy has the advantage that it doesn't depend on the signal power contained within a frequency band, and provides a general statement on the binary spectro-temporal separability of the two signals. However, other measures are needed to provide a statement on context relevant information which is preserved in the separation.

## Perceptual Evaluation of Speech Quality

PESQ scores (quote) are computed for the target audio signal  $\tilde{x}(t)$  and the estimated audio signal  $\hat{x}(t)$ . PESQ models subjective tests of human perception of speech quality and is a standardized measure by the International Telecommunication Union (ITU). More detailed interpretations of PESQ scores are found in [76].

## Short-Time Objective Intelligibility

STOI scores (quote) are computed for the target audio signal  $\tilde{x}(t)$  and the estimated audio signal  $\hat{x}(t)$ . STOI is based on speech intelligibility at short (386ms) time scales and has shown good correlation with human perceived speech intelligibility [165].

### 7.9.3 Evaluation

Results for the four metrics are provided in table 7.1 as well as in fig. 7.2-7.5.

In regards to SNR (fig. 7.2), MLRF and RNN perform comparably for all speaker combinations, with neither significantly outperforming the other. Both methods significantly outperform the linear methods, which indicates the separation task is noticeably non-linear, also in the spectro-temporal domain.

The current method is built to classify individual T-F bins of all frequencies, which makes it less surprising the MLRF has the best accuracies for all speaker combinations, although not significantly for the male-pink noise case.

The effects of SNR values and Acc on the human perception of audio signals is not straight forward. We suspect that human audio and in particular speech recognition will require reconstructed signals which have a good combination of weighting frequency bands

| Method   | Mean SNR (dB)<br>(MF,MM,MN) | Mean Acc (%)<br>(MF,MM,MN)     | Mean PESQ<br>(MF,MM,MN)             | Mean STOI<br>(MF,MM,MN)      |
|----------|-----------------------------|--------------------------------|-------------------------------------|------------------------------|
| MLRF     | <b>(10.32,8.01,8.60)</b>    | <b>(78.10*, 77.62*, 92.59)</b> | <b>(2.30*, 2.00, 1.57)</b>          | <b>(0.87*, 0.86*, 0.79*)</b> |
| RNN      | (10.24, <b>8.72</b> , 8.52) | (60.43, 62.37, 88.17)          | (1.98, 1.70, 1.57)                  | (0.83, 0.83, 0.76)           |
| NNSC     | (7.24, 5.11, 6.51)          | Not applicable                 | (2.15, <b>2.06</b> , <b>1.98*</b> ) | (0.81, 0.79, 0.77)           |
| NNSC(BM) | (7.51, 4.59, 7.17)          | (66.5, 63.38, 86.71)           | (2.03, 1.79, 1.63)                  | (0.80, 0.76, 0.74)           |

Table 7.1: Results for audio source separation for the methods: MLRF, RNN, NNSC and NNSC(BM) (binary masked NNSC) with in regards to the metrics SNR, Acc, PESQ and STOI for speaker pairs of male-female(MF), male-male(MM) and male-pink noise(MN). The results are averages of five pairs. The best scores for each metric are presented in bold. The value has an \* if the value was significantly better than the second best result using a two-sided t-test with  $p < 0.05$ .

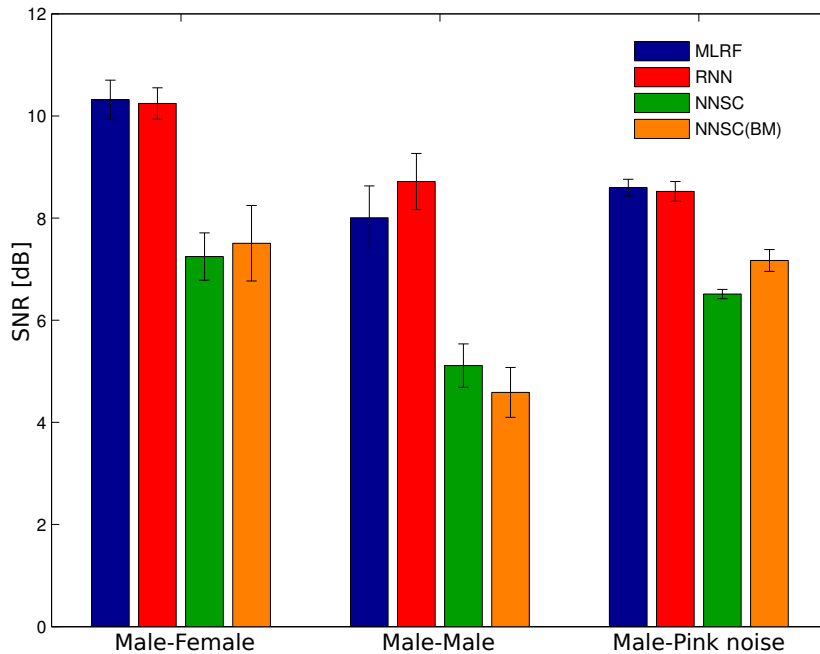


Figure 7.2: SNR values in dB with error bars for the methods MLRF, RNN, NNSC and NNSC(BM). Three speaker pairings are presented: male-female, male-male, male-pink noise.

and correctly classifying key T-F bins with a high accuracy. PESQ scores are used to estimate human perceived quality of speech in telephony. A curious observation, is that NNSC which underperformed in regards to SNR and Acc, had significantly higher PESQ scores for the male-pink noise case. It can be interpreted that binary classification has a detrimental effect on audio quality, in particular when masker and target strongly overlap within individual T-F bins in the mixture. This suggests that biology does not use a binary masking when performing source separation tasks.

STOI focuses on speech intelligibility as opposed to speech quality. That these two are not

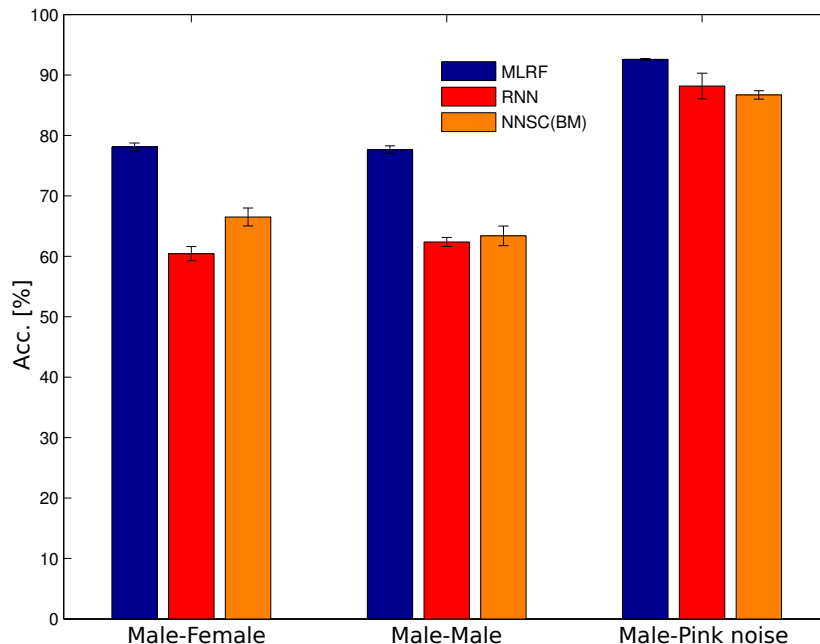


Figure 7.3: Accuracy values in percent with error bars for the methods MLRF, RNN, NNSC and NNSC(BM). Three speaker pairings are presented: male-female, male-male, male-pink noise.

the same is reflected in the scoring results of STOI. Here MLRF significantly outperforms the other methods in all three pairing cases. This supports results where it previously shown that an improved quality of the reconstructed speech (higher PESQ score), does not necessarily lead to an improvement in speech intelligibility (quote).

### 7.9.4 Frequency-wise Analysis

Different frequencies can have different roles in the perception of sound (quotes). This is also reflected in the results from the different results of the previous section. Therefore the source separation was also analyzed in a frequency specific manner. Mean squared error between the target and estimated frequency bands (example fig. 7.6) indicates better performance of MLRF in the male-female pairings (see appendix D.1), in particular in higher frequency bands ( $> 1kHz$ ). MLRF takes an equal number of samples for each source in each frequency band, irrespective of relative power ratio of both sources in that band. As a result, high recall is achieved for the target even if its power within the band is low in respect to the masker. Conversely for the other methods, the full audio content of the masker remains present in frequency bands where the masker has less power. The result is a worse masker suppression and allows a possibly unwanted reconstruction of the masker signal from the target.

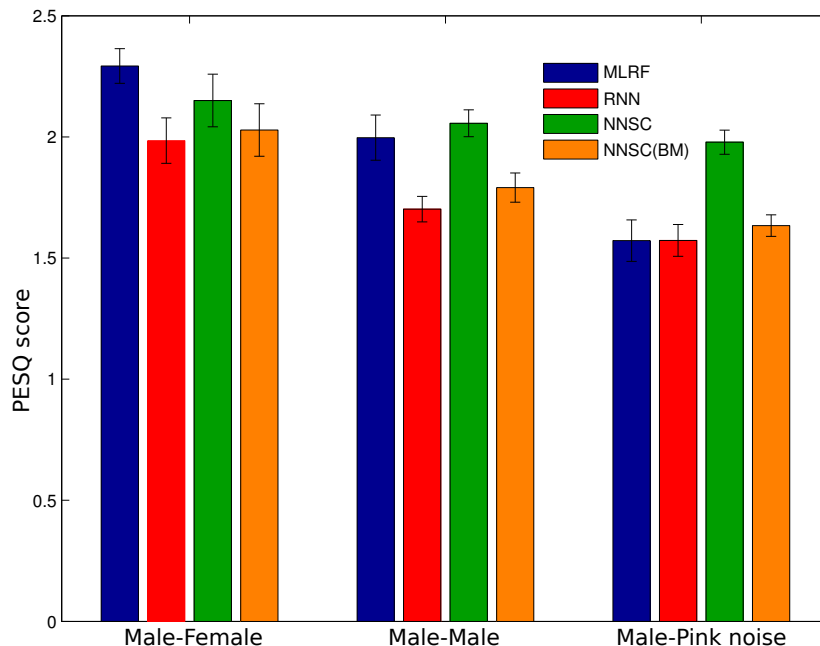


Figure 7.4: PESQ values with error bars for the methods MLRF, RNN, NNSC and NNSC(BM). Three speaker pairings are presented: male-female, male-male, male-pink noise.

### 7.9.5 Discussion

In this section a method based on a reclassifying, frequency-wise random forest was presented for audio source separation. Based on four metrics the method outperforms the linear approach NNSC, except in regards to PESQ scores in the separation of male speakers and pink noise. The method also compares favorably in comparison to a more state-of-the-art approach using deep recurrent neural networks, in particular in regards to STOI as well as PESQ in the male-female and male-male cases. However, we believe the emphasis should not be on which classification algorithm is better, deep neural networks or random forest, but on the fact that considering all frequency bands as well as using a context based reclassification provide significant improvements for audio source separation. Investigating the individual frequency bands and considering human derived metrics can help understand the human approach of source separation better. In future work, a further evaluation metric for comparison could be provided by letting a language recognition system work with the various reconstructions and check how the system’s recognition score improves based on the separation approach. In this context it would also be good to check masker suppression, and see to what degree a masker signal can be reconstructed from a target.

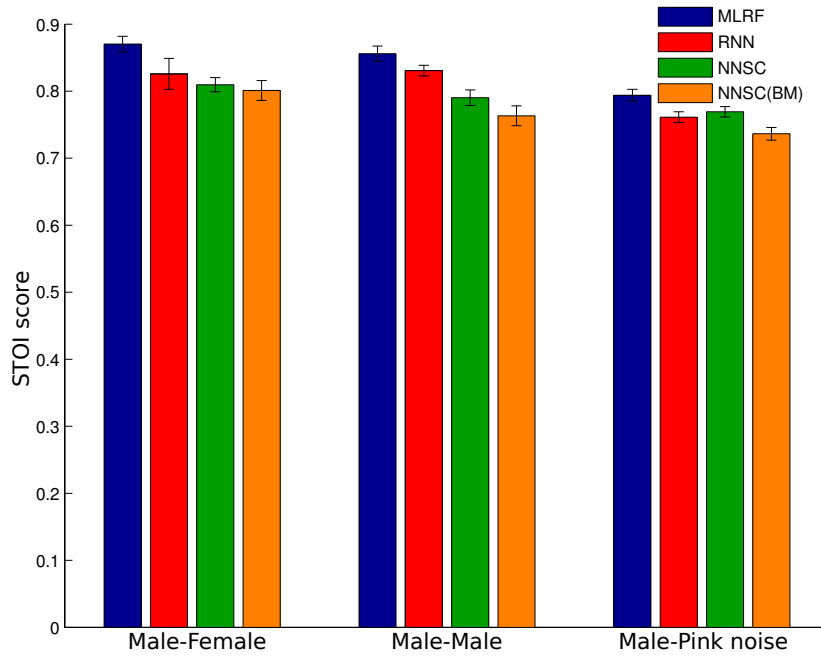


Figure 7.5: STOI values with error bars for the methods MLRF, RNN, NNSC and NNSC(BM). Three speaker pairings are presented: male-female, male-male, male-pink noise.

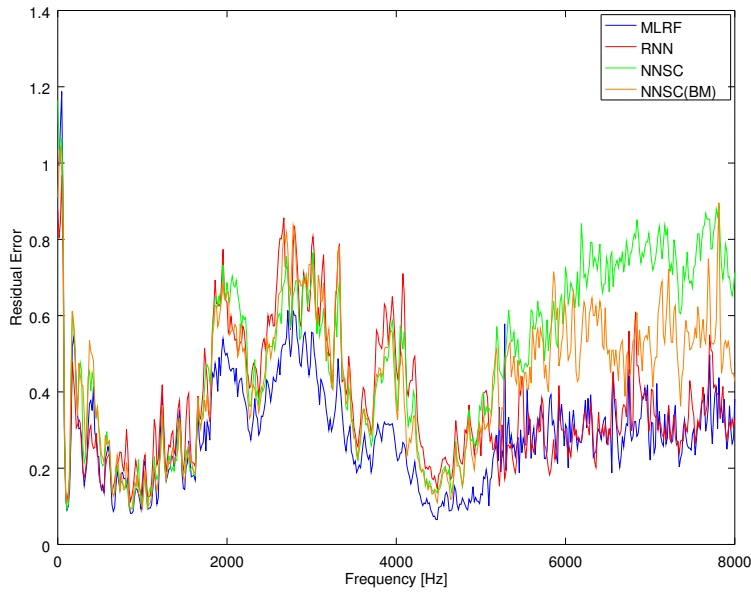


Figure 7.6: Normalized residuals for each frequency band for a male example from a male-female pairing. MLRF has lower residuals for the intermediate frequency range, but has similar residuals to RNN otherwise

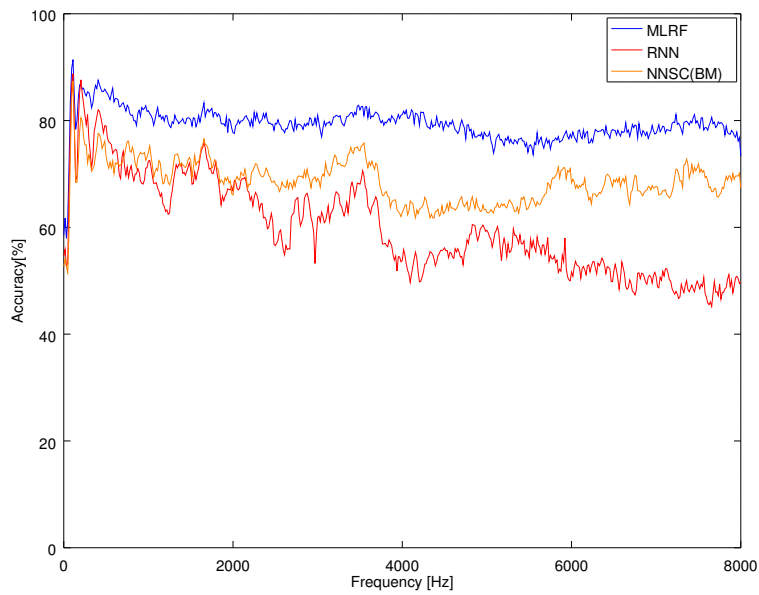


Figure 7.7: Classification accuracy within each frequency band for a male-female pairing. MLRF has fairly constant accuracy values at all frequencies, while the other methods focus on classifying high power bands correctly.



# Variables

$f$  spectral frequency for spectrograms. 70, 71, 73, 75

$\mathbf{1}$  indicator function. 71

$L$  length of corresponding training set. 71

$\mathcal{N}$  normal distribution. 70

$t$  time for the waveform domain. 70, 74, 75

$\sigma$  half-width of the temporal smoothing. 70, 72

$\tau$  temporal offset for spectrograms. 70, 71, 73, 75

# Glossary

- Acc** classification accuracy for T-F bins. 74–76
- BM** binary mask. 73–79
- CASA** computational auditory scene analysis. 69
- CVRP** covariance ratio projection dimensions. 72
- IBM** ideal binary mask. 69, 70, 74, 75
- MLRF** multi-layered random forest. 69, 75–80
- NNSC** non-negative sparse coding. 69, 74, 76–79
- PESQ** perceptual evaluation of speech quality. 74–78
- RMS** root mean squared. 70
- RNN** recurrent neural networks. 69, 74–79
- RRF** reclassification random forest. 69
- SNR** signal-to-noise-ratio. 69, 72–76
- STFT** short time Fourier transform. 70
- STOI** short-time objective intelligibility. 74–79
- T-F** time-frequency (bin). 70–76

# Chapter 8

## Rate-based Neural Networks for Amplification

### Summary

The goal of measuring synaptic connectivity is to estimate general cortical connectivity patterns. Such connectivity measurements, for example as from the cat, already give some connectivity constraints for cortical processing which can be used as a basis for modeling the firing behavior of neuron populations. In this chapter some basic connectivity constraints are presented, as well as two simple, modular, local synaptic learning rules for excitation and inhibition. Finally, resulting rate firing dynamics of a whole neuron population based on the proposed network rules are presented.

### 8.1 Network Constraints

Biological neural networks are subject to various constraints regarding connectivity. Some constraints are due to physical restrictions of space and energy consumption and some are due to the adequacy of a “simple” system to perform some desired task. The following presents a few general, restrictive connectivity patterns found in Mammalian neocortex:

- Sparsity of long-range connectivity vs local connectivity

The number of long-range inputs into a cortical area is only a fraction of the number of recurrent connections within the area [12], [162]. It can therefore be expected that the fraction of external input (from an other area) a single cortical neuron receives will be substantially smaller than the input from cortical neurons residing in the same area.

- Strong recurrent excitation and inhibition

The distance from a neuron’s resting potential to its threshold potential is around 15-20 mV [152]. Each synapse can relay an excitatory post-synaptic potential(EPSP) with an average

amplitude around 0.1-1.5mV ([18],[147],[103]) to its target which on average can provide between 0.5%-10% of the potential needed to bring the neuron's membrane potential from baseline to threshold . Considering that each neuron receives on average 1'000-10'000 input synapses and when disregarding the exponential decay of EPSPs this corresponds to an impressive excitatory input amplification of a factor of 5-1000. Factors such as inhibitory input, exponential decay of EPSPs over time and adaptation of firing rates provide means of counteracting this strong excitation and allowing stable network activity.

- Local connectivity

Cortical neurons receive many inputs, but spatial constraints and sparsity of possible connections still prevent all to all connectivity within the area. Further, the typical spatial expanse of inhibitory axonal arbors is on average less than that of excitatory arbors [168].

A model which aims to explain cortical processing should be able to function under the constraints listed above.

## 8.2 Neuron Model

Here we modeled neurons as discrete time rectified units, which is a discrete extension of linear rectified units (slightly modified from [58]) with additional control terms and of the form:

$$x_i(t + 1) = \frac{R(\sum_{j=1}^n w_{ij}(\frac{x_j(t)}{\zeta_i})^{\Omega_i} \zeta_i - \theta_i + \eta_i x_i(t))}{1 + \eta_i}, \quad (8.1)$$

where  $R(x) = \max(x, 0)$  is the ramp function,  $x_j(t)$  is the firing rate of unit  $j$  at time  $t$ ,  $w_{ij}$  is the weight of the synapse from unit  $j$  onto unit  $i$ ,  $\zeta_i$  is a firing scaling factor,  $\Omega_i$  is a firing amplification exponent,  $\theta_i$  is the firing threshold of unit  $i$  and  $\eta_i$  is a constant for temporal smoothing of the firing rates.

The model neurons essentially average and then amplify incoming activity according to synapse weights, under consideration of inhibitory input. The amplification exponent  $\Omega_i$  allows for better balance control of excitation and inhibition. As a crude model it can be used to reflect differences in the EPSP and inhibitory post-synaptic potential(EPSP) decay rates as well as other various forms of current leakage at increased firing rates. The temporal smoothing constant  $\eta_i$  provides a means of controlling the discreteness of a network simulation. Excitatory and inhibitory units are distinguished in that outgoing synapse weights of excitatory units are always positive and outgoing weights of inhibitory units are always negative.

## 8.3 Learning Model

Various rules for synaptic learning have been established for feed-forward (non-recurrent) networks of artificial neurons. However, many statements on stability and convergence are

lost or unclear when these same learning rules are applied in a network with recurrent connectivity. The following learning rule takes inspiration from such learning rules, but is also modular to a degree that aspects of the learning can be investigated fairly independently. Going back to the neuron model in eq. 8.1 we can rewrite a unit's input weights as:

$$w_{ij} = A_{i\mathbf{k}} \cdot \tilde{w}_{ij}, \quad (8.2)$$

where  $A_{i\mathbf{k}}$  is the input amplitude of the  $k^{th}$  unit compartment which contains a subset of all synapses formed onto unit  $i$  and  $\tilde{w}_{ij} = \frac{w_{ij}}{\sum_{l \in \mathbf{k}} w_{il}}$  is the normalized input weight (by compartment) of unit  $j$  onto unit  $i$  with  $\sum_{l \in \mathbf{k}} \tilde{w}_{il} = 1$  and  $\tilde{w}_{ij} \geq 0$ . These amplitudes are used to separate the learning of general input magnification and the learning of relative weights between different synaptic inputs. Having different compartments allows for input type specific learning, for example to distinguish learning time scales of local cortical inputs, and external inputs.

To regulate the system and keep neuron in an optimal long term firing regime, it is useful to keep the average firing rate of a unit constant. The following learning component is used to perturb the unit in this direction:

$$A_{i\mathbf{k}}(t+1) = A_{i\mathbf{k}}(t) + \text{sgn}(A_{i\mathbf{k}}(t)) \cdot \delta_{\mathbf{k}} |\Theta_{\mathbf{k}} - \bar{x}_{i\tau}|^{\gamma_{\mathbf{k}}} \cdot \text{sgn}(\Theta_{\mathbf{k}} - \bar{x}_{i\tau}), \quad (8.3)$$

where  $\text{sgn}$  is the signum function,  $\delta_{\mathbf{k}}$  is the amplitude learning rate of compartment  $\mathbf{k}$ ,  $\bar{x}_{i\tau}$  is the exponential moving average firing rate of unit  $x_i$  with time constant  $\tau$ ,  $\gamma_{\mathbf{k}}$  is a scaling component for compartment  $\mathbf{k}$  and  $\Theta_{\mathbf{k}}$  is the target average firing rate according to  $\mathbf{k}$ .

Individual excitatory synapse weight learning is derived from Hebbian learning:

$$r_{ij} = (x_i(\mathbf{k})x_j(\mathbf{k}))^{\beta_{i\mathbf{k}}}, \quad (8.4)$$

where  $\beta_{i\mathbf{k}}$  is a scaling exponent for the hebbian learning.

This is factored in to a more general synaptic learning rule for excitatory synapses:

$$\tilde{w}_{ij}(t+1) = \tilde{w}_{ij}(t) + r_{ij} \cdot \alpha_{\mathbf{k}} (|\mathbf{k}| \tilde{w}_{ij}(t) + 1)^{-\gamma_{\mathbf{k}}} T_{ij}^{exc}, \quad (8.5)$$

where  $\alpha_{\mathbf{k}}$  is the synaptic learning rate of compartment  $\mathbf{k}$ ,  $\gamma_{\mathbf{k}}$  is an exponent for synaptic weight balancing and  $T_{ij}^{exc}$  given by:

$$T_{ij}^{exc} = \exp\left(-\frac{|\bar{x}_{i\tau} - \Theta_{\mathbf{k}}| + |\bar{x}_{j\tau} - \Theta_{\mathbf{k}}|}{2\sigma_{\mathbf{k}}}\right), \quad (8.6)$$

where  $\sigma_{\mathbf{k}}$  is a tolerance parameter for deviant average firing for compartment  $\mathbf{k}$ . The exponent  $\gamma_{\mathbf{k}}$  can be used to prevent the learning from forming bimodal distributions. The tolerance  $T_{ij}^{exc}$  focuses learning to units which have appropriate long term firing averages.

The learning rule for inhibitory synapses is a bit more involved:

$$\tilde{w}_{ij}(t+1) = \tilde{w}_{ij}(t) + r_{ij} \cdot \alpha_{\mathbf{k}} (\text{H}(\bar{x}_{i\tau} - \Theta_{\mathbf{k}} \cdot \phi_{\mathbf{k}}) \cdot (|\mathbf{k}| \tilde{w}_{ij}(w) + 1)^{-\gamma_{\mathbf{k}}} - \text{H}(\tau_{\mathbf{k}} \cdot \frac{1}{\phi_{\mathbf{k}}} - \bar{x}_{i\tau})) T_{ij}^{inh}, \quad (8.7)$$

where  $H(x)$  is the Heaviside function,  $\phi_{\mathbf{k}}$  is a scaling factor for the long term average to determine if there should be inhibitory learning and its sign and  $T_{ij}^{inh}$  is given by:

$$T_{ij}^{inh} = \exp\left(-\frac{|x_i(t) - x_i(t-1)| + |x_j(t) - x_j(t-1)|}{2\sigma_{\mathbf{k}}}\right), \quad (8.8)$$

where  $\sigma_{\mathbf{k}}$  is a tolerance parameter for fluctuating firing rates for compartment  $\mathbf{k}$ . The tolerance  $\sigma_{\mathbf{k}}$  is used to focus on inhibitory learning between slow changing neurons. Otherwise, the neurons shift between Hebbian and anti-Hebbian learning, depending on the neuron's average firing rate.

At each timestep  $\tilde{w}_{ij}(t+1)$  is renormalized to guaranty  $\sum_{j \in \mathbf{k}} \tilde{w}_{ij} = 1$  and  $\tilde{w}_{ij} \geq 0$ .

The first exponential term in eq. 8.6 diminishes synaptic learning for and from units with aberrant firing averages outside of the tolerance given by  $\sigma_{\mathbf{k}}$ . This is important to balance synaptic learning against amplitude learning of the unit. The second exponential term allows for a focus on weights between units which promote slow firing adaptation.

All the learning rules presented here require only local information, or information which is accumulated for the whole cell (e.g. statistics of the synapse, as well as statistics of its input, as well as the neuron's firing rate, and average firing rate)

## 8.4 Evaluation Measures

A network of model neurons as presented in the previous sections can be uniquely represented by a state vector  $\mathbf{x}(t)$ , which contains the momentary firing rates of each unit  $x_i$  at the time point  $t$ , and the weight matrix  $W_t$  with entries  $w_{tij}$  which are the synaptic weights from unit  $j$  unto unit  $i$  at time point  $t$ . Different networks can be compared by their temporal response to different inputs, as well as by comparing the evolution of their weight matrices. Using this representation, common network properties can be defined for these comparisons.

### 8.4.1 Equilibrium Firing Statistics

If the system is allowed to evolve for sufficiently long from an initial state  $\mathbf{x}(0)$  at time  $t = 0$ , it can be assumed that the system will converge towards a system attractor (should constant input be mentioned here?). Given a sufficiently large time window  $w_t = [\tau_1, \tau_2]$  starting at time  $\tau_1$  and ending at time  $\tau_2$  it can be assumed that various statistics of this attractor can be estimated with some confidence. Measures over time window  $w_t$  used here are presented in the following:

- Cumulative distribution of the average firing rate of a neuron population

The average activation, or firing rate, of each neuron allows for the comparison of population firing distributions given different static inputs to the network. The Cumulative Distribution Function (CDF) is chosen to avoid difficulties regarding bin sizes and outliers which may be present in histogram representations, as well as allowing for an easier comparison of many CDFs at once. These CDFs can further be compared to the CDF over all units of the average firing rate for different network inputs. (See fig. 8.3 for an example)

- Cumulative distribution of the standard deviation of firing rates of a neuron population
- Projection of mean and standard deviation onto the first two principle components of the union of population responses to different inputs.

Different inputs  $\mathbf{b}_j$  result in different population responses  $\mathbf{x}_j(t)$ . principle component analysis(PCA) can be used to find the general directions of largest variance for all inputs at once by considering the union  $\mathbf{x}_{bw}$  of all discrete time points within  $w_t$  of all population responses  $\mathbf{x}_j(t)$ . Projecting the mean population firing rates unto the first two principle components provides a tool for the visualization of the separation of responses based on different inputs. (See fig. 8.5 for an example)

### 8.4.2 Equilibrium Self-Similarity

Network attractors can exhibit oscillatory behavior which may or may not contain repeating patterns. Using the autocorrelation function is a common approach to estimate the duration of such a repetition. Here a related, but slightly different approach is used to address this problem. Given an equilibrium network response  $\mathbf{x}$ , the temporal sum of squared firing differences of this signal to a shifted version of another network response  $\mathbf{y}$  over the same network units is given by:

$$\begin{aligned}
d^2(\mathbf{x}, \mathbf{y}, \tau) &= \int_{t_s}^{t_e - \tau} (\mathbf{x}(t) - \mathbf{y}(t + \tau))^T (\mathbf{x}(t) - \mathbf{y}(t + \tau)) dt = \int_{t_s}^{t_e - \tau} \sum_{i=1}^n (x_i(t) - y_i(t + \tau))^2 dt \\
&= \sum_{i=1}^n \int_{t_s}^{t_e - \tau} (x_i(t) - y_i(t + \tau))^2 dt = \sum_{i=1}^n \int_{t_s}^{t_e - \tau} (x_i(t)^2 - 2x_i(t)y_i(t + \tau) + y_i(t + \tau)^2) dt \\
&= \sum_{i=1}^n \int_{t_s}^{t_e - \tau} x_i(t)^2 dt + \int_{t_s}^{t_e - \tau} y_i(t + \tau)^2 dt - 2(x_i \star y_i)(\tau), \quad (8.9)
\end{aligned}$$

where  $\tau$  is the time shift between  $\mathbf{x}$  and  $\mathbf{y}$ ,  $t_s$  and  $t_e$  are the start and end times to consider for the signals  $\mathbf{x}$  and  $\mathbf{y}$ ,  $n$  is the number of units in the network and  $\star$  is the cross-correlation operator. In particular the temporal average of squared firing differences is given by:

$$\overline{d^2}(\mathbf{x}, \mathbf{y}, \tau) = \frac{d^2(\mathbf{x}, \mathbf{y}, \tau)}{t_e - \tau - t_s} \quad (8.10)$$

It should be noted that domain  $\int_{t_s}^{t_e - \tau}$  is chosen for integration. This is used as a condition to guaranty that  $\mathbf{x}$  and  $\mathbf{y}$  are both defined at all time points considered. Evaluating  $\overline{d^2}$  has a complexity of  $O(n \cdot t \log(t))$ , where  $t$  is the number of time points considered in the interval  $[t_s, t_e]$ . If this function has zero values for any time  $\tau$ , this directly implies that the signal is periodic with a period of  $\tau$ .

### 8.4.3 Reassignment Similarity

Another measure of similarity between signals, which is introduced here, called reassignment similarity, captures the distance between two sets of multi-dimensional points. This measure is defined for two sets with an equal number of points  $m$  which all reside in the same euclidean space  $\mathcal{R}^n$ . Pairs of points  $\mathbf{x}(t_1)$  and  $\mathbf{y}(t_2)$  (taken as sample points from the network configurations of  $\mathbf{x}$  and  $\mathbf{y}$  at time points  $t_1$  and  $t_2$  respectively) have a well defined distance metric given by:

$$d(\mathbf{x}(t_1), \mathbf{y}(t_2)) = \sqrt{\sum_{i=1}^n (x_i(t_1) - y_i(t_2))^2} \quad (8.11)$$

The reassignment similarity is defined as the minimum sum of distances between all disjoint pairs of points between the sets  $\mathbf{x}_m$  and  $\mathbf{y}_m$  divided by  $m$ , the number of such pairs. This is more formally given by:

$$rs(\mathbf{x}_m, \mathbf{y}_m) = \min_f \frac{\sum_{i=1}^m d(\mathbf{x}(i), f(\mathbf{x}(i)))}{m}, \quad (8.12)$$

where the function  $f$  is a bijection  $f: \mathbf{x}_m \rightarrow \mathbf{y}_m$ .

Eq. 8.12 is an example of the assignment problem and can be solved in polynomial time, for example using the Hungarian algorithm [97]. This measure allows for the comparison of different network equilibria as well as comparison of perturbed and non-perturbed network states.

## 8.5 Network Examples

The goal of the network simulations was to find a set of network parameters and learning rules which let the network evolve towards a steady state under the constraints that the sum of external input weights is small in comparison to the recurrent input weights, that neurons have a spatially limited connectivity and that within the steady state, units should exhibit different firing statistics for different inputs.

For this we used a simple network of recurrently connected units. The units are on a 28x28x5 3D grid (x-y-z) with periodic boundary conditions and connectivity is random and restricted by the distance between units on the x-y plane. Four of the five layers in z-direction contain excitatory units and the fifth layer is for inhibitory units. For initial testing the MNIST data base (quote origin) was used as a constant input to the network. Pixel intensities of the MNIST images were directly converted to firing rates and provided to the network via constant input synapses. Excitatory units within the network have three compartments: one for external excitatory input synapses, one for excitatory feedback synapses from the network and one for inhibitory feedback synapses from the network. Inhibitory units have two compartments: one for excitatory feedback synapses from the network and



one for inhibitory feedback synapses from the network. In all following discussions there is no learning in the input compartment and the input compartment amplitude  $A_{i0}(t)$  is 0.1 for all neurons and at all times. In all simulations, excitatory synapses have an equal probability of being formed on neighbors within a cylinder of radius 7. Inhibitory units form synapses on neighbors within a cylinder of radius 3.5. In case necessary, maximum firing rates are capped at 4000. The temporal smoothing constant was set  $\eta_i = 4$  for all networks.

### 8.5.1 Linear Excitatory Network

The simplest network used here was a linear network ( $\Omega_i = 1, \forall i$  with no inhibition ( $A_{i2} = 0$  for excitatory units and  $A_{i1} = 0$  for inhibitory units) and constrained excitation ( $A_{i1} = 0.9$  for excitatory units and  $A_{i0} = 0.9$  for inhibitory units)) was used as a reference network.

The weight matrix of this simple linear recurrent network can be written as:

$$W = aP, \quad (8.13)$$

where  $a$  is the common amplitude of all recurrently connected units and  $P$  is a right stochastic matrix with  $\|P_{i,:}\|_1 = 1 \forall i$  from which follows  $\|P\|_1 = 1$ . This provides an upper bound of  $\lambda_1 \leq 1$  for the largest eigenvalue of  $P$ . It can be concluded from

$$\lambda_1 \cdot \mathbf{1} = P \cdot \mathbf{1}, \quad (8.14)$$

with  $\mathbf{1} = (1, 1, \dots, 1)^T$  that the largest eigenvalue of  $P$  is  $\lambda_1 = 1$ . The development of the system with a constant input  $\mathbf{b}$  can be written as follows:

$$\begin{aligned} \mathbf{x}(t+1) &= W\mathbf{x}(t) + \mathbf{b} = W(W\mathbf{x}(t-1) + \mathbf{b}) + \mathbf{b} \\ &= \left( \sum_{k=0}^t W^k \right) \mathbf{b} + W^{t+1} \mathbf{x}(0) = (I - W)^{-1} (I - W^{t+1}) \mathbf{b} + W^{t+1} \mathbf{x}(0) \end{aligned} \quad (8.15)$$

When  $a < 1$  the system converges towards a steady state vector of the form  $(I - W)^{-1} \mathbf{b}$  for  $t \rightarrow \infty$ . The system diverges for  $a \geq 1$ .

The largest eigenvalue of this linear system can be determined analytically using eq. 8.13 and 8.14 as:

$$\lambda_1 = \frac{\eta_i}{1 + \eta_i} + \frac{a}{1 + \eta_i} = 0.8 + 0.18 = 0.98 \quad (8.16)$$

From this it is expected that the system will converge towards a point attractor as can be observed in fig. 8.1.

The distribution of mean firing of the units has a fairly narrow spread as can be seen from the cumulative distribution of mean neuron firing rate (fig. 8.3), or from the projections of the means for 50 inputs onto the first two principle components of the 50 inputs combined is shown in fig. 8.2.

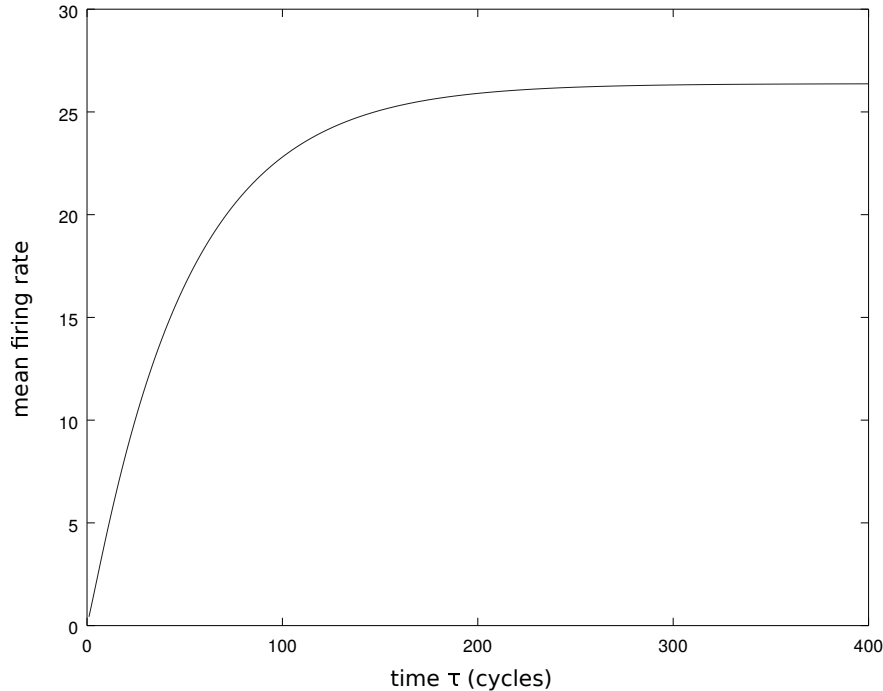


Figure 8.1: Simple linear excitatory network. Mean firing response of the network for a specific input. The network reaches full activation after about 300 time steps.

### 8.5.2 Rectified Linear Network with Inhibition

The excitatory linear network from the previous section produces outputs which are well separable, in particular because all inputs result a unique fix point. The excitatory linear system can amplify inputs and keep them separable. However, stronger amplification pushes the network closer towards emulating the matrix inversion of a singular matrix, where small errors in the network weights can disrupt the amplification or cause the network to become unstable. The linear excitatory network has a memory of previous inputs which decays exponentially as can be seen from the last term in eq. 8.15. To expand upon the previous linear network, a network including inhibition was simulated. This inhibition is strong ( $A_{i2} = 20$  for excitatory units and  $A_{i1} = 20$  for inhibitory units) in comparison to excitation ( $A_{i1} = 3$  for excitatory units and  $A_{i0} = 3$  for inhibitory units). This network has a different distribution of mean unit firing rates as compared to the previous network as can be seen in 8.4 (compare fig. 8.3). A large fraction of the units do not fire for a particular input and a smaller, but still substantial fraction of the units (around 20%) do not fire for any of the inputs. The equilibrium firing of the units is no longer given by input specific fix points, but various types of more complex attractors. The projections of the means for 50 inputs onto the first two principle components of the 50 inputs combined is shown in fig. 8.5. It becomes apparent that the network responses are more difficult to separate due to the overlap of the ellipses.

Individual equilibrium firing patterns range from fix point attractors, over saddle-like

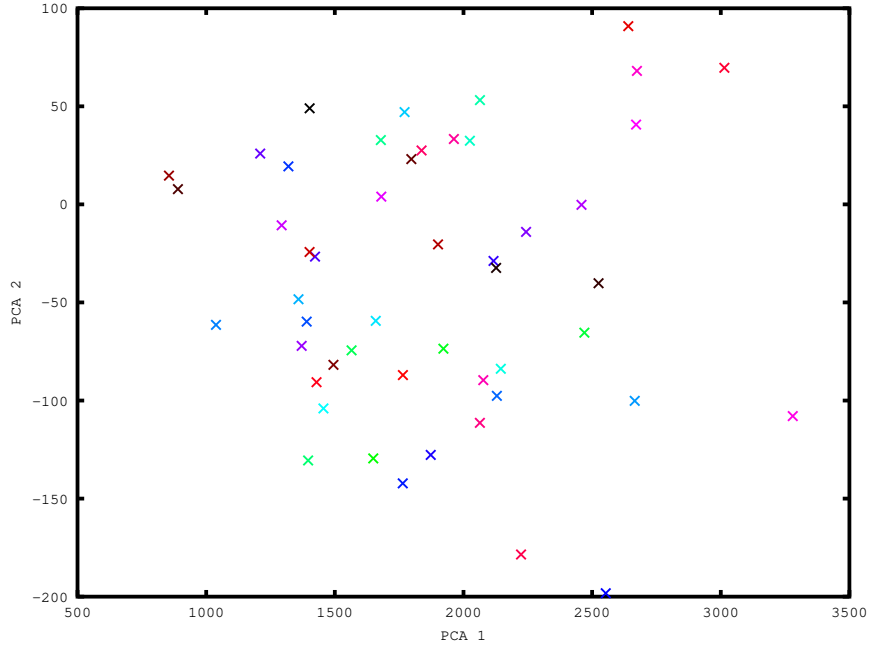


Figure 8.2: Simple linear excitatory network.  $\times$ 's show mean values of the neuron firing rates, given 50 different inputs, projected onto the first two principle components of all responses to all inputs. Ellipses provide the standard deviation for each input in both directions. For the excitatory linear network, the standard deviation of individual neurons is zero.

shapes (fig. 8.6), to partially regular attractors (fig. 8.7), to irregular attractors (fig. 8.8).

The three attractors in fig. 8.6-8.8 provide a good opportunity to check for periodicity using eq. 8.10. Applying eq. 8.10 to the signal against itself result in fig. 8.9-8.11. In the case of the saddle it is apparent that the attractor is periodic, where the period can be found by measuring the time of the first zero value (with some tolerance) of the curve after initiation. It should be noted that here eq. 8.10 guaranties that the saddle is periodic for all dimensions, not just the first three PCA dimensions. From fig. 8.10-8.11 it becomes clear that this is not true for the other two attractors.

It is also possible to test the reassignment similarity given by eq. 8.12 to get a measure of how much a signal changes given different perturbations. If the assignment problem in eq. 8.12 is solved using the Hungarian Algorithm [97], the time complexity of the task is  $O(n^3)$ , where  $n$  is the number of time points compared within the assignment problem. As example, one can consider the self-similarity (no perturbation) of each of the previously presented attractors on the time interval  $I_1 = [5000, 6000]$  when sampling 200 points without replacement and solving the assignment problem in eq. 8.12. Standard errors are obtained by resampling 100 times. For the attractor from fig. 8.6 a an assigment error of  $23.3 \pm 0.55$  is obtained. Similarly for fig. 8.7 an error of  $118.2 \pm 0.62$  is obtained and for fig. 8.8 an error of  $134.7 \pm 0.58$ . These errors result only from the random sampling of points within the interval. Different intervals can be compared to get a general impression of the similarity

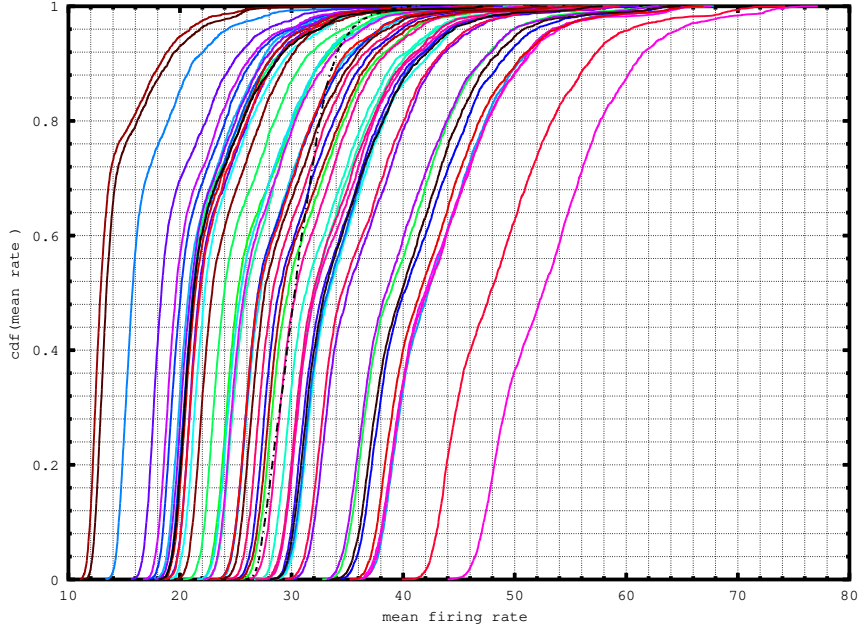


Figure 8.3: Simple linear excitatory network. Cumulative distribution function of the neurons mean firing rate for 50 different inputs

of neighborhoods of the attractor. For this, the interval  $I_1$  is compared with the intervals  $I_2 = [3000, 4000]$  to obtain assignment errors of:  $24.31 \pm 0.61$  for fig. 8.6,  $176.6 \pm 0.96$  for fig. 8.7 and  $292.7 \pm 0.60$  for fig. 8.8. We consider what happens when the system input is perturbed for a certain amount of time. To test this, the previously presented attractors were presented with another input image starting at  $t = 3000$  for 1000 time steps. A first question is: Does the attractor return to the region of the phase space? Again, the similarity of the attractor with the perturbed attractor can be compared in  $I_3 = [9000, 10000]$ . Assignment errors are:  $26.4 \pm 1.0$  for fig. 8.6,  $165.2 \pm 1.1$  for fig. 8.7 and  $279.4 \pm 0.6$  for fig. 8.8 which indicates the attractors remain in the same region of the phase once the perturbation is over. What happens during the actual perturbation in the time interval  $I_2$ ? Here the assignment errors between the perturbed and the non-perturbed system are  $817.7 \pm 0.3$  for fig. 8.6,  $449.8.2 \pm 1.1$  for fig. 8.7 and  $489.2 \pm 0.7$  for fig. 8.8. These values can be compared with the assignment errors between the perturbed system and a system which uses the perturbing image as a constant input: The assignment errors are  $180.2 \pm 0.3$  for fig. 8.6,  $265.6 \pm 0.3$  for fig. 8.7 and  $451.3 \pm 1.3$  for fig. 8.8. From this we follow that as with the excitatory linear network that the rectified linear network has a decaying memory of previous (constant) inputs. Self-similarity of the signal is measured taking the assignment error of 200 points within the time windows  $I_1$  and  $I_4 = [3500, 4000]$  with the same input. The self assignment error is  $271.26 \pm 19.9$  for the first 49 training images of the MNIST data set. The assignment error of each attractor to the attractor of the subsequent input image, within time window  $I_4$  for both inputs, is  $461.16 \pm 20.5$ . The assignment error of the signal while perturbing

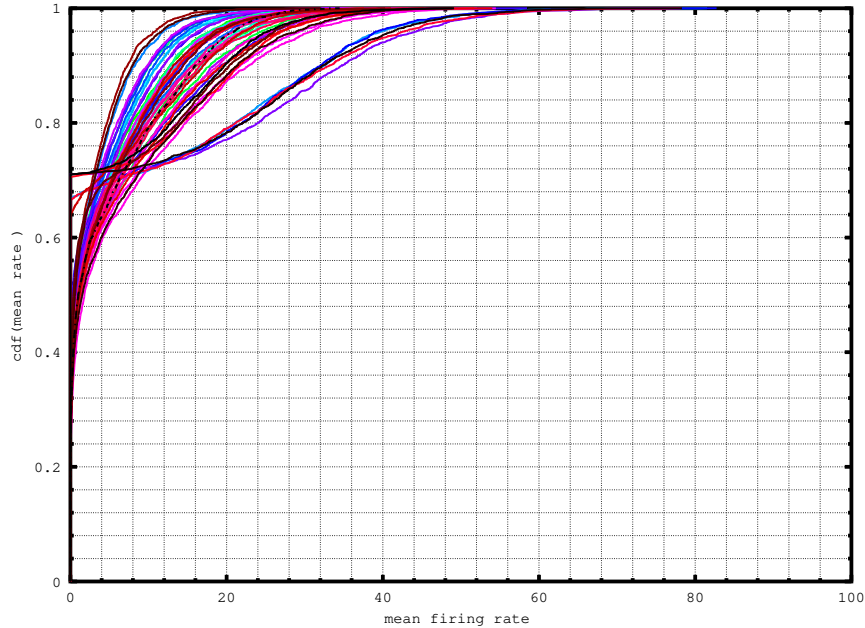


Figure 8.4: Rectified linear network. Cumulative distribution function of the units mean firing rate for 50 different inputs. The dashed line represents the standard deviation of all mean values. The dashed black line indicates that 20% of the units never fire in this network.

the system with the subsequent image as input in time window  $I_4$  is  $444.23 \pm 19.9$ . Finally, the assignment error of the perturbed system to the unperturbed system of the subsequent image is  $306.19 \pm 19.9$ . The system seems to react more strongly to the momentary input and forgets the previous input.

Just from these observations, it would seem that adding inhibition and rectification does not provide any useful additional attributes to the network, as separability becomes worse, amplification of the system is similarly difficult to control (or is not particularly robust against perturbations of the synapse weights within the network) and similarly the memory of previous inputs decays within an order of magnitude of the time required to reach an attractor state. One possible advantage of the system is that the firing distributions (fig. 8.4) of individual neurons is much more similar than in the case of the excitatory linear network (fig. 8.3).

### 8.5.3 Root Network

It is non-trivial to keep a non-learning, strongly exciting linear network within a stable firing regime when synaptic weights might be randomly perturbed. Here the adaptation exponent is  $\Omega_i = 0.5$  for all excitatory units. This can reflect factors which might reduce the effect of excitation against inhibition, such as firing adaptation, or stronger temporal decay of the EPSP vs the IPSP function. Intuitively, it can be expected that this will make

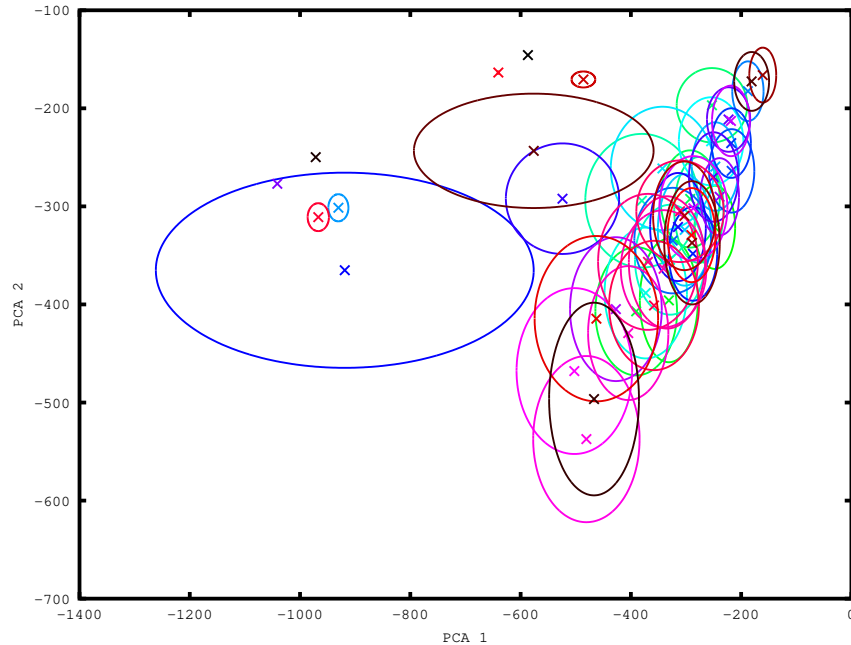


Figure 8.5: Rectified linear network.  $\times$ 's show mean values of the neuron firing rates, given 50 different inputs, projected onto the first two principle components of all responses to all inputs. Ellipses provide the standard deviation for each input in both directions. It can be observed that different inputs have very different spreads of their responses.

finding parameter regimes with bounded and stable firing behavior more simple as increased excitation leads to disproportionately larger inhibition. Again, inhibition is strong ( $A_{i2} = 20$  for excitatory units and  $A_{i1} = 20$  for inhibitory units), as well as excitation, though not as strong as inhibition ( $A_{i1} = 10$  for excitatory units and  $A_{i0} = 10$  for inhibitory units).

As with the previous networks, we let the networks run and observe the mean firing in an estimated equilibrium. The distribution of the means and standard deviations of the first two PCA components of the network for different inputs can be seen in fig. 8.12. In comparison to the linear rectified network individual separability has improved. However, there is a phenomenon which was not as prominent in the previous networks: While separability is better than for the linear rectified network, the root network has groups of inputs which seem to be drawn towards very similar attractors (in regards to mean and standard deviation).

It is easier to establish a stable amplification of weak input synapses within the root network, that mean there are fewer input which lead to a divergent firing behavior. The stronger amplification can be seen in the higher average firing rates observed in the CDFs in fig. 8.13. Also, it can be noted that the neuron firing rate CDFs are more stereotyped than for the root network than for the other networks, i.e. different inputs to the network result in a very similar shape of the distribution.

As with the rectified linear unit network, we visualize the first three principle components of typical attractors: Among the tested inputs, no explicit point attractors were found. As

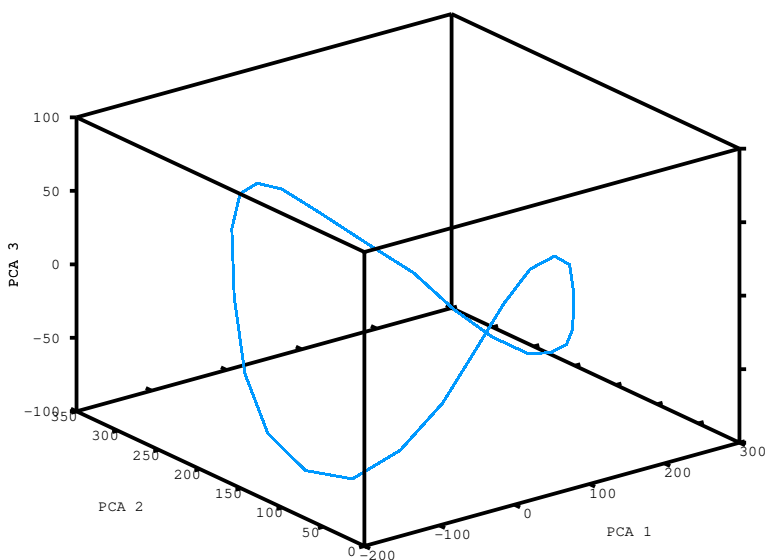


Figure 8.6: Rectified linear network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

with the LRU networks there are saddle-shaped attractors (fig. 8.14), which can become less regular (fig. 8.15) and even less regular (fig. 8.16). Two attractors, or network dynamics, can be observed which were not observed for the LRU network. Many of the attractors have the form of a torus shaped cage (fig. 8.17). It should be noted that the stability of the attractors is much less certain than in the previous cases. Some saddle-shaped attractors suddenly shift to another saddle attractor after thousands of simulation steps without any additional external driving (fig. 8.18).

As in the previous section we investigate the periodicity of the more regular attractors. For the saddle attractor from fig. 8.14 we find the self distance based periodicity in fig. 8.19. As the saddle is not totally regular it takes many time steps until a fully complete cycle is achieved, i.e. the total period is large in comparison to intermediate periods where the attractor only travels along a close by path. Interestingly, the torus attractor from fig. 8.17 is more regular and has a shorter period according to fig. 8.20.

Again as with the RLU network, we can make a comparison of assignment errors of the signals to themselves as well as during the input perturbations. Self-similarity of the signal is measured taking the reassignment error of 200 points within the time windows  $I_1$  and  $I_4 = [3500, 4000]$  with the same input. The reassignment error is  $3.56e3 \pm 1.43e2$  for the first 49 training images of the MNIST data set. The reassignment error of each signal to the attractor of the subsequent input image within time window  $I_4$  for both inputs is  $1.42e4 \pm 5.74e2$ . The

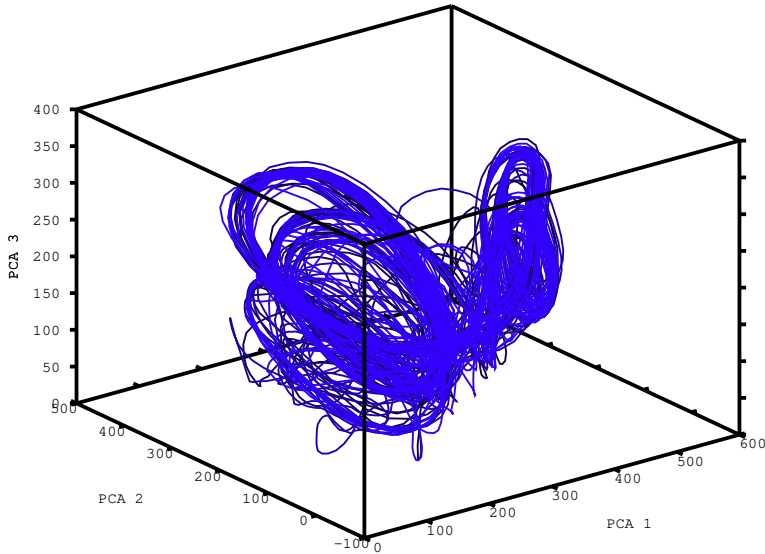


Figure 8.7: Rectified linear network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

reassignment error of the signal while perturbing the system with the subsequent image as input in time window  $I_4$  is  $3.68e3 \pm 1.44e2$ . Finally, the reassignment error of the perturbed system to the unperturbed system of the subsequent image is  $1.41e4 \pm 5.76e2$ . This behavior is markedly different from the RLU network's behavior. Even when provided with a totally different input image, once the network has been initiated, the attractor is perturbed little in regards to reassignment error. The same also holds true when the input is turned off completely during the perturbation window.

#### 8.5.4 Learning Root Network

The rectified root excitation (RRE) network is able to amplify and store states based on a small internal input for a much longer time than the RLU network. We investigate how the network behavior changes when applying a learning rule as in eq. 8.5 & 8.7. The previous RRE network was used and a redistribution of the individual compartment synaptic weight distributions was permitted based on eq. 8.5 & 8.7. The previous RRE network had a negative firing threshold (which could be considered as background noise), and for the learning we also ran a RRE network with a zero firing threshold. During learning, a constant input taken from the MNIST training set was presented to the network for 400 iterations. After 400 iterations the next MNIST training image was taken as input. One cycle consisted



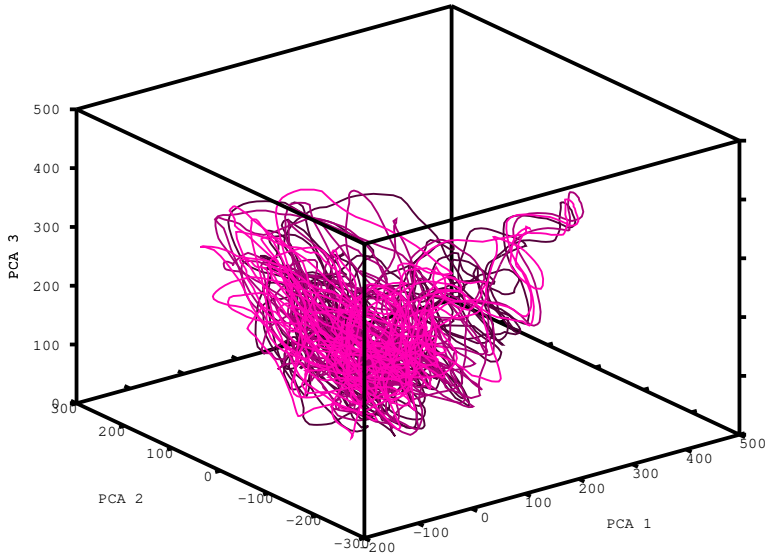


Figure 8.8: Rectified linear network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

of 20 such switches after which the network activity was reset to zero and again a new set of training images was taken for the next cycle.

Mean distributions after different numbers of cycles of learning can be seen in fig. 8.21. After 1016 cycles of learning ( $\approx 8.13e6$  iterations) the main difference between the firing distributions is visible in the mean firing rate distributions over all inputs. After learning, the fraction of units which never activate decreases (visible in the dashed line in fig. 8.21), reflecting a greater integration of all units into the network. Otherwise, the shape of the cdfs for individual inputs seems to remain very similar.

The development of the neuron mean firing rates for each input projected onto the first two PCA components (see fig. 8.22) lacks an obvious interpretation. Separability of the inputs seems to remain similar over time.

We consider firing behavior for an RRE network with firing rate threshold at 0. CDFs are shown in fig. 8.23 and for the PCA projected firing rate means in fig. 8.24.

The RRE network with a firing rate threshold of 0 starts off with a much greater overlap between inputs, in particular, there are many neurons with very similar firing patterns. However, the learning method chosen here (eq. 8.5 & 8.7) seems to be able to let the resulting network attractors separate to a substantial degree. As for the network with a negative firing threshold, the network with zero firing threshold has more units which participate in network attractors after the learning.

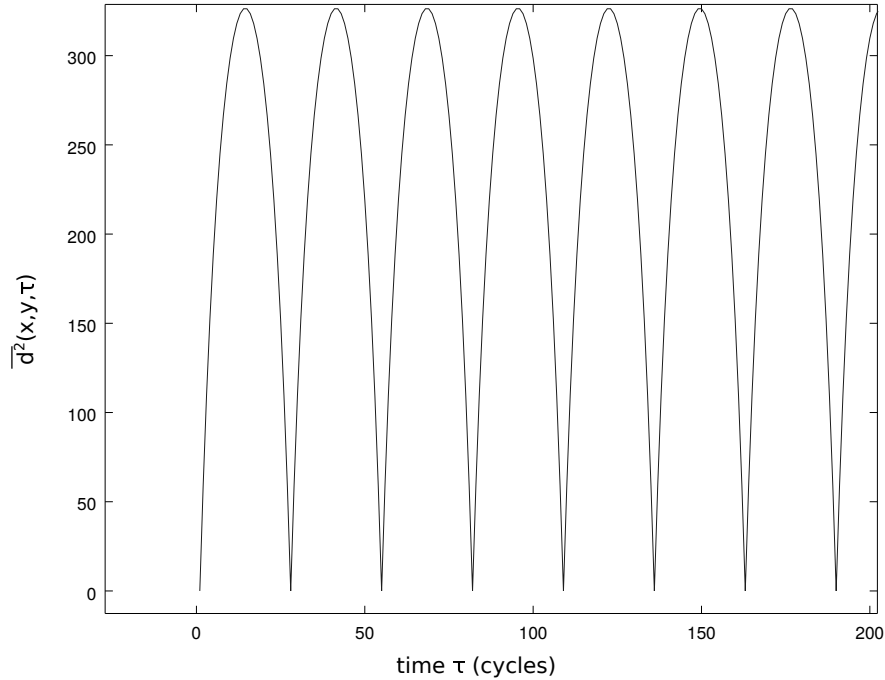


Figure 8.9: Rectified linear network. The average root square distance of the attractor from fig. 8.9 to itself over time.

## 8.6 Discussion

The presented neuron networks offer some interesting insights into network dynamics which can arise from even a mathematically very simple neuron model. The model fulfills the initially listed constraints of: input sparsity, strong recurrent excitation and inhibition, and local connectivity (here even without the need for long range inhibition). The contrasting attractor behavior of the linear rectified network and the rectified root excitation network is interesting. The former has an attractor state which decays when the input configuration is changed. The second network's attractor remains close to an initiated attractor even when a different input is provided to the system. Given the weak input strength, it is even difficult to reset the attractor of the network, even if desired. The only variables which were altered between both networks were the amplification of incoming excitation and inhibition as well as the firing amplification exponent. This implies that there are intermediate parameter configurations with different degrees of stability for the network's attractor state. Different networks could be able to change more stable attractors when the networks are more strongly connected. Connecting different networks with different degrees of attractor stability could provide a system with different degrees or hierarchies of working memory in the form of the attractor stability in different subnetworks. In this view, mammalian brains with 30+ regions become more complex to untangle in regards to interactions of brain regions and interareal connectivity could have a more considerable impact. No synaptic learning is required for these observations. Currently, one tested synaptic learning rule allows for the integration

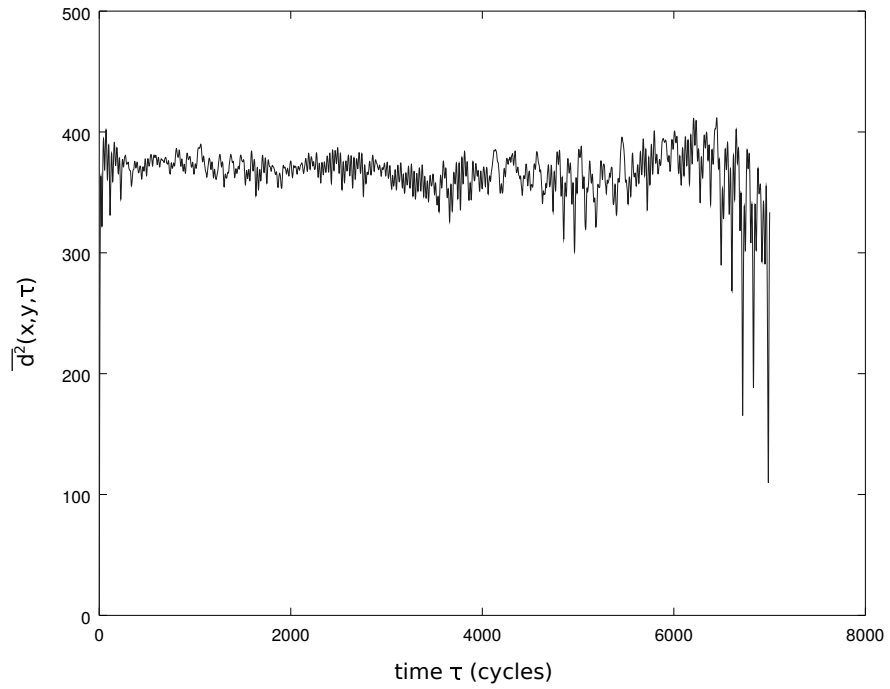


Figure 8.10: Rectified linear network. The average root square distance of the attractor from fig. 8.10 to itself over time.

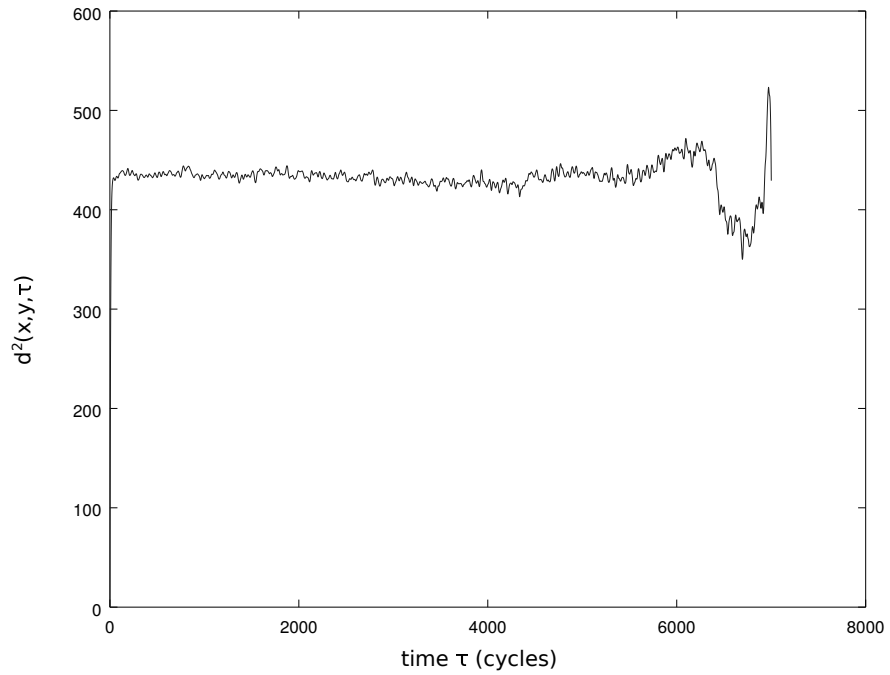


Figure 8.11: Rectified linear network. The average root square distance of the attractor from fig. 8.11 to itself over time.

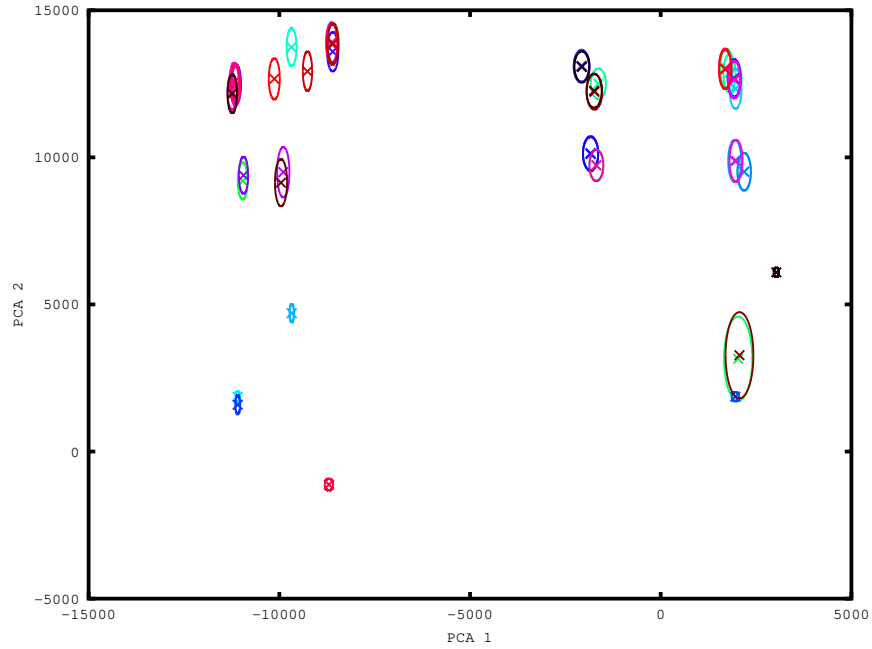


Figure 8.12: Rectified root network.  $\times$ 's show mean values of the neuron firing rates for the root network, given 50 different inputs, projected onto the first two principle components of all responses to all inputs. Ellipses provide the standard deviation for each input in both directions. It can be observed that different inputs have very different spreads of their responses.

of more units into the network attractor. Interestingly, in the current model the effect of background noise in the network (implicitly contained in the negative firing threshold of some of the networks) is not obvious, and it is not obvious if such background noise is advantageous or disadvantageous for the network dynamics.

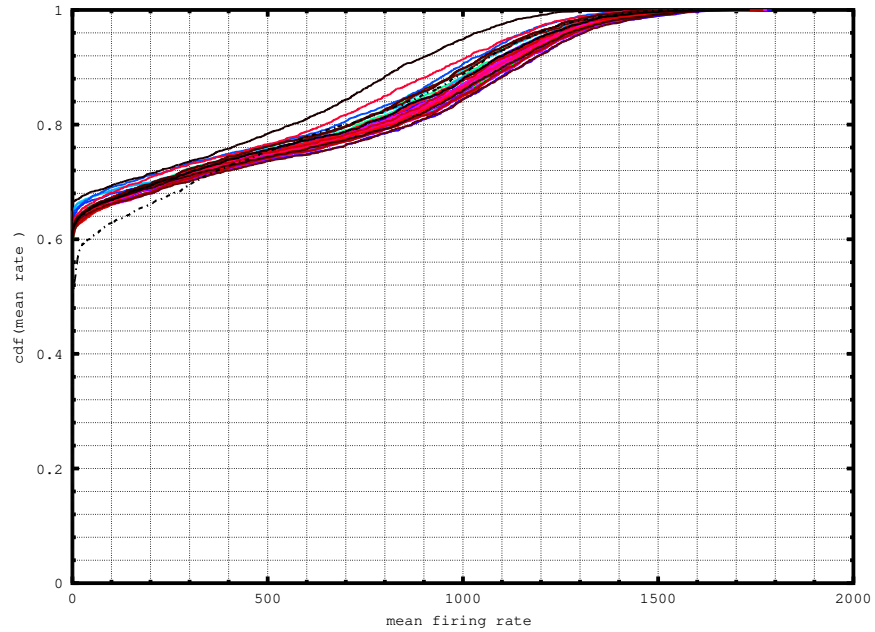


Figure 8.13: Rectified root network. Cumulative distribution function of the neurons mean firing rate for 50 different inputs

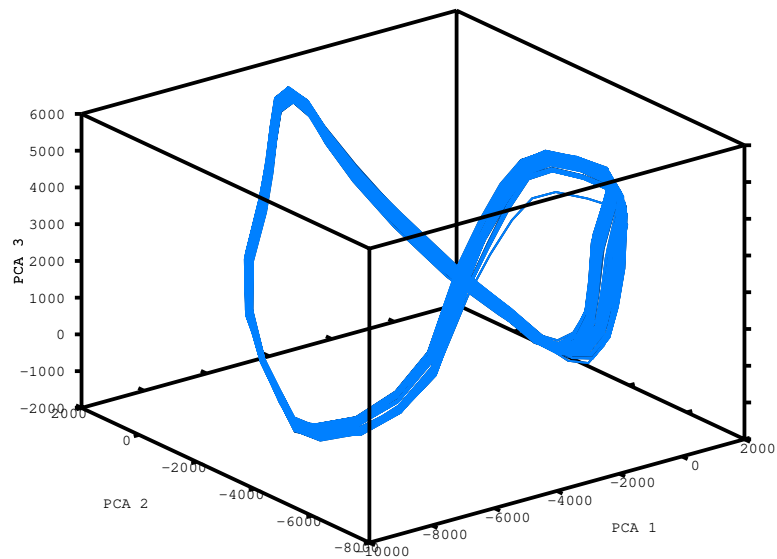


Figure 8.14: Rectified root network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

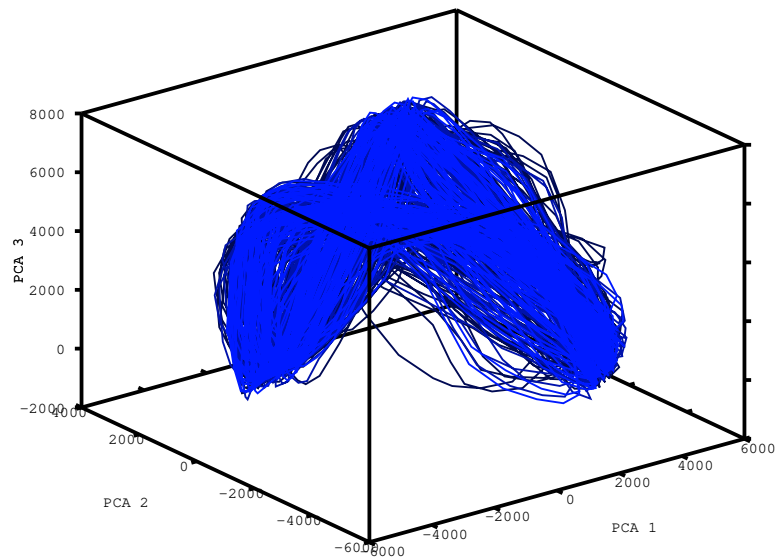


Figure 8.15: Rectified root network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

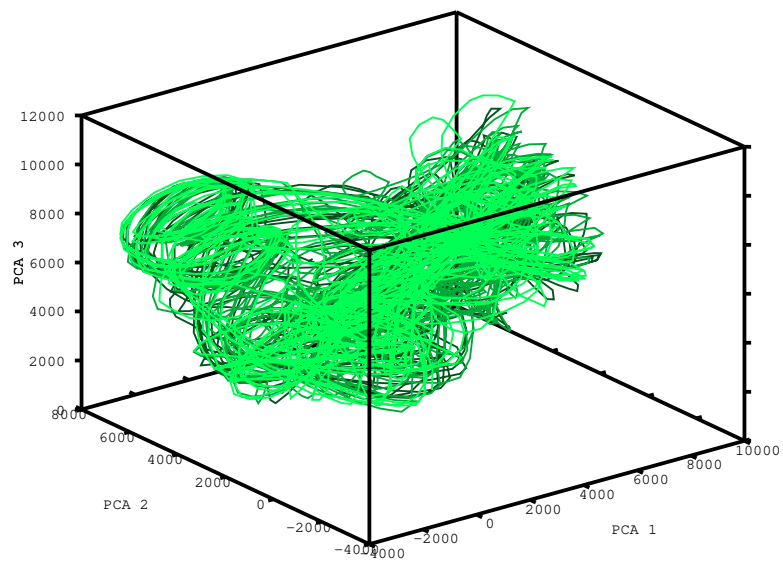


Figure 8.16: Rectified root network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

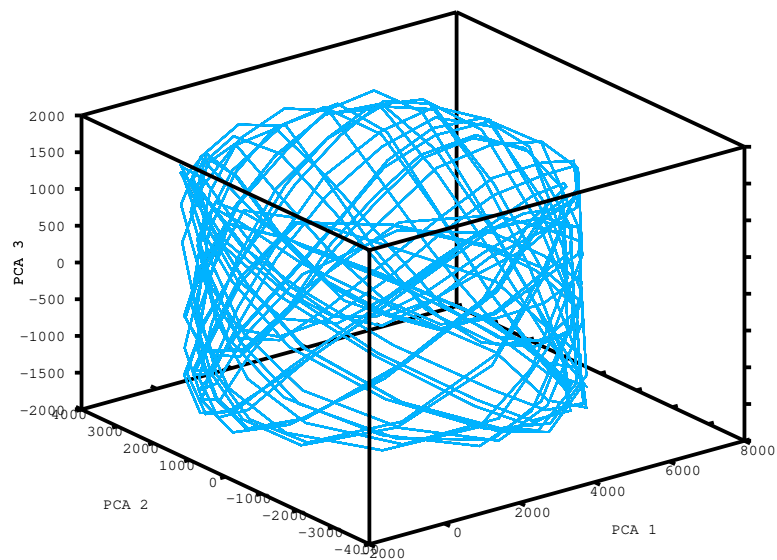


Figure 8.17: Rectified root network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.



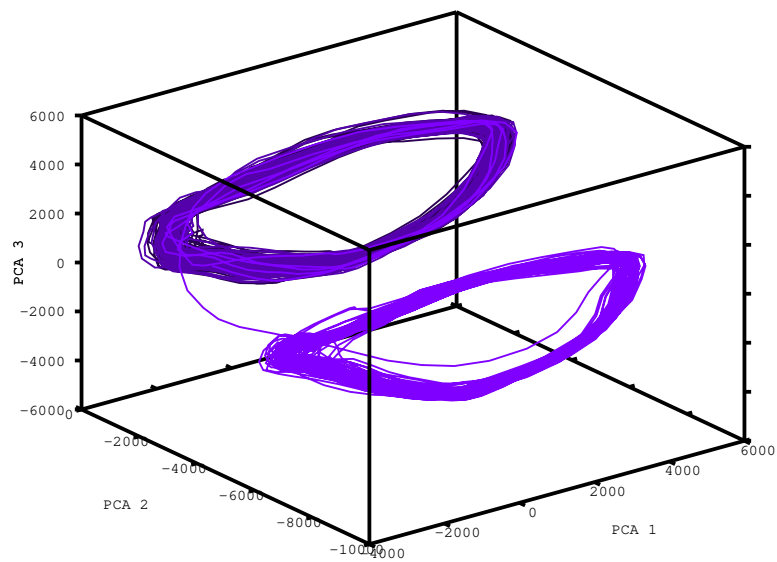


Figure 8.18: Rectified root network. Equilibrium firing response to a specific constant input. Depicted are 7000 timesteps of network projected onto its first three principle components. The first 3000 timesteps of the simulation are not shown. Early time steps are represented in darker shades than late time steps.

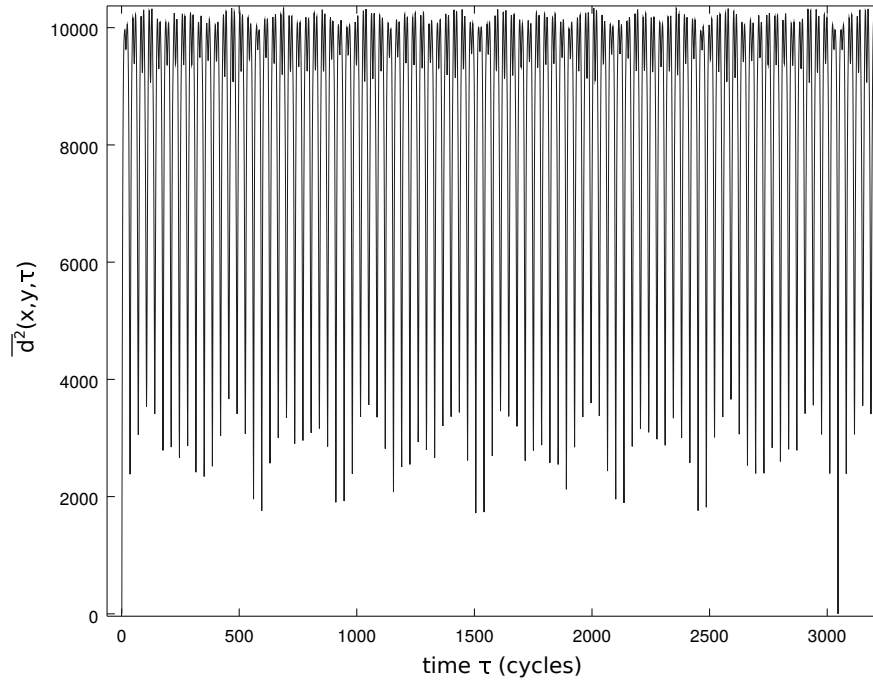


Figure 8.19: Rectified root network. The average root square distance of the attractor from fig. 8.14 to itself over time.

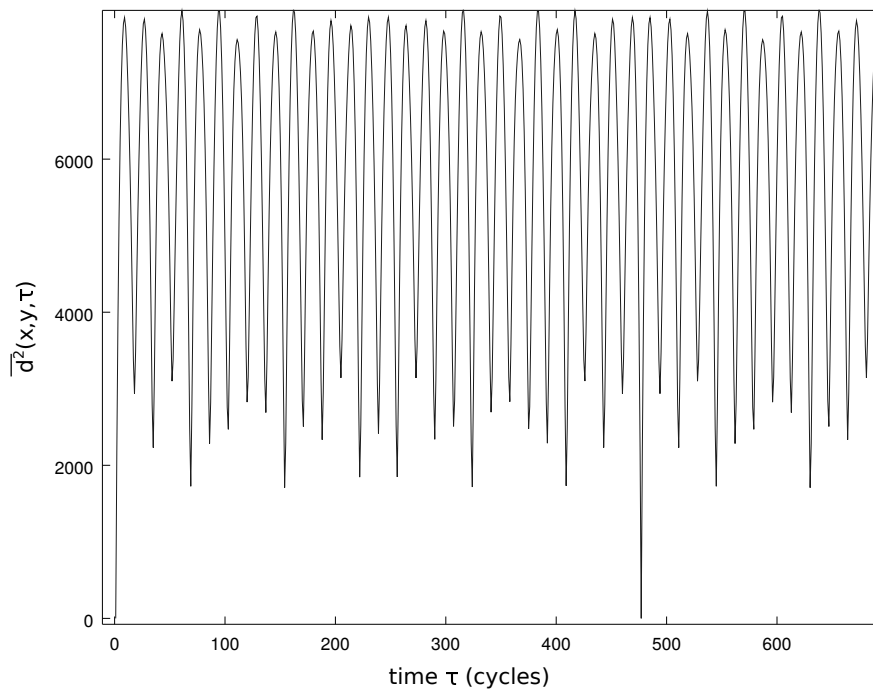


Figure 8.20: Rectified root network. The average root square distance of the attractor from fig. 8.17 to itself over time.

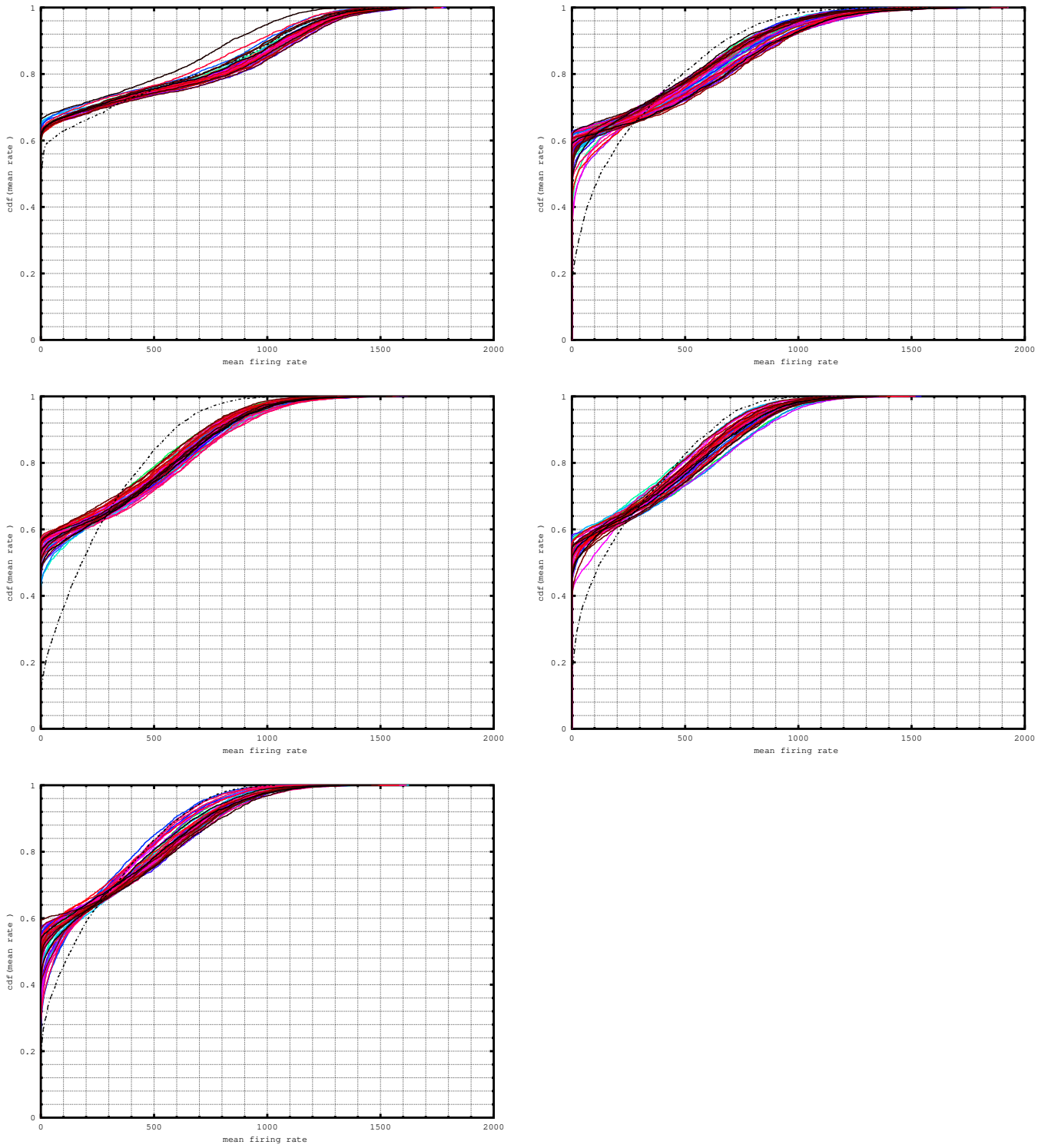


Figure 8.21: Temporal development of the cdfs of the RRE network (threshold at  $-5$ ) for 50 different input images. Time development from right to left and top to bottom at cycles: 0, 1016, 2573, 4138, 6233. The black dashed line in each plot shows the distribution of the neurons mean firing over all 50 inputs.

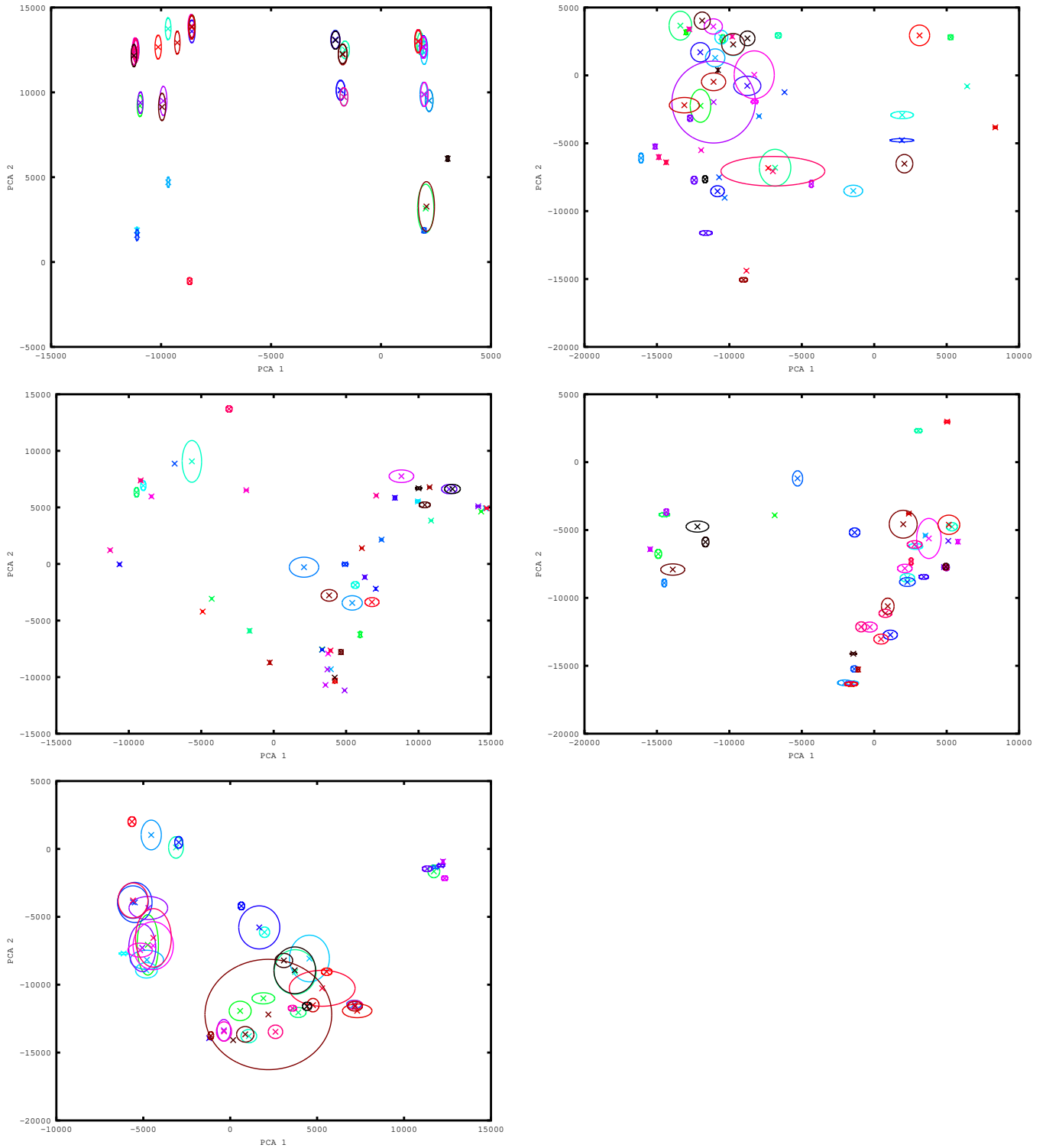


Figure 8.22: Time development of the network means projected onto the RRE network's (threshold at  $-5$  first two PCA components for 50 different input images. Time development from right to left and top to bottom at cycles: 0, 1016, 2573, 4138, 6233

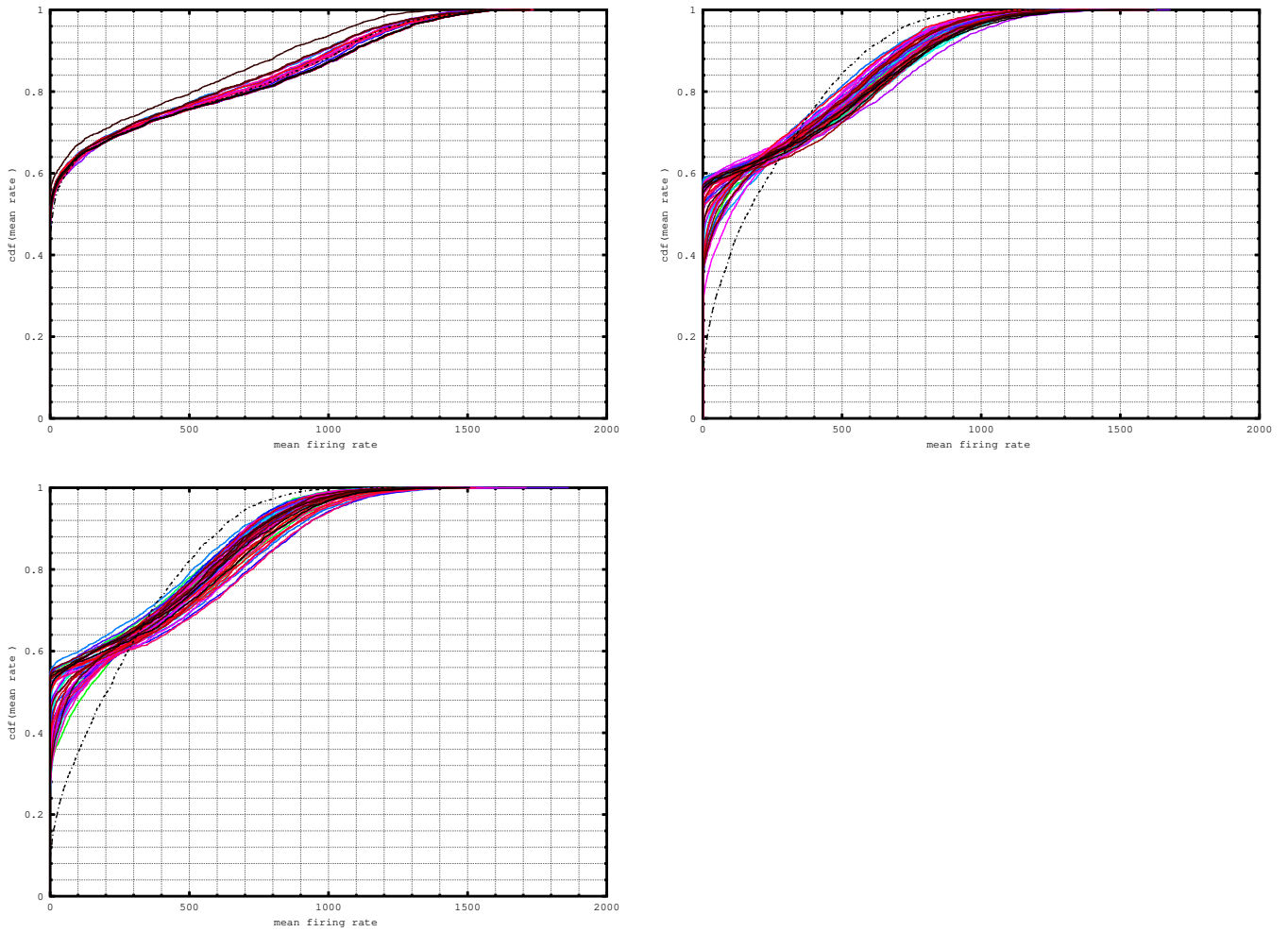


Figure 8.23: Temporal development of the cdfs of the RRE network (threshold at 0) for 50 different input images. Time development from right to left and top to bottom at cycles: 0, 2112, 4436. The black dashed line in each plot shows the distribution of the neurons mean firing over all 50 inputs.

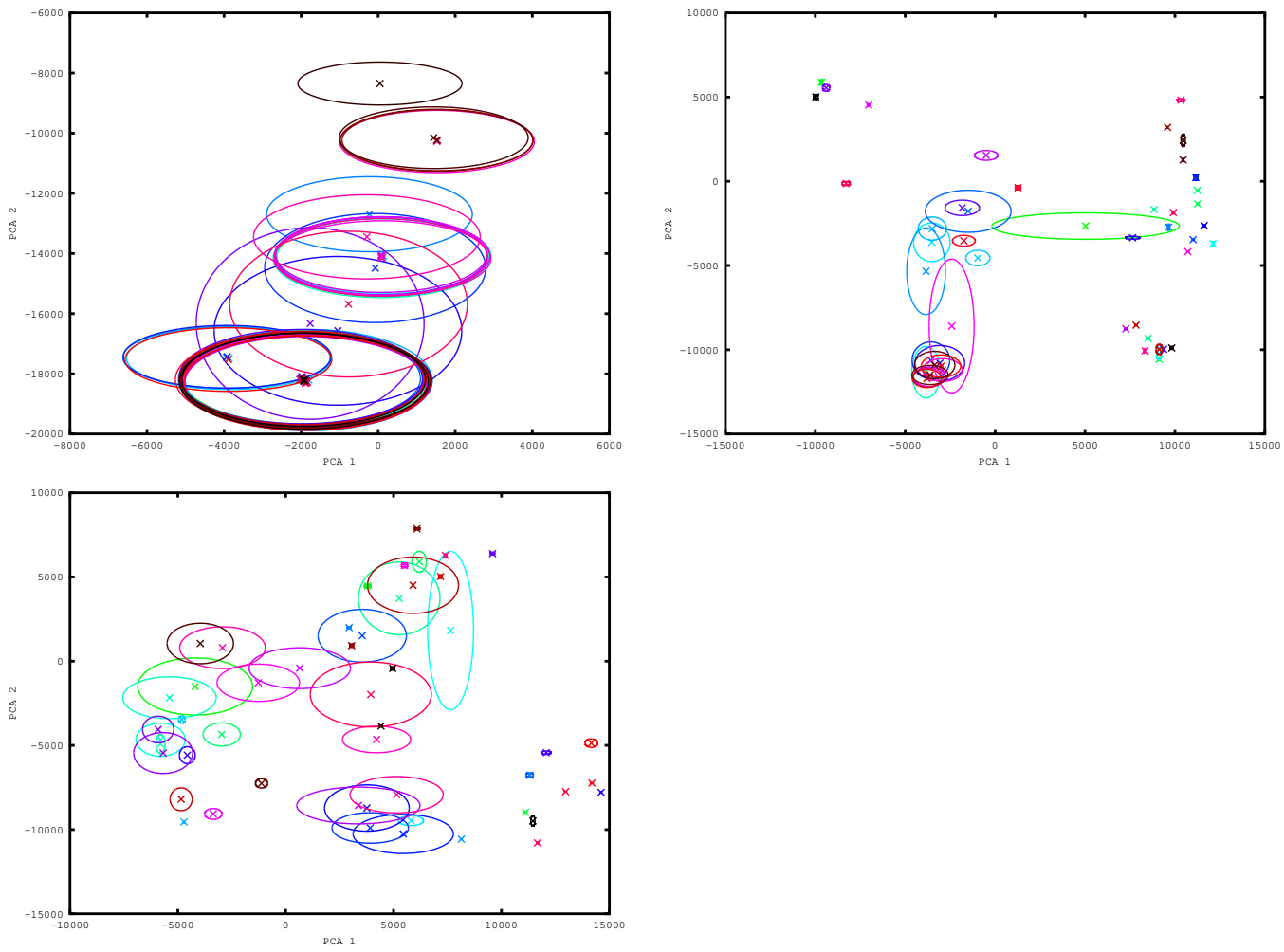


Figure 8.24: Time development of the network means projected onto the RRE network's (threshold at 0 first two PCA components for 50 different input images. Time development from right to left and top to bottom at cycles: 0, 2112, 4436

# Variables

$A$  input synapse amplification for part of the neuron. 85, 88

$H$  Heaviside function. 85

$J$  Jacobian of matrix. 87

$\mathbf{k}$   $k^{th}$  department related variable. 85, 86

$\max$  maximum function. 84

$R$  ramp function. 84

$r$  Hebbian-like prefactor for synaptic learning. 85

$\text{sgn}$  sign function. 85

$t$  time step of measurement. 84–88

$w$  synaptic input weight. 84–86, 88

$x$  neuron firing rate. 84–86

$y$  pca network firing rate. 86

$\alpha$  learning rate for synapses. 85

$\beta$  firing rate exponent for synapse learning. 85

$\gamma$  firing rate exponent for amplitude learning. 85

$\delta$  time constant for exponential moving average. 85

$\zeta$  firing scaling constant. 84

$\Theta$  target long term firing average. 85

$\theta$  unit firing threshold/background noise. 84

$\eta$  smoothing factor for firing updates. 84

$\tau$  time constant for exponential moving average. 85

$\phi$  range around target average for inhibitory learning. 85

$\Omega$  firing adaption exponent. 84, 88



# Glossary

**EPSP** excitatory post-synaptic potential. 84

**IPSP** inhibitory post-synaptic potential. 84

**PCA** principle component analysis. 86

# Chapter 9

## Discussion

This work contains measurements of synaptic densities in the mouse neocortex as well as a varied set of tools for general purpose image segmentation applied to neocortical tissue recorded under the electron microscope. A very similar set of tools can be and was applied to an audio source separation task. Finally, a very simple network model which functions under the constraint of input sparsity, and strong and local recurrent connectivity. The question of how these items may relate, or how insights from one of these projects might relate to the others is a basis for discussion.

One of the key observations from the synapse and neuron density measurements is that neurons in the mouse neocortex receive thousands of input synapses each. Previous synapse and neuron density measurements from the literature (abridged in appendix A) align with these findings and it seems that cortical neurons typically receive in the range of 1'000 to 20'000 synapses. Considering typical magnitudes of excitatory and inhibitory post-synaptic potentials and considering one of the simplest models of input-output amplification found in linear rectified units, we find that cortical units can easily amplify inputs by a factor of 100. Results from chapter 8 indicate that inhibition which grows non-linearly faster than excitation provides a robust means of providing an amplifying network with a stable equilibrium. However, it is not clear how to reset a very stable network attractor once initiated. In this regards, it would be interesting to investigate if strongly coupling multiple such networks gives better control of the network attractor. Such a coupling of networks could reflect the relationship between different the layers of the neocortex which are more strongly coupled and might help distill the concept of a canonical microcircuit. An unaddressed observation from chapter 2 are why the ratio of excitatory and inhibitory neurons as well as synapses are so similar over so many cortical areas, and species and if this is more based on functionality, resource optimization, or how the cortex is formed during development.

While acquiring and evaluating the data to measure synapse densities it was immediately obvious that this was a laborious task, and quickly the question arose, if there wasn't some way to automate the process. The previous decade has seen great strides in the advance of automatic recognition systems, commonly using neural networks. It is partially because of this large scale focus on neural networks, that an alternative approach was chosen in this

work. Not for the sake of competing or comparing with these methods, but more with the intention of gaining additional insights into general recognition problems by attacking the problem from a slightly different angle. This approach can essentially be separated into three modular components. The first component addresses the general task of classification (addressed in chapter 3). Given a set of labeled data points, can the labeling pattern be expanded to new data points? To address this problem the random forest algorithm was chosen for its ease of implementation as well as its intuitive simplicity. Sadly this classifier learns less to predict a labeling pattern, but more to recognize clusters of similarly labeled data points in a feature space. The strength of the algorithm lies within the choice of the feature dimensions. This is not fundamentally different from what multi-layered perceptron networks do. The second component provides additional features by spatially linking data points as seen in the spatial filters in chapter 4. This can only be done, when we realize that the data points are not necessarily independent of one another. The third component requires the interaction of multiple classifiers to identify more general spatial relationships between data points (as described in chapter 5). Fitting these three components together allow for the improvement of classifications based purely on the random forest algorithm as can be seen in chapter 6 and chapter 7.

It would be interesting to investigate if a combination of some of the expansions for a classification proposed in this thesis in chapter 3, chapter 4 or chapter 5 would be able to improve the currently best performing machine learning methods found in neural networks, either in increasing performance or in regards to reducing training time for networks or reducing the training data required.

A more bold hypothesis links the three previously mentioned components to biological networks using the following questions: Can individual neurons be regarded as a spatial linkage between data points as with the second component? Can the recurrent network connectivity with thresholded activity function as a classification system somewhat akin to the random forest classifier? Do interacting brain regions provide a method of reclassification as seen in the third component?

At the very least chapter 4 provides a scheme to create regional super pixels at various scales as well as an adaptive scale-invariant (and partially rotation-invariant) shape descriptor. chapter 3 provides useful basic comparisons of different splitting methods and expansions of the feature space for small data sets using the random forest algorithm.

The source separation task from chapter 7 uses a very similar classification pipeline as used for electron microscopy image segmentation in chapter 6. This shows that at least visual and audio classification tasks aren't fundamentally different. Could the same be true for further sensory modalities? Could motor output, or output from association cortex be handled in similar manner? What happens when 50+ such classification tasks are solved in parallel and outputs are constantly being exchanged between classifiers?

Much effort is spent going into great depth when researching classification methods, developing microscopy tools or describing new neuron models. This effort is not in vain as

research in all these aspects of neuroscience make great strides. However, it can be useful to every now and then take a step back and look at the elements from a different perspective and obtain a more holistic point of view.

It's true that the current artificial neural networks used in machine learning may not be biologically plausible, but are we able to discern which aspects of the artificial networks are indispensable for general classification, which we expect to be present, in some form, in biological neural networks? Can we state, why we expect this? Biological systems are good at solving many learning tasks. What is the importance of the interplay of sensory modalities. How are solving skills impaired when the areas which process additional sensory or motor information are impaired? For example, how far would visual scene analysis be hindered in the lack of a frontal eye field region? Or if other regions aren't accessible. If a model of a neuron network can be locked into an attractor state, could this have any potential uses? Can similar locked states be observed to some degree in biological networks?

It seems most research leads to more unanswered questions than answered ones, and maybe a yet unanswered question which challenges our way of thinking is as valuable as many of the questions we have already answered.

# Appendix A

## Appendix A - Density Reviews

### A.1 Review of Synapse and Neuron Density Papers

#### Tabular Review

| location                | $\rho_S[10^9 mm^{-3}]$ | $\rho_N[10^4 mm^{-3}]$ | $S/N[10^3]$       | reference |
|-------------------------|------------------------|------------------------|-------------------|-----------|
| mouse Br8               | 0.675                  | 6.7                    | 10.2              | [151]     |
| mouse Br6               | 0.705                  | 9.35                   | 7.65              | [151]     |
| mouse Br17              | 0.78                   | 11.45                  | 6.85              | [151]     |
| mouse somatosensory     | 0.7 – 1.2              | -                      | -                 | [51]      |
| mouse piriform          | -                      | 0.917 – 1.12           | -                 | [160]     |
| rat visual              | 0.65 – 0.68(0.56)      | 7.309(8.76)            | 9.15 – 9.51(6.45) | [170]     |
| rat visual              | 0.92                   | 8.69                   | 10.737            | [111]     |
| rat visual              | 0.824 – 1.03           | 6.91 – 7.63            | 10.35 – 15.32     | [180]     |
| albino rat visual       | -                      | 6.0                    | -                 | [47]      |
| rat Fr1                 | -                      | 4.85                   | -                 | [157]     |
| rat HL                  | -                      | 6.94                   | -                 | [157]     |
| rat Oc2                 | -                      | 7.69                   | -                 | [157]     |
| rat oc1M                | -                      | 5.18                   | -                 | [5]       |
| rat Par1                | -                      | 4.75                   | -                 | [5]       |
| rat Fr1                 | -                      | 3.37                   | -                 | [5]       |
| rat hypoglossal nucleus | 0.2(1.5 – 3.5)         | -                      | -                 | [125]     |
| rat S2 dorsal horn      | 0.46 – 0.79            | -                      | -                 | [107]     |
| echidna                 | 0.266 – 0.289          | 9.99 – 11.72           | -                 | [62]      |
| rabbit visual           | 0.04 – 0.673           | 3.8 – 22.8             | 0.17 – 17.71      | [175]     |
| rabbit motor            | 0.126 – 0.707          | 2.3 – 13.4             | 0.90 – 30.74      | [175]     |
| rabbit auditory         | -                      | 5.52                   | -                 | [106]     |
| chicken IMHV            | 0.495 – 0.513          | 9.05 – 10.0            | 4.95 – 5.66       | [31]      |

| location           | $\rho_S[10^9 mm^{-3}]$ | $\rho_N[10^4 mm^{-3}]$ | $S/N[10^3]$ | reference |
|--------------------|------------------------|------------------------|-------------|-----------|
| cat visual         | -                      | 4.80 – 5.0             | -           | [9]       |
| cat Br17           | -                      | 4.8 – 5.0              | -           | [6]       |
| cat Br18           | -                      | 2.7                    | -           | [6]       |
| cat PMLS           | -                      | 2.74                   | -           | [6]       |
| cat motor          | 0.26 – 0.29            | 2.89 – 2.95            | 8.86 – 9.80 | [8]       |
| macaque visual     | 0.276                  | 12                     | 2.3         | [123]     |
| monkey visual      | -                      | 7.91 – 8.98            | -           | [124]     |
| monkey motor       | 0.3                    | -                      | -           | [188]     |
| monkey frontal     | 0.12                   | 14                     | 0.86        | [117]     |
| monkey frontal     | 0.13                   | 7.4                    | 1.8         | [117]     |
| monkey frontal     | 0.17                   | 7.5                    | 2.3         | [117]     |
| temporal           |                        |                        |             |           |
| monkey temporal    | 0.14                   | 7.2                    | 1.9         | [117]     |
| parietal           |                        |                        |             |           |
| monkey parietal-   | 0.18                   | 11                     | 1.6         | [117]     |
| occipital          |                        |                        |             |           |
| monkey occipital   | 0.22                   | 13                     | 1.7         | [117]     |
| monkey frontal     | -                      | 9.0                    | -           | [25]      |
| monkey dorsal      | -                      | 9.0                    | -           | [25]      |
| prefrontal         |                        |                        |             |           |
| monkey ventral     | -                      | 10.0                   | -           | [25]      |
| prefrontal         |                        |                        |             |           |
| monkey posterior   | -                      | 8.0                    | -           | [25]      |
| frontal            |                        |                        |             |           |
| monkey temporal    | -                      | 16.0                   | -           | [25]      |
| occipital          |                        |                        |             |           |
| monkey parietal    | -                      | 32.0                   | -           | [25]      |
| occipital          |                        |                        |             |           |
| monkey Br4         | -                      | 4.38                   | -           | [66]      |
| monkey Br3b        | -                      | 7.10                   | -           | [66]      |
| monkey Br1-2       | -                      | 6.42                   | -           | [66]      |
| monkey Br5         | -                      | 6.07                   | -           | [66]      |
| monkey Br17        | -                      | 11.98                  | -           | [66]      |
| human temporal     | 1.508 – 1.626          | -                      | -           | [34]      |
| neocortex          |                        |                        |             |           |
| human Br21         | 0.717 – 1.061          | 2.59 – 2.76            | -           | [2]       |
| human frontal      | -                      | 3.59 – 3.67            | -           | [126]     |
| human temporal     | -                      | 5.10 – 5.98            | -           | [126]     |
| human parietal     | -                      | 4.52 – 4.73            | -           | [126]     |
| human occipital    | -                      | 6.69 – 7.09            | -           | [126]     |
| dolphin, whale vi- | 0.79                   | -                      | 24          | [53]      |
| sual               |                        |                        |             |           |

| location                      | $\rho_S[10^9 mm^{-3}]$ | $\rho_N[10^4 mm^{-3}]$ | $S/N[10^3]$ | reference |
|-------------------------------|------------------------|------------------------|-------------|-----------|
| porpoise visual               | -                      | 1.3(2.57)              | -           | [49]      |
| minke visual                  | -                      | 0.937                  | -           | [41]      |
| minke auditory                | -                      | 0.845                  | -           | [41]      |
| dolphin al                    | -                      | 2.85                   | -           | [50]      |
| dolphin pl                    | -                      | 4.25                   | -           | [50]      |
| dolphin calc                  | -                      | 3.48                   | -           | [50]      |
| rat cervical gan-<br>glion    | -                      | 3.67                   | -           | [139]     |
| capybara cervical<br>ganglion | -                      | 0.705                  | -           | [139]     |
| horse cervical<br>ganglion    | -                      | 0.825                  | -           | [139]     |
| mouse visual                  | 0.66                   | 8.73 – 10.74           | 7           | [29]      |
| mouse motor                   | 0.85                   | 4.62 – 6.44            | 13          | [29]      |
| monkey visual                 | 0.62                   | 10.42 – 12.67          | 5.6         | [29]      |
| monkey motor                  | 0.96                   | 1.61 – 2.07            | 60          | [29]      |
| rat visual                    | -                      | 4.68 – 5.39            | -           | [29]      |
| galago visual                 | -                      | 9.57 – 10.67           | -           | [29]      |
| galago motor                  | -                      | 3.35 – 4.93            | -           | [29]      |
| rabbit visual                 | -                      | 4.13 – 4.70            | -           | [29]      |
| rabbit motor                  | -                      | 2.28 – 2.76            | -           | [29]      |
| cat visual                    | -                      | 3.40 – 4.94            | -           | [29]      |
| cat motor                     | -                      | 2.54                   | -           | [29]      |
| pig visual                    | -                      | 2.15                   | -           | [29]      |
| pig motor                     | -                      | 1.75                   | -           | [29]      |
| human visual                  | -                      | 2.79 – 5.19            | -           | [29]      |
| human motor                   | -                      | 0.99                   | -           | [29]      |
| mouse                         | -                      | 15                     | -           | [167]     |
| rat                           | -                      | 8 – 11                 | -           | [167]     |
| tarsius                       | -                      | 18                     | -           | [167]     |
| guinea pig                    | -                      | 5.5                    | -           | [167]     |
| hapale                        | -                      | 8                      | -           | [167]     |
| rabbit                        | -                      | 4.7                    | -           | [167]     |
| cat                           | -                      | 3.5                    | -           | [167]     |
| dog                           | -                      | 2.7                    | -           | [167]     |
| cecopithecus                  | -                      | 7.5                    | -           | [167]     |
| macacca                       | -                      | 1.8                    | -           | [167]     |
| beef                          | -                      | 7                      | -           | [167]     |
| human                         | -                      | 1 – 3                  | -           | [167]     |
| elephant                      | -                      | 0.9                    | -           | [167]     |
| whale                         | -                      | 0.8                    | -           | [167]     |

## Paper Comments

**Synapse Measurements** [107]: A questionable paper on the synaptic densities in the layers of rat dorsal horn (spinal cord related).

[188] : Synapse densities in monkey motor cortex during development. Contains individual layers. Shaft to non-shaft synapse ratios.

[171] : Synapse counts in tree shrew visual cortex during development (contains symmetric/asymmetric synapse ratios as well)

[125] : synapse counts in the rat hypoglossal nucleus during development.

[34] : Synapse densities in human using different methods. Shrinkage? Layer densities are compared, symmetric to asymmetric synapse ratios are listed.

[51] : A paper on synapse densities in mouse somatosensory cortex. Ratios of symmetric to asymmetric synapses are listed. Dendrite properties are mentioned as well.

[2] : Synapse density measurements in the human neocortex. Area 21... Ratio of synapse types given. Glia to neuron ratios are present in the paper as well.

**Neuron Measurements** [167] : A classical paper on neuron density counts in various species.

[124] : Another study on neuron density in monkey visual cortex. Number during development under the influence of visual deprivation.

[9] : First paper in a large set of cat visual cortex investigations. Here the number of neurons in the different layers of cortex. Differentiation between monocular and binocular regions. Thickness of cortex presented as well.

[6] : Neuron densities in various areas of the cat visual cortex. Different layers as well as cortical thickness are investigated. Most of the difference in densities could be explained by the difference in cortical thickness of the different areas.

[50] : Neuron and glia density in dolphin lateral gyrus. Individual layers were investigated as well. The paper contains useful references to other neuron density papers.

[66] : In depth investigation of ratio of GABA and Non-GABA neurons in monkey cortex. Different layers and areas investigated. Different ratio in visual cortex...

[47] : Study of glia and neurons in the visual cortex of the albino rat.

[49] : Study on whale neuron densities in visual cortex, as well as the ratio of GABA and non-GABAergic cells. Data is given on a layer basis. Absolute density value is quite low. Ratios between layers seem reasonable. (shrinkage accounted)

[5] : Neuron type ratios in different layers in rat cortex (oc1M,Par1,Fr1).

[106] : Neuron density measurements in rabbit auditory cortex. Also contains ratio of a subclass of GABA (parvalbumin) vs all neurons throughout the layers of cortex.

[157] : A good set of measurements of neuron densities in different areas in the rat. Challenges Rockel et al. uniformity statement. Also has measurement of number of neurons under a surface of cortex. Contains individual layer densities as well.

[126] : Neuron densities in the human cortex (quite a few brains were used for the study).

[161] : Contains a short review on different neuron densities in rat, mouse and monkey. Also contains interesting looking dendrite measurements. Definitely useful when checking



for dendrite as well as bouton measurements.

[139] : Neuron density measurements in cervical ganglion of the rat, capybara and horse.

[132] : Neuron counts in various Cetaceans. Observe decreasing neuron density per brain mass.

[69] : Neuron counts for various rodent species put together to obtain multiple scaling laws.

[67] : Neuron counts for various primate species put together to obtain multiple scaling laws.

[41] : It takes some getting used to the difference in cell density in size between cetaceans and land mammals. Cell density is commonly lower for cetaceans, however the cells are bigger and the glia to neuron ratio is much higher. Are the numbers still intelligently comparable? The paper presents neurons densities in visual and auditory cortex of the Minke whale. Sizes of cells are given as well.

[25] : Neuron density in monkey hippocampus as well as neuron to glia ratios. Visual cortex has a bizarrely high neuron density.

[68] : This paper contains number of neurons under an area of cortex for many species. Disagrees with [143]. Doesn't agree well with [123] for Macaque monkey.

[146] : Neuron counts for various insectivore species put together to obtain multiple scaling laws.

[84] : Neuron density measurements (glutamergic and non-glutamergic) in various hippocampus locations in the mouse.

**Synapse to Neuron Ratios** [29] : The most classic paper relating synapse-neuron ratios for various species. Tissue shrinkage might be a more severe issue for this paper.

[175] : synapse-neuron ratios in rabbit visual and motor cortex. The paper considers the densities throughout development. The upper values seem to high, but should help estimate the true densities (the neuron values decrease monotonically towards adulthood)

[180] : synapse-neuron ratios in undernourished rats in visual cortex (with control rat values).

[123] : Another classic paper. Contains neuron, synapse and glia densities of the macaque monkey visual cortex. Very detailed information is presented.

[170] : synapse-neuron ratios in rat visual cortex (taking differential rearing into account.

[7] : The paper in which cat visual cortex is investigated. Presented are the neuron+synapse densities throughout the layers, their ratios, number of symmetric vs asymmetric synapse as well as synapse locations on the dendrite or soma.

[151] : The classic paper contains synapse-neuron ratios in the mouse cortex areas 8,6,17 . Very thorough measurements throughout the individual layers in the three areas. Contains symmetric/asymmetric synapse ratios as well. (There is more disagreement than agreement with [143])

[31] : A non-mammal synapse-neuron ratio as well as glia relations, in this case in the chicken medial hyperstriatum ventrale, during development. Amazingly the results are well within the range of values for mammalian species.

[8] : Extensive study of the cat motor cortex containing number of synapses and neurons in the laminae as well as cortical thickness. Symmetric vs asymmetric synapse ratios are given, plus location of the synapses (spines, shaft etc.).

[53] : A paper presenting Dolphin+Whale synapse and neuron densities. Has some nice neuron reconstructions and layer representations. Synapse densities of the different cortical layers are presented. Qualitative descriptions of axons and dendrites. Synapse type ratios are also investigated and compared with land mammals. They support the notion of a constant number of synapses under a surface of neocortex. Somehow the paper isn't quite that able to quote synapse to neuron ratios correctly...

[117] : The results presented here disagree somewhat with those of [123]. Reading the methods it seems well possible that a large set of synapses was ignored. However the results should have useful relations among themselves, since measurements are from the same tissue.

[111] : Study on synapse to neuron ratios in rat visual cortex. More of a testing of a certain measurement technique. (layers 2-4)

[62] : A lot of information on synapse+neuron densities of the echidna. It also contains much reviewing of these densities, as well as glia ratios for various other species. Even dendrite measures are incorporated. (Not sure how well shrinkage was taken into account within this paper)

[108] : A slightly different, yet nevertheless interesting paper. It contains a review of neuron and synapse counts in various species, many of which are invertebrates. (Even though its presentation might seem questionable).

[30] : Synapse-Neuron ratio is inversely related to neuronal density in mature neuronal cultures.

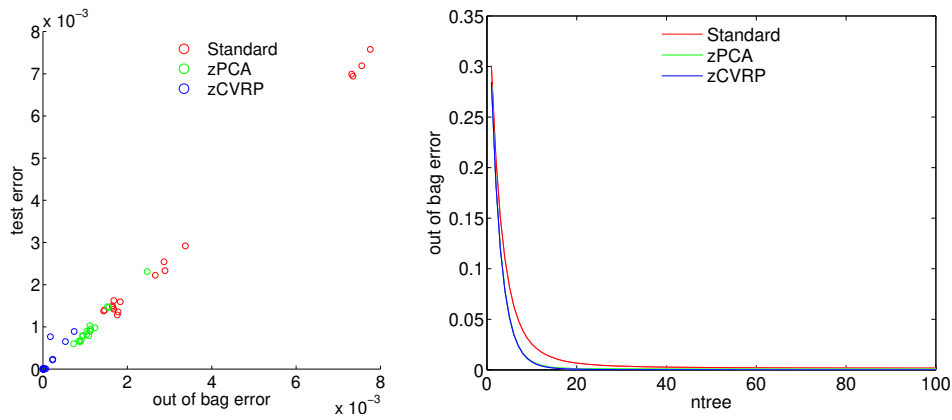
# Appendix B

## Appendix B - Random Forest Result Tables

### B.1 Random Forest Classification Results

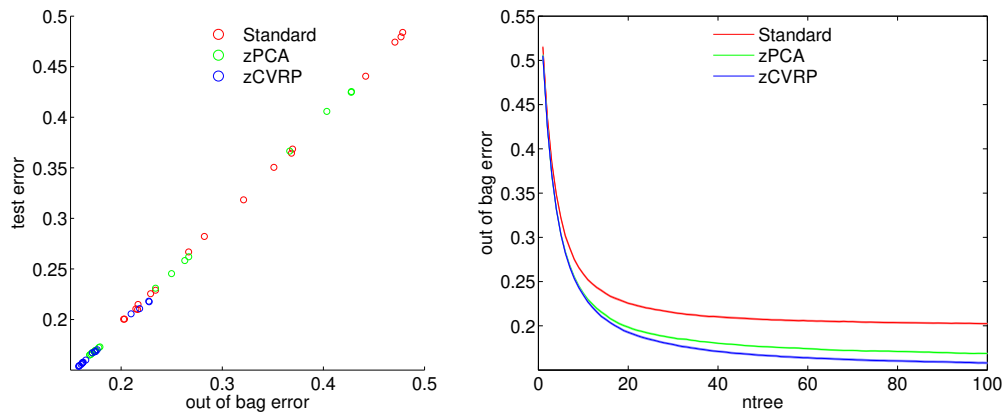
|   |                                       |                           |                           |                                       |
|---|---------------------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Banknote (1000 runs) d=4, c=2, tot=1372, n=138 |                                       |                           |                           |                                       |
| Coordinate system: Standard d=4                     |                                       |                           |                           |                                       |
|   | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 3$                       |
| $G_{gini}$  | $0.008 \pm 2.36e-4^{***}$             | $0.007 \pm 2.27e-4^{***}$ | $0.007 \pm 2.28e-4^{***}$ | $0.007 \pm 2.31e-4^{***}$             |
| $G_{inf}$   | <b><math>0.001 \pm 9.66e-5</math></b> | $0.001 \pm 9.76e-5$       | $0.001 \pm 9.86e-5$       | $0.002 \pm 1.06e-4^*$                 |
| $G_{flip}$  | $0.001 \pm 9.61e-5$                   | $0.001 \pm 9.45e-5$       | $0.001 \pm 9.77e-5$       | $0.002 \pm 1.00e-4$                   |
| $G_{rs}$  | $0.002 \pm 1.31e-4^{***}$             | $0.002 \pm 1.31e-4^{***}$ | $0.003 \pm 1.37e-4^{***}$ | $0.003 \pm 1.43e-4^{***}$             |
| Best OOB  | $0.002 \pm 1.06e-4^{**}$              |                           |                           |                                       |
| Coordinate system: zPCA d=8                         |                                       |                           |                           |                                       |
|   | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 4$                       |
| $G_{gini}$  | $0.001 \pm 8.35e-5^{***}$             | $0.001 \pm 1.02e-4^{***}$ | $0.001 \pm 1.02e-4^{***}$ | $0.002 \pm 1.26e-4^{***}$             |
| $G_{inf}$   | $0.001 \pm 7.96e-5^{**}$              | $0.001 \pm 7.83e-5^{**}$  | $0.001 \pm 8.11e-5^{**}$  | $0.001 \pm 7.36e-5^*$                 |
| $G_{flip}$  | $0.001 \pm 7.40e-5^*$                 | $0.001 \pm 7.36e-5^*$     | $0.001 \pm 7.47e-5^*$     | <b><math>0.001 \pm 6.40e-5</math></b> |
| $G_{rs}$  | $0.001 \pm 6.70e-5$                   | $0.001 \pm 6.73e-5$       | $0.001 \pm 6.52e-5$       | $0.001 \pm 8.26e-5^{***}$             |
| Best OOB  | $0.001 \pm 8.63e-5^{***}$             |                           |                           |                                       |
| Coordinate system: zCVRP d=12                       |                                       |                           |                           |                                       |
|   | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 3$           | $d_{trial} = 4$                       |
| $G_{gini}$  | $0.000 \pm 0.00e+0$                   | $0.000 \pm 3.91e-5^{***}$ | $0.001 \pm 6.56e-5^{***}$ | $0.001 \pm 7.73e-5^{***}$             |
| $G_{inf}$   | $0.000 \pm 0.00e+0$                   | $0.000 \pm 0.00e+0$       | $0.000 \pm 0.00e+0$       | $0.000 \pm 0.00e+0$                   |
| $G_{flip}$  | $0.000 \pm 7.24e-6$                   | $0.000 \pm 0.00e+0$       | $0.000 \pm 0.00e+0$       | $0.000 \pm 0.00e+0$                   |
| $G_{rs}$  | $0.000 \pm 2.17e-5$                   | $0.000 \pm 0.00e+0$       | $0.000 \pm 7.24e-6$       | $0.000 \pm 4.03e-5^{***}$             |
| Best OOB  | $0.001 \pm 7.20e-5^{***}$             |                           |                           |                                       |

Table B.1: Result table of the Banknote data set



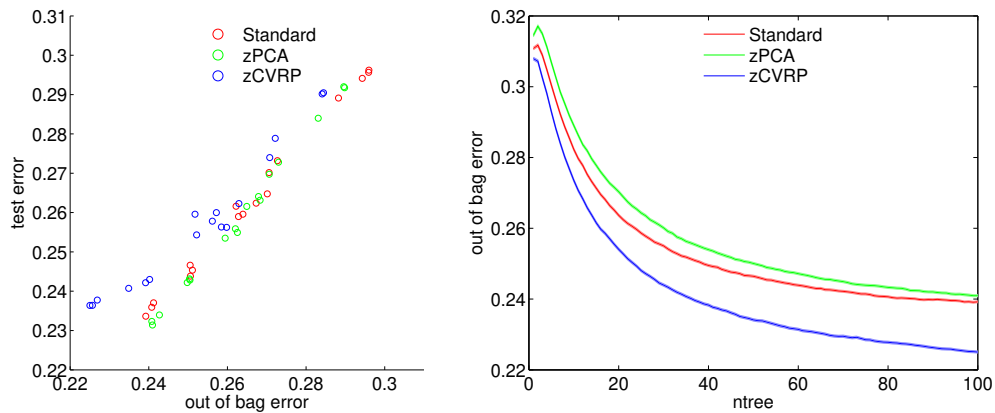
|   |                           |                           |                           |                                       |
|---|---------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Covertypes (20 runs) d=54, c=7, tot=581012, n=565892 |                           |                           |                           |                                       |
| Coordinate system: Standard d=54                          |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 6$           | $d_{trial} = 7$                       |
| $G_{gini}$  | $0.441 \pm 4.81e-3^{***}$ | $0.318 \pm 1.94e-3^{***}$ | $0.215 \pm 5.16e-4^{***}$ | <b><math>0.200 \pm 4.82e-4</math></b> |
| $G_{inf}$   | $0.484 \pm 3.22e-3^{***}$ | $0.368 \pm 1.02e-3^{***}$ | $0.282 \pm 9.42e-4^{***}$ | $0.267 \pm 7.90e-4^{***}$             |
| $G_{flip}$  | $0.480 \pm 2.85e-3^{***}$ | $0.364 \pm 1.36e-3^{***}$ | $0.229 \pm 7.00e-4^{***}$ | $0.210 \pm 5.79e-4^{***}$             |
| $G_{rs}$  | $0.474 \pm 3.73e-3^{***}$ | $0.350 \pm 1.21e-3^{***}$ | $0.226 \pm 5.30e-4^{***}$ | $0.211 \pm 5.93e-4^{***}$             |
| Best OOBE   | $0.200 \pm 5.47e-4$       |                           |                           |                                       |
| Coordinate system: zPCA d=108                             |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 7$           | $d_{trial} = 10$                      |
| $G_{gini}$  | $0.367 \pm 1.64e-3^{***}$ | $0.231 \pm 7.79e-4^{***}$ | $0.167 \pm 3.81e-4^{***}$ | <b><math>0.165 \pm 3.62e-4</math></b> |
| $G_{inf}$   | $0.425 \pm 2.28e-3^{***}$ | $0.258 \pm 9.80e-4^{***}$ | $0.173 \pm 4.44e-4^{***}$ | $0.170 \pm 4.08e-4^{***}$             |
| $G_{flip}$  | $0.425 \pm 2.45e-3^{***}$ | $0.262 \pm 1.01e-3^{***}$ | $0.172 \pm 4.00e-4^{***}$ | $0.168 \pm 4.37e-4^{***}$             |
| $G_{rs}$  | $0.406 \pm 1.78e-3^{***}$ | $0.245 \pm 8.47e-4^{***}$ | $0.169 \pm 3.93e-4^{***}$ | $0.166 \pm 3.71e-4$                   |
| Best OOBE   | $0.165 \pm 3.94e-4$       |                           |                           |                                       |
| Coordinate system: zCVRP d=432                            |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 9$           | $d_{trial} = 20$                      |
| $G_{gini}$  | $0.206 \pm 8.32e-4^{***}$ | $0.167 \pm 6.70e-4^{***}$ | $0.158 \pm 6.05e-4^{***}$ | $0.156 \pm 4.90e-4^{**}$              |
| $G_{inf}$   | $0.218 \pm 1.03e-3^{***}$ | $0.170 \pm 6.91e-4^{***}$ | $0.160 \pm 5.79e-4^{***}$ | $0.157 \pm 5.67e-4^{***}$             |
| $G_{flip}$  | $0.218 \pm 9.78e-4^{***}$ | $0.168 \pm 7.37e-4^{***}$ | $0.157 \pm 6.20e-4^{***}$ | <b><math>0.154 \pm 5.78e-4</math></b> |
| $G_{rs}$  | $0.211 \pm 9.19e-4^{***}$ | $0.168 \pm 7.02e-4^{***}$ | $0.157 \pm 5.84e-4^{***}$ | $0.154 \pm 5.36e-4$                   |
| Best OOBE   | $0.154 \pm 6.43e-4$       |                           |                           |                                       |

Table B.2: Result table of the Covertypes data set



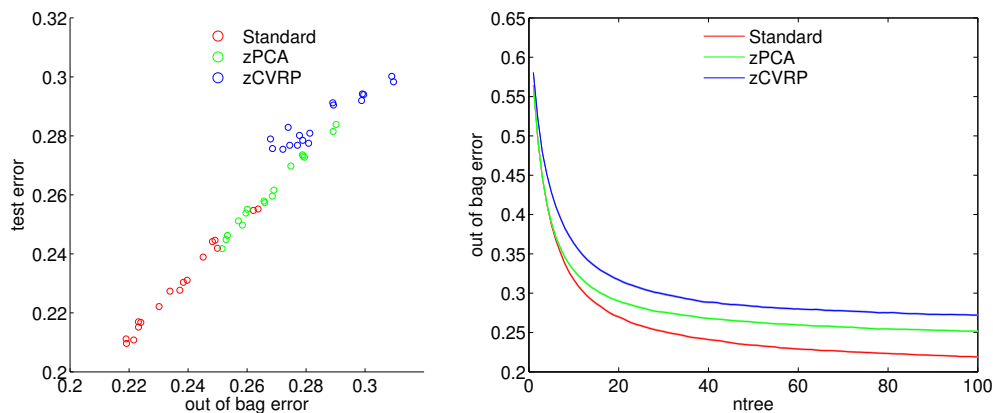
|  |                           |                           |                           |                                       |
|--|---------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Credit (1000 runs) d=24, c=2, tot=1000, n=100 |                           |                           |                           |                                       |
| Coordinate system: Standard d=24                   |                           |                           |                           |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 4$           | $d_{trial} = 5$                       |
| $G_{gini}$   | $0.289 \pm 1.38e-3^{***}$ | $0.247 \pm 1.31e-3^{***}$ | $0.236 \pm 1.30e-3$       | <b><math>0.234 \pm 1.31e-3</math></b> |
| $G_{inf}$  | $0.296 \pm 1.37e-3^{***}$ | $0.273 \pm 1.37e-3^{***}$ | $0.262 \pm 1.31e-3^{***}$ | $0.265 \pm 1.33e-3^{***}$             |
| $G_{flip}$   | $0.296 \pm 1.36e-3^{***}$ | $0.270 \pm 1.37e-3^{***}$ | $0.259 \pm 1.34e-3^{***}$ | $0.260 \pm 1.33e-3^{***}$             |
| $G_{rs}$   | $0.294 \pm 1.37e-3^{***}$ | $0.262 \pm 1.35e-3^{***}$ | $0.245 \pm 1.33e-3^{***}$ | $0.244 \pm 1.31e-3^{***}$             |
| Best OOBE  | $0.237 \pm 1.28e-3^*$     |                           |                           |                                       |
| Coordinate system: zPCA d=48                       |                           |                           |                           |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 6$           | $d_{trial} = 6$                       |
| $G_{gini}$   | $0.270 \pm 1.34e-3^{***}$ | $0.242 \pm 1.30e-3^{***}$ | $0.232 \pm 1.31e-3$       | <b><math>0.231 \pm 1.30e-3</math></b> |
| $G_{inf}$  | $0.292 \pm 1.36e-3^{***}$ | $0.262 \pm 1.31e-3^{***}$ | $0.255 \pm 1.29e-3^{***}$ | $0.256 \pm 1.33e-3^{***}$             |
| $G_{flip}$   | $0.292 \pm 1.37e-3^{***}$ | $0.273 \pm 1.37e-3^{***}$ | $0.264 \pm 1.33e-3^{***}$ | $0.263 \pm 1.34e-3^{***}$             |
| $G_{rs}$   | $0.284 \pm 1.36e-3^{***}$ | $0.253 \pm 1.32e-3^{***}$ | $0.243 \pm 1.31e-3^{***}$ | $0.243 \pm 1.30e-3^{***}$             |
| Best OOBE  | $0.234 \pm 1.28e-3$       |                           |                           |                                       |
| Coordinate system: zCVRP d=72                      |                           |                           |                           |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 7$           | $d_{trial} = 8$                       |
| $G_{gini}$   | $0.260 \pm 1.30e-3^{***}$ | $0.241 \pm 1.26e-3^{**}$  | $0.236 \pm 1.28e-3$       | <b><math>0.236 \pm 1.24e-3</math></b> |
| $G_{inf}$  | $0.290 \pm 1.36e-3^{***}$ | $0.262 \pm 1.33e-3^{***}$ | $0.256 \pm 1.32e-3^{***}$ | $0.256 \pm 1.33e-3^{***}$             |
| $G_{flip}$   | $0.290 \pm 1.35e-3^{***}$ | $0.274 \pm 1.35e-3^{***}$ | $0.260 \pm 1.34e-3^{***}$ | $0.258 \pm 1.34e-3^{***}$             |
| $G_{rs}$   | $0.279 \pm 1.38e-3^{***}$ | $0.254 \pm 1.32e-3^{***}$ | $0.243 \pm 1.30e-3^{***}$ | $0.242 \pm 1.30e-3^{***}$             |
| Best OOBE  | $0.238 \pm 1.25e-3$       |                           |                           |                                       |

Table B.3: Result table of the Credit data set



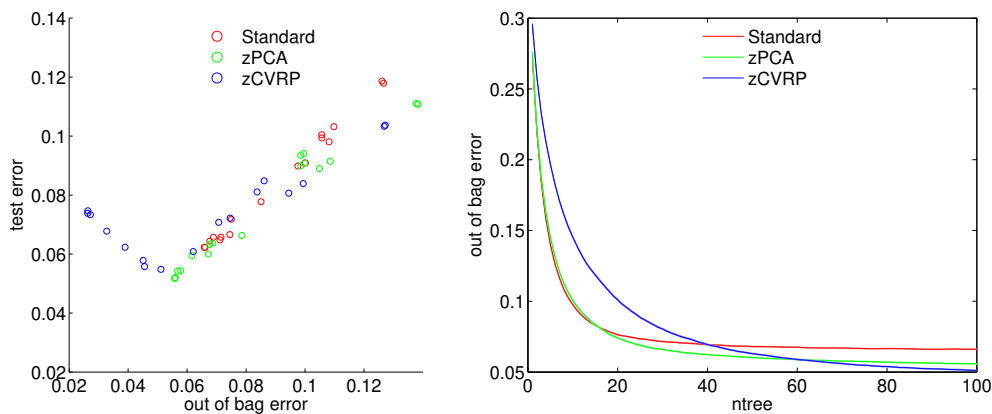
|  |                           |                           |                                       |                                       |
|--|---------------------------|---------------------------|---------------------------------------|---------------------------------------|
| Set: Glass (1000 runs) d=9, c=6, tot=214, n=22 |                           |                           |                                       |                                       |
| Coordinate system: Standard d=9                |                           |                           |                                       |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 3$                       | $d_{trial} = 4$                       |
| $G_{gini}$                                     | $0.228 \pm 2.80e-3^{***}$ | $0.211 \pm 2.71e-3$       | $0.217 \pm 2.66e-3^*$                 | $0.222 \pm 2.61e-3^{***}$             |
| $G_{inf}$                                      | $0.255 \pm 2.79e-3^{***}$ | $0.231 \pm 2.80e-3^{***}$ | $0.230 \pm 2.82e-3^{***}$             | $0.227 \pm 2.75e-3^{***}$             |
| $G_{flip}$                                     | $0.255 \pm 2.86e-3^{***}$ | $0.239 \pm 2.92e-3^{***}$ | $0.244 \pm 2.89e-3^{***}$             | $0.245 \pm 2.93e-3^{***}$             |
| $G_{rs}$                                       | $0.242 \pm 2.86e-3^{***}$ | $0.215 \pm 2.71e-3$       | $0.211 \pm 2.76e-3$                   | <b><math>0.210 \pm 2.71e-3</math></b> |
| Best OOBE                                      | $0.217 \pm 2.81e-3^*$     |                           |                                       |                                       |
| Coordinate system: zPCA d=18                   |                           |                           |                                       |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 4$                       | $d_{trial} = 5$                       |
| $G_{gini}$                                     | $0.258 \pm 2.86e-3^{***}$ | $0.245 \pm 2.90e-3$       | <b><math>0.242 \pm 2.76e-3</math></b> | $0.250 \pm 2.86e-3^*$                 |
| $G_{inf}$                                      | $0.282 \pm 3.01e-3^{***}$ | $0.270 \pm 2.90e-3^{***}$ | $0.255 \pm 2.89e-3^{***}$             | $0.260 \pm 2.90e-3^{***}$             |
| $G_{flip}$                                     | $0.284 \pm 2.93e-3^{***}$ | $0.273 \pm 2.96e-3^{***}$ | $0.262 \pm 2.89e-3^{***}$             | $0.274 \pm 2.97e-3^{***}$             |
| $G_{rs}$                                       | $0.273 \pm 2.95e-3^{***}$ | $0.257 \pm 2.89e-3^{***}$ | $0.246 \pm 2.88e-3$                   | $0.254 \pm 2.91e-3^{**}$              |
| Best OOBE                                      | $0.251 \pm 2.82e-3^{**}$  |                           |                                       |                                       |
| Coordinate system: zCVRP d=72                  |                           |                           |                                       |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 7$                       | $d_{trial} = 8$                       |
| $G_{gini}$                                     | $0.290 \pm 2.93e-3^{***}$ | $0.280 \pm 2.90e-3$       | $0.276 \pm 2.83e-3$                   | $0.279 \pm 2.83e-3$                   |
| $G_{inf}$                                      | $0.298 \pm 3.00e-3^{***}$ | $0.292 \pm 2.92e-3^{***}$ | $0.281 \pm 2.85e-3$                   | $0.277 \pm 2.84e-3$                   |
| $G_{flip}$                                     | $0.300 \pm 2.90e-3^{***}$ | $0.294 \pm 2.92e-3^{***}$ | $0.278 \pm 2.84e-3$                   | $0.278 \pm 2.88e-3$                   |
| $G_{rs}$                                       | $0.294 \pm 2.92e-3^{***}$ | $0.291 \pm 2.91e-3^{***}$ | $0.277 \pm 2.88e-3$                   | <b><math>0.275 \pm 2.87e-3</math></b> |
| Best OOBE                                      | $0.283 \pm 2.96e-3^*$     |                           |                                       |                                       |

Table B.4: Result table of the Glass data set



|  |                  |                      |                      |                      |
|--|------------------|----------------------|----------------------|----------------------|
| Set: Ionosphere (1000 runs) d=34, c=2, tot=351, n=36 |                  |                      |                      |                      |
| Coordinate system: Standard d=34                     |                  |                      |                      |                      |
|  | $d_{trial} = 1$  | $d_{trial} = 2$      | $d_{trial} = 5$      | $d_{trial} = 6$      |
| $G_{gini}$   | 0.072±1.29e-3*** | 0.066±1.27e-3*       | <b>0.062±1.22e-3</b> | 0.062±1.23e-3        |
| $G_{inf}$  | 0.119±1.68e-3*** | 0.103±1.52e-3***     | 0.100±1.54e-3***     | 0.099±1.49e-3***     |
| $G_{flip}$   | 0.118±1.68e-3*** | 0.098±1.54e-3***     | 0.091±1.50e-3***     | 0.090±1.47e-3***     |
| $G_{rs}$   | 0.078±1.38e-3*** | 0.067±1.25e-3**      | 0.066±1.26e-3*       | 0.065±1.22e-3        |
| Best OOBE  | 0.064±1.26e-3    |                      |                      |                      |
| Coordinate system: zPCA d=78                         |                  |                      |                      |                      |
|  | $d_{trial} = 1$  | $d_{trial} = 2$      | $d_{trial} = 7$      | $d_{trial} = 8$      |
| $G_{gini}$   | 0.059±1.20e-3*** | 0.054±1.17e-3        | 0.052±1.17e-3        | <b>0.052±1.16e-3</b> |
| $G_{inf}$  | 0.111±1.72e-3*** | 0.089±1.47e-3***     | 0.093±1.45e-3***     | 0.094±1.47e-3***     |
| $G_{flip}$   | 0.111±1.68e-3*** | 0.091±1.49e-3***     | 0.091±1.47e-3***     | 0.090±1.45e-3***     |
| $G_{rs}$   | 0.066±1.29e-3*** | 0.060±1.21e-3***     | 0.063±1.25e-3***     | 0.064±1.22e-3***     |
| Best OOBE  | 0.054±1.18e-3    |                      |                      |                      |
| Coordinate system: zCVRP d=102                       |                  |                      |                      |                      |
|  | $d_{trial} = 1$  | $d_{trial} = 2$      | $d_{trial} = 7$      | $d_{trial} = 10$     |
| $G_{gini}$   | 0.062±1.25e-3*** | 0.068±1.29e-3***     | 0.075±1.36e-3***     | 0.074±1.35e-3***     |
| $G_{inf}$  | 0.104±1.68e-3*** | 0.081±1.39e-3***     | 0.081±1.40e-3***     | 0.085±1.41e-3***     |
| $G_{flip}$   | 0.103±1.62e-3*** | 0.084±1.45e-3***     | 0.072±1.35e-3***     | 0.071±1.30e-3***     |
| $G_{rs}$   | 0.061±1.22e-3*** | <b>0.055±1.16e-3</b> | 0.056±1.17e-3        | 0.058±1.19e-3*       |
| Best OOBE  | 0.073±1.33e-3*** |                      |                      |                      |

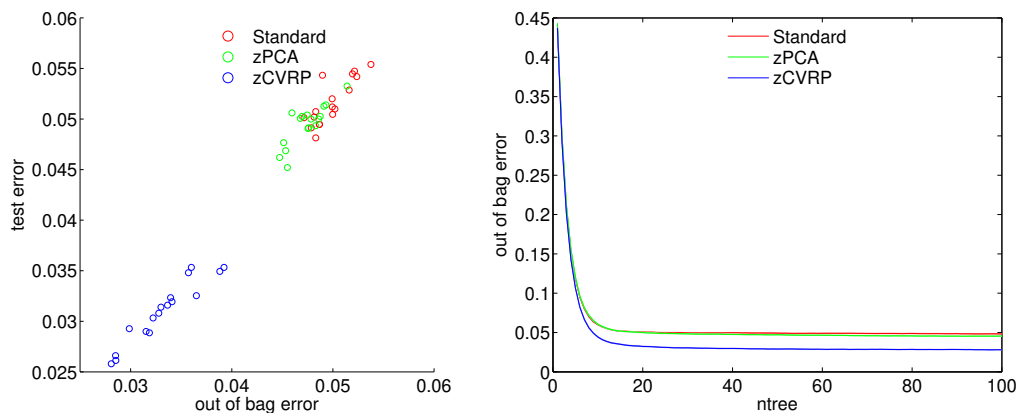
Table B.5: Result table of the Ionosphere data set





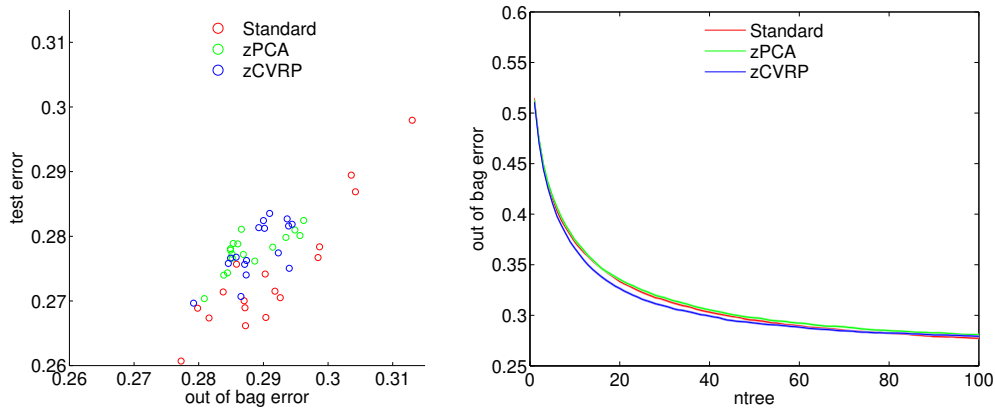
|   |                           |                           |                           |                                       |
|---|---------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Iris (1000 runs) d=4, c=3, tot=150, n=15 |                           |                           |                           |                                       |
| Coordinate system: Standard d=4               |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 3$                       |
| $G_{gini}$                                    | $0.055 \pm 1.69e-3^{**}$  | $0.054 \pm 1.71e-3^{**}$  | $0.054 \pm 1.73e-3^{**}$  | $0.055 \pm 1.71e-3^{**}$              |
| $G_{inf}$                                     | $0.051 \pm 1.65e-3$       | $0.050 \pm 1.65e-3$       | $0.051 \pm 1.66e-3$       | $0.050 \pm 1.61e-3$                   |
| $G_{flip}$                                    | $0.050 \pm 1.64e-3$       | $0.049 \pm 1.60e-3$       | $0.049 \pm 1.62e-3$       | $0.049 \pm 1.64e-3$                   |
| $G_{rs}$                                      | $0.053 \pm 1.68e-3^*$     | $0.052 \pm 1.68e-3$       | $0.051 \pm 1.64e-3$       | <b><math>0.048 \pm 1.66e-3</math></b> |
| Best OOBE                                     | $0.054 \pm 1.71e-3^{**}$  |                           |                           |                                       |
| Coordinate system: zPCA d=8                   |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 4$                       |
| $G_{gini}$                                    | $0.053 \pm 1.75e-3^{***}$ | $0.049 \pm 1.71e-3^*$     | $0.050 \pm 1.72e-3^*$     | <b><math>0.045 \pm 1.68e-3</math></b> |
| $G_{inf}$                                     | $0.050 \pm 1.68e-3^*$     | $0.050 \pm 1.69e-3^*$     | $0.050 \pm 1.67e-3^*$     | $0.048 \pm 1.63e-3$                   |
| $G_{flip}$                                    | $0.051 \pm 1.69e-3^{**}$  | $0.050 \pm 1.69e-3^*$     | $0.050 \pm 1.71e-3^*$     | $0.047 \pm 1.66e-3$                   |
| $G_{rs}$                                      | $0.051 \pm 1.71e-3^{**}$  | $0.049 \pm 1.70e-3^*$     | $0.049 \pm 1.69e-3$       | $0.046 \pm 1.67e-3$                   |
| Best OOBE                                     | $0.051 \pm 1.71e-3^*$     |                           |                           |                                       |
| Coordinate system: zCVRP d=16                 |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 4$           | $d_{trial} = 5$                       |
| $G_{gini}$                                    | $0.032 \pm 1.40e-3^{***}$ | $0.031 \pm 1.38e-3^{**}$  | $0.029 \pm 1.39e-3^*$     | <b><math>0.026 \pm 1.32e-3</math></b> |
| $G_{inf}$                                     | $0.035 \pm 1.49e-3^{***}$ | $0.035 \pm 1.46e-3^{***}$ | $0.035 \pm 1.46e-3^{***}$ | $0.029 \pm 1.36e-3$                   |
| $G_{flip}$                                    | $0.035 \pm 1.50e-3^{***}$ | $0.032 \pm 1.40e-3^{***}$ | $0.031 \pm 1.37e-3^{**}$  | $0.027 \pm 1.30e-3$                   |
| $G_{rs}$                                      | $0.033 \pm 1.44e-3^{***}$ | $0.032 \pm 1.41e-3^{**}$  | $0.030 \pm 1.40e-3^{**}$  | $0.026 \pm 1.30e-3$                   |
| Best OOBE                                     | $0.029 \pm 1.40e-3^*$     |                           |                           |                                       |

Table B.6: Result table of the Iris data set



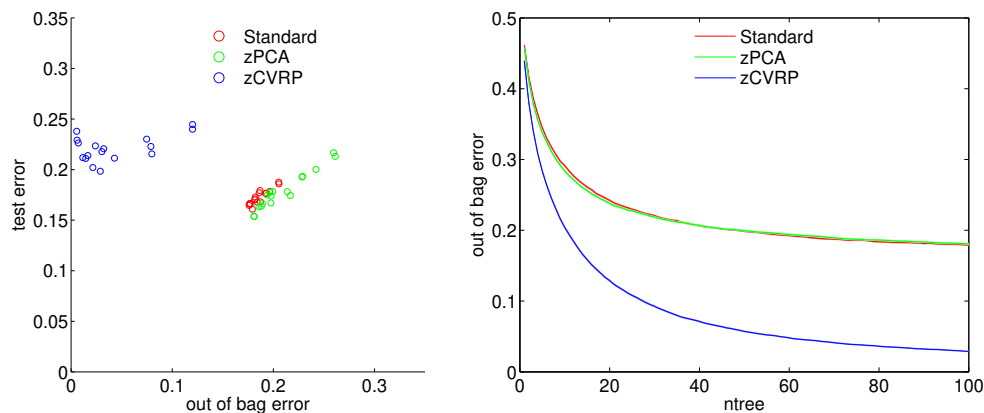
|  |                      |                  |                  |                      |
|--|----------------------|------------------|------------------|----------------------|
| Set: Liver (1000 runs) d=6, c=2, tot=345, n=35 |                      |                  |                  |                      |
| Coordinate system: Standard d=6                |                      |                  |                  |                      |
|  | $d_{trial} = 1$      | $d_{trial} = 2$  | $d_{trial} = 2$  | $d_{trial} = 3$      |
| $G_{gini}$                                     | <b>0.261±2.16e-3</b> | 0.269±2.20e-3**  | 0.267±2.21e-3*   | 0.276±2.24e-3***     |
| $G_{inf}$                                      | 0.278±2.21e-3***     | 0.272±2.22e-3*** | 0.271±2.23e-3*** | 0.267±2.24e-3*       |
| $G_{flip}$                                     | 0.277±2.27e-3***     | 0.287±2.26e-3*** | 0.289±2.26e-3*** | 0.298±2.31e-3***     |
| $G_{rs}$                                       | 0.266±2.23e-3*       | 0.270±2.21e-3**  | 0.269±2.19e-3**  | 0.274±2.25e-3***     |
| Best OOBE                                      | 0.271±2.21e-3***     |                  |                  |                      |
| Coordinate system: zPCA d=12                   |                      |                  |                  |                      |
|  | $d_{trial} = 1$      | $d_{trial} = 2$  | $d_{trial} = 3$  | $d_{trial} = 4$      |
| $G_{gini}$                                     | 0.277±2.23e-3*       | 0.278±2.16e-3**  | 0.279±2.22e-3**  | 0.281±2.24e-3***     |
| $G_{inf}$                                      | 0.282±2.32e-3***     | 0.277±2.32e-3*   | 0.274±2.27e-3    | <b>0.270±2.27e-3</b> |
| $G_{flip}$                                     | 0.280±2.30e-3**      | 0.281±2.29e-3*** | 0.280±2.33e-3**  | 0.278±2.24e-3**      |
| $G_{rs}$                                       | 0.276±2.29e-3*       | 0.278±2.24e-3**  | 0.277±2.32e-3*   | 0.274±2.20e-3        |
| Best OOBE                                      | 0.279±2.17e-3**      |                  |                  |                      |
| Coordinate system: zCVRP d=18                  |                      |                  |                  |                      |
|  | $d_{trial} = 1$      | $d_{trial} = 2$  | $d_{trial} = 4$  | $d_{trial} = 5$      |
| $G_{gini}$                                     | <b>0.270±2.22e-3</b> | 0.276±2.31e-3*   | 0.281±2.25e-3*** | 0.284±2.24e-3***     |
| $G_{inf}$                                      | 0.277±2.31e-3**      | 0.274±2.26e-3    | 0.277±2.25e-3*   | 0.276±2.24e-3*       |
| $G_{flip}$                                     | 0.275±2.31e-3*       | 0.282±2.34e-3*** | 0.283±2.27e-3*** | 0.282±2.30e-3***     |
| $G_{rs}$                                       | 0.271±2.33e-3        | 0.276±2.26e-3*   | 0.281±2.25e-3*** | 0.282±2.27e-3***     |
| Best OOBE                                      | 0.277±2.27e-3*       |                  |                  |                      |

Table B.7: Result table of the Liver data set



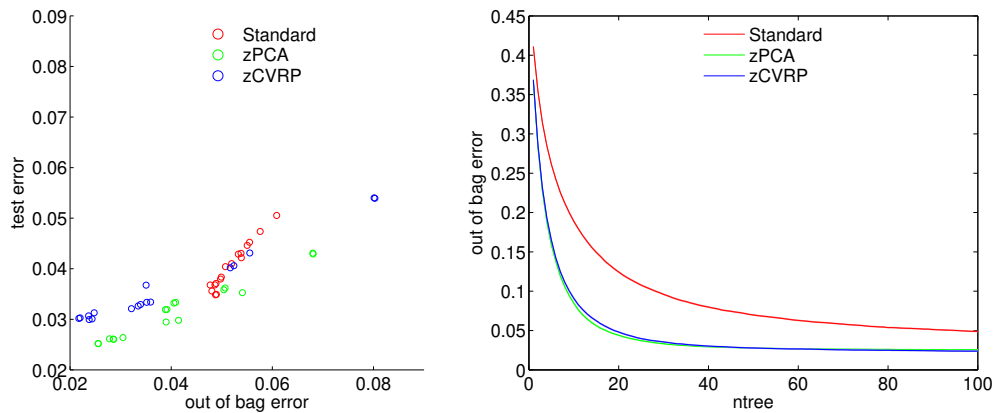
|   |                           |                                       |                                       |                           |
|---|---------------------------|---------------------------------------|---------------------------------------|---------------------------|
| Set: Sonar (1000 runs) d=60, c=2, tot=208, n=21 |                           |                                       |                                       |                           |
| Coordinate system: Standard d=60                |                           |                                       |                                       |                           |
|   | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 6$                       | $d_{trial} = 7$           |
| $G_{gini}$                                      | $0.168 \pm 2.63e-3^*$     | <b><math>0.161 \pm 2.53e-3</math></b> | $0.165 \pm 2.55e-3$                   | $0.166 \pm 2.58e-3$       |
| $G_{inf}$                                       | $0.188 \pm 2.78e-3^{***}$ | $0.177 \pm 2.62e-3^{***}$             | $0.171 \pm 2.60e-3^{**}$              | $0.173 \pm 2.66e-3^{***}$ |
| $G_{flip}$                                      | $0.186 \pm 2.69e-3^{***}$ | $0.176 \pm 2.70e-3^{***}$             | $0.177 \pm 2.63e-3^{***}$             | $0.179 \pm 2.67e-3^{***}$ |
| $G_{rs}$  | $0.178 \pm 2.67e-3^{***}$ | $0.168 \pm 2.57e-3^*$                 | $0.166 \pm 2.53e-3$                   | $0.166 \pm 2.60e-3$       |
| Best OOBE                                       | $0.170 \pm 2.59e-3^{**}$  |                                       |                                       |                           |
| Coordinate system: zPCA d=120                   |                           |                                       |                                       |                           |
|   | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 7$                       | $d_{trial} = 10$          |
| $G_{gini}$                                      | $0.174 \pm 2.65e-3^{***}$ | $0.167 \pm 2.56e-3^{***}$             | <b><math>0.153 \pm 2.50e-3</math></b> | $0.154 \pm 2.57e-3$       |
| $G_{inf}$                                       | $0.213 \pm 2.81e-3^{***}$ | $0.193 \pm 2.71e-3^{***}$             | $0.174 \pm 2.66e-3^{***}$             | $0.175 \pm 2.65e-3^{***}$ |
| $G_{flip}$                                      | $0.217 \pm 2.81e-3^{***}$ | $0.193 \pm 2.73e-3^{***}$             | $0.178 \pm 2.67e-3^{***}$             | $0.178 \pm 2.58e-3^{***}$ |
| $G_{rs}$  | $0.200 \pm 2.84e-3^{***}$ | $0.178 \pm 2.58e-3^{***}$             | $0.166 \pm 2.55e-3^{***}$             | $0.163 \pm 2.61e-3^{**}$  |
| Best OOBE                                       | $0.164 \pm 2.55e-3^{**}$  |                                       |                                       |                           |
| Coordinate system: zCVRP d=180                  |                           |                                       |                                       |                           |
|   | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 8$                       | $d_{trial} = 13$          |
| $G_{gini}$                                      | $0.221 \pm 2.90e-3^{***}$ | $0.214 \pm 2.87e-3^{***}$             | $0.229 \pm 2.97e-3^{***}$             | $0.238 \pm 3.11e-3^{***}$ |
| $G_{inf}$                                       | $0.245 \pm 3.12e-3^{***}$ | $0.216 \pm 2.91e-3^{***}$             | <b><math>0.198 \pm 2.83e-3</math></b> | $0.202 \pm 2.83e-3$       |
| $G_{flip}$                                      | $0.240 \pm 2.94e-3^{***}$ | $0.223 \pm 2.98e-3^{***}$             | $0.218 \pm 2.93e-3^{***}$             | $0.224 \pm 3.04e-3^{***}$ |
| $G_{rs}$  | $0.230 \pm 2.82e-3^{***}$ | $0.211 \pm 2.96e-3^{***}$             | $0.211 \pm 2.84e-3^{***}$             | $0.212 \pm 2.85e-3^{***}$ |
| Best OOBE                                       | $0.226 \pm 2.90e-3^{***}$ |                                       |                                       |                           |

Table B.8: Result table of the Sonar data set



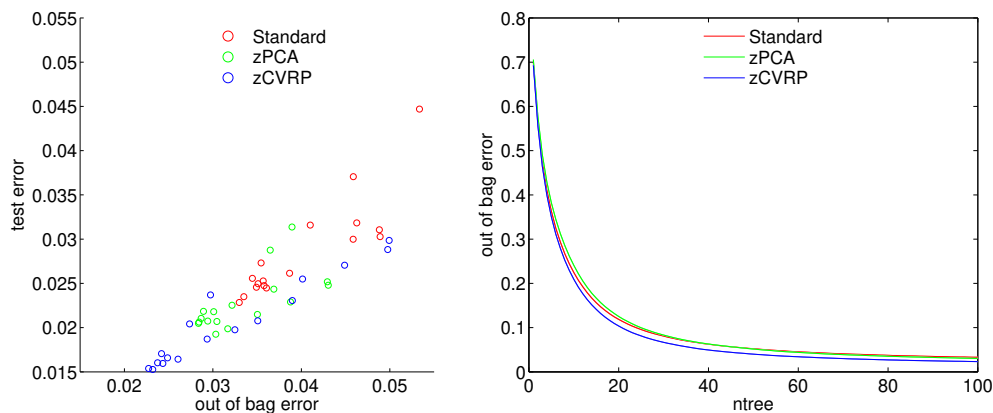
|  |                                       |                           |                           |                                       |
|--|---------------------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Twonorm (1000 runs) d=20, c=2, tot=7400, n=7100 |                                       |                           |                           |                                       |
| Coordinate system: Standard d=20                     |                                       |                           |                           |                                       |
|  | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 4$           | $d_{trial} = 5$                       |
| $G_{gini}$   | $0.037 \pm 8.62e-5^{***}$             | $0.040 \pm 1.03e-4^{***}$ | $0.047 \pm 1.58e-4^{***}$ | $0.051 \pm 1.94e-4^{***}$             |
| $G_{inf}$  | <b><math>0.035 \pm 9.10e-5</math></b> | $0.037 \pm 9.42e-5^{***}$ | $0.041 \pm 1.11e-4^{***}$ | $0.043 \pm 1.23e-4^{***}$             |
| $G_{flip}$   | $0.035 \pm 8.79e-5$                   | $0.037 \pm 9.90e-5^{***}$ | $0.042 \pm 1.32e-4^{***}$ | $0.045 \pm 1.50e-4^{***}$             |
| $G_{rs}$   | $0.036 \pm 8.92e-5^{***}$             | $0.038 \pm 9.20e-5^{***}$ | $0.043 \pm 1.25e-4^{***}$ | $0.045 \pm 1.45e-4^{***}$             |
| Best OOB   | $0.038 \pm 1.53e-4^{***}$             |                           |                           |                                       |
| Coordinate system: zPCA d=40                         |                                       |                           |                           |                                       |
|  | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 6$           | $d_{trial} = 6$                       |
| $G_{gini}$   | $0.030 \pm 8.91e-5^{***}$             | $0.026 \pm 6.38e-5^{***}$ | $0.025 \pm 6.29e-5$       | <b><math>0.025 \pm 6.28e-5</math></b> |
| $G_{inf}$  | $0.043 \pm 1.58e-4^{***}$             | $0.036 \pm 1.03e-4^{***}$ | $0.033 \pm 9.35e-5^{***}$ | $0.033 \pm 9.35e-5^{***}$             |
| $G_{flip}$   | $0.043 \pm 1.64e-4^{***}$             | $0.036 \pm 1.03e-4^{***}$ | $0.032 \pm 8.60e-5^{***}$ | $0.032 \pm 8.53e-5^{***}$             |
| $G_{rs}$   | $0.035 \pm 1.24e-4^{***}$             | $0.029 \pm 7.83e-5^{***}$ | $0.026 \pm 5.96e-5^{***}$ | $0.026 \pm 6.16e-5^{***}$             |
| Best OOB   | $0.026 \pm 8.97e-5^{***}$             |                           |                           |                                       |
| Coordinate system: zCVRP d=60                        |                                       |                           |                           |                                       |
|  | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 6$           | $d_{trial} = 7$                       |
| $G_{gini}$   | $0.037 \pm 1.33e-4^{***}$             | $0.031 \pm 8.56e-5^{***}$ | $0.030 \pm 8.93e-5^*$     | $0.030 \pm 9.01e-5^{**}$              |
| $G_{inf}$  | $0.054 \pm 2.31e-4^{***}$             | $0.041 \pm 1.30e-4^{***}$ | $0.033 \pm 9.05e-5^{***}$ | $0.033 \pm 8.74e-5^{***}$             |
| $G_{flip}$   | $0.054 \pm 2.31e-4^{***}$             | $0.040 \pm 1.29e-4^{***}$ | $0.033 \pm 8.43e-5^{***}$ | $0.032 \pm 8.35e-5^{***}$             |
| $G_{rs}$   | $0.043 \pm 1.83e-4^{***}$             | $0.033 \pm 9.75e-5^{***}$ | $0.030 \pm 8.15e-5$       | <b><math>0.030 \pm 8.12e-5</math></b> |
| Best OOB   | $0.031 \pm 9.51e-5^{***}$             |                           |                           |                                       |

Table B.9: Result table of the Twonorm data set



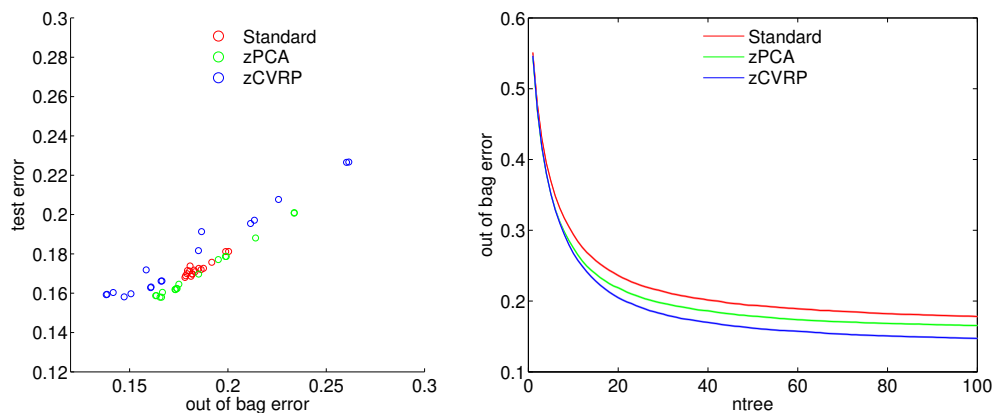
|  |                           |                                       |                                       |                           |
|--|---------------------------|---------------------------------------|---------------------------------------|---------------------------|
| Set: Vowel (1000 runs) d=10, c=11, tot=990, n=99 |                           |                                       |                                       |                           |
| Coordinate system: Standard d=10                 |                           |                                       |                                       |                           |
|  | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 3$                       | $d_{trial} = 4$           |
| $G_{gini}$                                       | $0.032 \pm 5.77e-4^{***}$ | $0.032 \pm 5.63e-4^{***}$             | $0.037 \pm 6.20e-4^{***}$             | $0.045 \pm 6.68e-4^{***}$ |
| $G_{inf}$  | $0.030 \pm 5.82e-4^{***}$ | $0.026 \pm 5.23e-4^{***}$             | $0.024 \pm 5.03e-4^{**}$              | $0.025 \pm 4.97e-4^{**}$  |
| $G_{flip}$                                       | $0.031 \pm 5.81e-4^{***}$ | $0.025 \pm 4.81e-4^{**}$              | <b><math>0.023 \pm 4.78e-4</math></b> | $0.023 \pm 4.65e-4$       |
| $G_{rs}$   | $0.030 \pm 5.59e-4^{***}$ | $0.025 \pm 5.27e-4^{***}$             | $0.026 \pm 5.18e-4^{***}$             | $0.027 \pm 5.14e-4^{***}$ |
| Best OOB   | $0.025 \pm 4.97e-4^{**}$  |                                       |                                       |                           |
| Coordinate system: zPCA d=20                     |                           |                                       |                                       |                           |
|  | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 4$                       | $d_{trial} = 5$           |
| $G_{gini}$                                       | $0.024 \pm 5.26e-4^{***}$ | $0.023 \pm 5.14e-4^{***}$             | $0.029 \pm 5.58e-4^{***}$             | $0.031 \pm 5.70e-4^{***}$ |
| $G_{inf}$  | $0.025 \pm 5.30e-4^{***}$ | $0.021 \pm 5.11e-4^{***}$             | $0.021 \pm 4.69e-4^*$                 | $0.022 \pm 4.78e-4^{***}$ |
| $G_{flip}$                                       | $0.025 \pm 5.24e-4^{***}$ | $0.020 \pm 4.77e-4$                   | $0.020 \pm 4.68e-4^*$                 | $0.021 \pm 4.66e-4^{**}$  |
| $G_{rs}$   | $0.023 \pm 5.13e-4^{***}$ | <b><math>0.019 \pm 4.65e-4</math></b> | $0.021 \pm 4.82e-4^*$                 | $0.022 \pm 4.83e-4^{***}$ |
| Best OOB   | $0.021 \pm 4.79e-4^*$     |                                       |                                       |                           |
| Coordinate system: zCVRP d=120                   |                           |                                       |                                       |                           |
|  | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 7$                       | $d_{trial} = 10$          |
| $G_{gini}$                                       | $0.025 \pm 5.53e-4^{***}$ | $0.019 \pm 4.77e-4^{***}$             | $0.020 \pm 4.74e-4^{***}$             | $0.024 \pm 5.23e-4^{***}$ |
| $G_{inf}$  | $0.030 \pm 5.92e-4^{***}$ | $0.023 \pm 5.13e-4^{***}$             | $0.016 \pm 4.29e-4^*$                 | $0.017 \pm 4.36e-4^*$     |
| $G_{flip}$                                       | $0.029 \pm 5.83e-4^{***}$ | $0.021 \pm 5.07e-4^{***}$             | <b><math>0.015 \pm 4.13e-4</math></b> | $0.015 \pm 4.12e-4$       |
| $G_{rs}$   | $0.027 \pm 5.78e-4^{***}$ | $0.020 \pm 4.83e-4^{***}$             | $0.016 \pm 4.14e-4$                   | $0.017 \pm 4.32e-4^{**}$  |
| Best OOB   | $0.016 \pm 4.28e-4$       |                                       |                                       |                           |

Table B.10: Result table of the Vowel data set



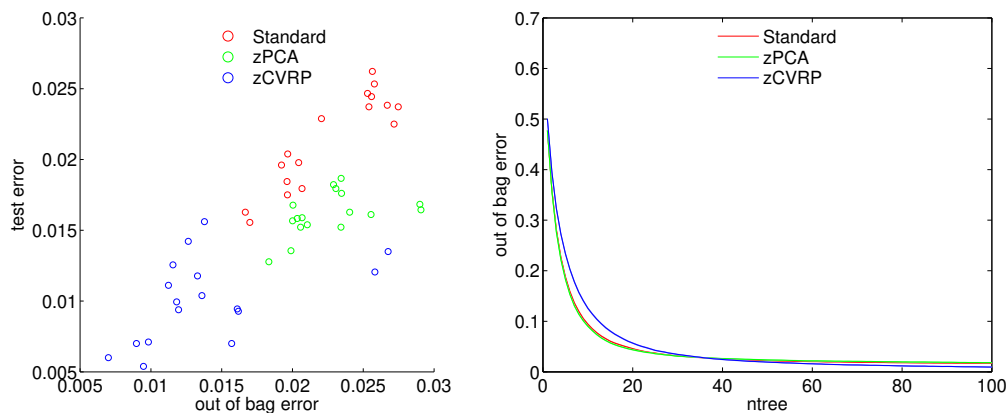
|   |                           |                                       |                                       |                                       |
|---|---------------------------|---------------------------------------|---------------------------------------|---------------------------------------|
| Set: Waveform (1000 runs) d=21, c=3, tot=5000, n=4700 |                           |                                       |                                       |                                       |
| Coordinate system: Standard d=21                      |                           |                                       |                                       |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 4$                       | $d_{trial} = 5$                       |
| $G_{gini}$  | $0.173 \pm 2.36e-4^{***}$ | <b><math>0.168 \pm 2.00e-4</math></b> | $0.172 \pm 2.14e-4^{***}$             | $0.174 \pm 2.21e-4^{***}$             |
| $G_{inf}$   | $0.181 \pm 3.82e-4^{***}$ | $0.173 \pm 3.07e-4^{***}$             | $0.170 \pm 2.47e-4^{***}$             | $0.170 \pm 2.36e-4^{***}$             |
| $G_{flip}$  | $0.181 \pm 3.80e-4^{***}$ | $0.172 \pm 3.08e-4^{***}$             | $0.171 \pm 2.71e-4^{***}$             | $0.172 \pm 2.72e-4^{***}$             |
| $G_{rs}$  | $0.176 \pm 3.05e-4^{***}$ | $0.169 \pm 2.48e-4^*$                 | $0.169 \pm 2.12e-4^{**}$              | $0.170 \pm 2.19e-4^{***}$             |
| Best OOB  | $0.171 \pm 2.54e-4^{***}$ |                                       |                                       |                                       |
| Coordinate system: zPCA d=42                          |                           |                                       |                                       |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 6$                       | $d_{trial} = 6$                       |
| $G_{gini}$  | $0.177 \pm 3.10e-4^{***}$ | $0.165 \pm 2.32e-4^{***}$             | $0.159 \pm 1.84e-4^{**}$              | $0.159 \pm 1.84e-4^{**}$              |
| $G_{inf}$   | $0.201 \pm 4.73e-4^{***}$ | $0.179 \pm 3.58e-4^{***}$             | $0.162 \pm 2.39e-4^{***}$             | $0.162 \pm 2.38e-4^{***}$             |
| $G_{flip}$  | $0.201 \pm 4.84e-4^{***}$ | $0.179 \pm 3.80e-4^{***}$             | $0.162 \pm 2.46e-4^{***}$             | $0.162 \pm 2.56e-4^{***}$             |
| $G_{rs}$  | $0.188 \pm 4.16e-4^{***}$ | $0.170 \pm 2.99e-4^{***}$             | <b><math>0.158 \pm 2.05e-4</math></b> | $0.158 \pm 2.01e-4$                   |
| Best OOB  | $0.160 \pm 2.39e-4^{***}$ |                                       |                                       |                                       |
| Coordinate system: zCVRP d=84                         |                           |                                       |                                       |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$                       | $d_{trial} = 7$                       | $d_{trial} = 9$                       |
| $G_{gini}$  | $0.191 \pm 3.37e-4^{***}$ | $0.172 \pm 2.44e-4^{***}$             | $0.159 \pm 1.79e-4^{***}$             | $0.159 \pm 1.86e-4^{***}$             |
| $G_{inf}$   | $0.227 \pm 5.34e-4^{***}$ | $0.195 \pm 4.05e-4^{***}$             | $0.166 \pm 2.47e-4^{***}$             | $0.163 \pm 2.27e-4^{***}$             |
| $G_{flip}$  | $0.227 \pm 5.29e-4^{***}$ | $0.197 \pm 4.26e-4^{***}$             | $0.166 \pm 2.59e-4^{***}$             | $0.163 \pm 2.34e-4^{***}$             |
| $G_{rs}$  | $0.208 \pm 4.50e-4^{***}$ | $0.182 \pm 3.17e-4^{***}$             | $0.160 \pm 1.96e-4^{***}$             | <b><math>0.158 \pm 1.88e-4</math></b> |
| Best OOB  | $0.160 \pm 2.16e-4^{***}$ |                                       |                                       |                                       |

Table B.11: Result table of the Waveform data set



|  |                      |                      |                  |                  |
|--|----------------------|----------------------|------------------|------------------|
| Set: Wine (1000 runs) d=13, c=3, tot=178, n=18 |                      |                      |                  |                  |
| Coordinate system: Standard d=13               |                      |                      |                  |                  |
|  | $d_{trial} = 1$      | $d_{trial} = 2$      | $d_{trial} = 3$  | $d_{trial} = 4$  |
| $G_{gini}$                                     | <b>0.016±8.67e-4</b> | 0.016±9.05e-4        | 0.020±1.02e-3**  | 0.023±1.10e-3*** |
| $G_{inf}$                                      | 0.024±1.08e-3***     | 0.024±1.10e-3***     | 0.025±1.16e-3*** | 0.024±1.11e-3*** |
| $G_{flip}$                                     | 0.023±1.08e-3***     | 0.024±1.15e-3***     | 0.025±1.17e-3*** | 0.026±1.19e-3*** |
| $G_{rs}$                                       | 0.018±9.89e-4*       | 0.018±9.46e-4        | 0.018±1.01e-3*   | 0.020±1.01e-3*** |
| Best OOB E                                     | 0.020±1.08e-3***     |                      |                  |                  |
| Coordinate system: zPCA d=26                   |                      |                      |                  |                  |
|  | $d_{trial} = 1$      | $d_{trial} = 2$      | $d_{trial} = 5$  | $d_{trial} = 5$  |
| $G_{gini}$                                     | 0.014±8.17e-4        | <b>0.013±7.76e-4</b> | 0.017±9.28e-4*** | 0.016±9.09e-4**  |
| $G_{inf}$                                      | 0.017±9.32e-4***     | 0.016±9.23e-4**      | 0.018±9.30e-4*** | 0.019±9.64e-4*** |
| $G_{flip}$                                     | 0.016±9.20e-4**      | 0.016±9.11e-4**      | 0.018±9.37e-4*** | 0.018±9.43e-4*** |
| $G_{rs}$                                       | 0.015±8.59e-4*       | 0.015±8.75e-4*       | 0.016±8.76e-4**  | 0.015±8.62e-4*   |
| Best OOB E                                     | 0.016±9.00e-4**      |                      |                  |                  |
| Coordinate system: zCVRP d=52                  |                      |                      |                  |                  |
|  | $d_{trial} = 1$      | $d_{trial} = 2$      | $d_{trial} = 6$  | $d_{trial} = 7$  |
| $G_{gini}$                                     | 0.007±6.24e-4*       | 0.006±5.73e-4        | 0.014±8.90e-4*** | 0.016±9.33e-4*** |
| $G_{inf}$                                      | 0.013±8.64e-4***     | 0.009±6.96e-4***     | 0.009±7.17e-4*** | 0.010±7.18e-4*** |
| $G_{flip}$                                     | 0.012±8.01e-4***     | 0.009±7.05e-4***     | 0.010±7.50e-4*** | 0.012±8.22e-4*** |
| $G_{rs}$                                       | 0.007±6.14e-4*       | <b>0.005±5.43e-4</b> | 0.011±7.74e-4*** | 0.013±8.14e-4*** |
| Best OOB E                                     | 0.007±6.28e-4*       |                      |                  |                  |

Table B.12: Result table of the Wine data set

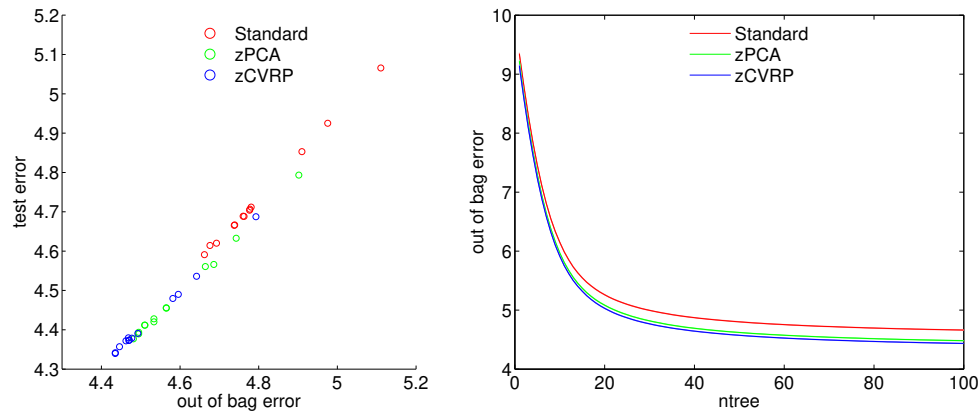


## B.2 Random Forest Regression Results



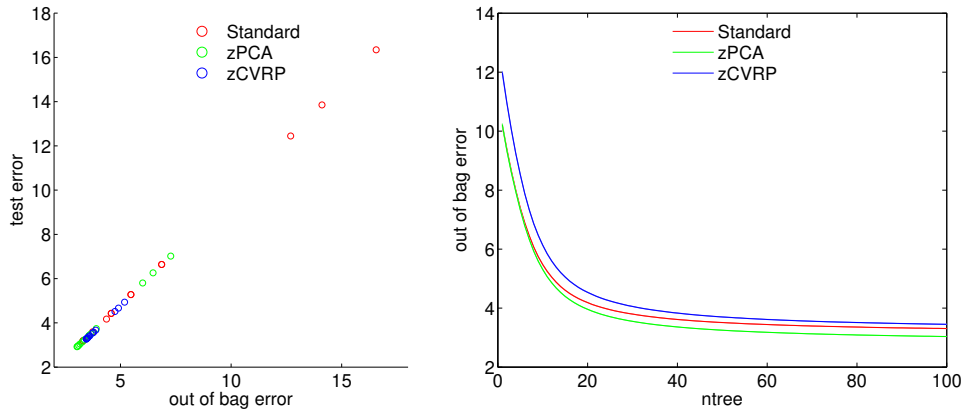
|   |                           |                           |                           |                                       |
|---|---------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Abalone (1000 runs) d=8, c=0, tot=4177, n=1045 |                           |                           |                           |                                       |
| Coordinate system: Standard d=8                     |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 4$                       |
| $G_{varr}$  | $4.853 \pm 1.07e-2^{***}$ | $4.688 \pm 9.74e-3^{***}$ | $4.689 \pm 9.79e-3^{***}$ | $4.712 \pm 9.36e-3^{***}$             |
| $G_{varf}$  | $5.066 \pm 1.17e-2^{***}$ | $4.704 \pm 1.04e-2^{***}$ | $4.707 \pm 1.04e-2^{***}$ | <b><math>4.591 \pm 9.67e-3</math></b> |
| $G_{rs}$  | $4.925 \pm 1.11e-2^{***}$ | $4.667 \pm 9.86e-3^{***}$ | $4.666 \pm 9.86e-3^{***}$ | $4.620 \pm 9.44e-3^*$                 |
| Best OOB  | $4.614 \pm 9.72e-3^*$     |                           |                           |                                       |
| Coordinate system: zPCA d=16                        |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 4$           | $d_{trial} = 5$                       |
| $G_{varr}$  | $4.561 \pm 9.71e-3^{***}$ | $4.428 \pm 8.98e-3^{***}$ | $4.412 \pm 8.73e-3^{**}$  | $4.412 \pm 8.66e-3^{**}$              |
| $G_{varf}$  | $4.793 \pm 1.08e-2^{***}$ | $4.566 \pm 9.91e-3^{***}$ | $4.455 \pm 9.40e-3^{***}$ | $4.420 \pm 9.27e-3^{***}$             |
| $G_{rs}$  | $4.633 \pm 1.01e-2^{***}$ | $4.456 \pm 9.23e-3^{***}$ | $4.390 \pm 8.84e-3$       | <b><math>4.377 \pm 8.82e-3</math></b> |
| Best OOB  | $4.394 \pm 8.81e-3$       |                           |                           |                                       |
| Coordinate system: zCVRP d=32                       |                           |                           |                           |                                       |
|   | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 5$           | $d_{trial} = 6$                       |
| $G_{varr}$  | $4.480 \pm 9.32e-3^{***}$ | $4.374 \pm 8.90e-3^{**}$  | $4.372 \pm 8.67e-3^{**}$  | $4.380 \pm 8.68e-3^{***}$             |
| $G_{varf}$  | $4.687 \pm 1.04e-2^{***}$ | $4.490 \pm 9.73e-3^{***}$ | $4.379 \pm 9.15e-3^{**}$  | $4.373 \pm 9.08e-3^{**}$              |
| $G_{rs}$  | $4.536 \pm 9.65e-3^{***}$ | $4.391 \pm 9.11e-3^{***}$ | $4.342 \pm 8.73e-3$       | <b><math>4.340 \pm 8.75e-3</math></b> |
| Best OOB  | $4.357 \pm 8.83e-3$       |                           |                           |                                       |

Table B.13: Result table of the Abalone data set



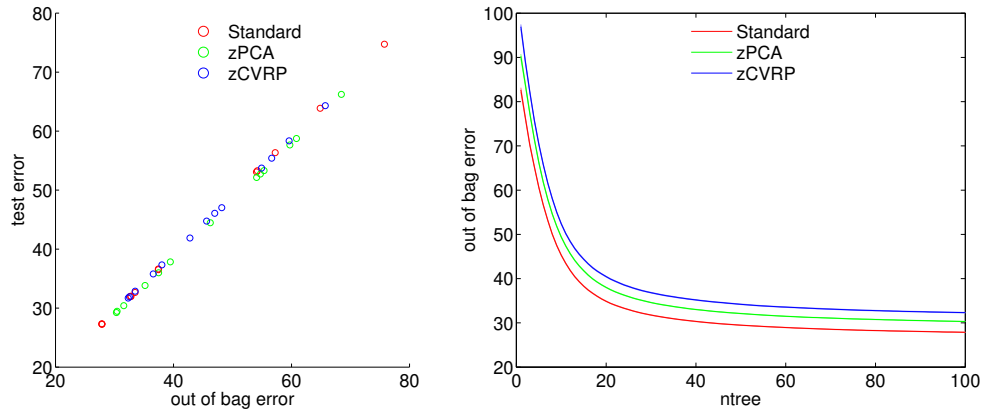
|  |                                       |                           |                           |                                       |
|--|---------------------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Airfoil (1000 runs) d=5, c=0, tot=1503, n=151 |                                       |                           |                           |                                       |
| Coordinate system: Standard d=5                    |                                       |                           |                           |                                       |
|  | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 3$                       |
| $G_{varr}$   | $12.446 \pm 4.82e-2^{***}$            | $4.432 \pm 2.60e-2^{***}$ | $4.426 \pm 2.57e-2^{***}$ | $3.193 \pm 2.13e-2$                   |
| $G_{varf}$   | $16.343 \pm 5.92e-2^{***}$            | $6.638 \pm 3.33e-2^{***}$ | $6.640 \pm 3.32e-2^{***}$ | $4.170 \pm 2.49e-2^{***}$             |
| $G_{rs}$   | $13.854 \pm 5.27e-2^{***}$            | $5.282 \pm 2.92e-2^{***}$ | $5.279 \pm 2.87e-2^{***}$ | $3.597 \pm 2.24e-2^{***}$             |
| Best OOB E   | <b><math>3.192 \pm 2.09e-2</math></b> |                           |                           |                                       |
| Coordinate system: zPCA d=10                       |                                       |                           |                           |                                       |
|  | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 3$           | $d_{trial} = 4$                       |
| $G_{varr}$   | $5.803 \pm 3.00e-2^{***}$             | $3.231 \pm 2.10e-2^{***}$ | $2.984 \pm 1.98e-2^*$     | <b><math>2.920 \pm 1.92e-2</math></b> |
| $G_{varf}$   | $7.018 \pm 3.31e-2^{***}$             | $3.740 \pm 2.32e-2^{***}$ | $3.502 \pm 2.21e-2^{***}$ | $3.431 \pm 2.20e-2^{***}$             |
| $G_{rs}$   | $6.262 \pm 3.10e-2^{***}$             | $3.411 \pm 2.14e-2^{***}$ | $3.135 \pm 2.02e-2^{***}$ | $3.042 \pm 1.99e-2^{***}$             |
| Best OOB E   | $2.939 \pm 1.91e-2$                   |                           |                           |                                       |
| Coordinate system: zCVRP d=20                      |                                       |                           |                           |                                       |
|  | $d_{trial} = 1$                       | $d_{trial} = 2$           | $d_{trial} = 4$           | $d_{trial} = 5$                       |
| $G_{varr}$   | $4.517 \pm 2.55e-2^{***}$             | $3.545 \pm 2.17e-2^{***}$ | $3.316 \pm 2.07e-2^*$     | $3.275 \pm 2.09e-2$                   |
| $G_{varf}$   | $4.936 \pm 2.74e-2^{***}$             | $3.670 \pm 2.25e-2^{***}$ | $3.419 \pm 2.18e-2^{***}$ | $3.366 \pm 2.19e-2^{***}$             |
| $G_{rs}$   | $4.666 \pm 2.60e-2^{***}$             | $3.578 \pm 2.20e-2^{***}$ | $3.318 \pm 2.08e-2^*$     | <b><math>3.265 \pm 2.08e-2</math></b> |
| Best OOB E   | $3.287 \pm 2.11e-2$                   |                           |                           |                                       |

Table B.14: Result table of the Airfoil data set



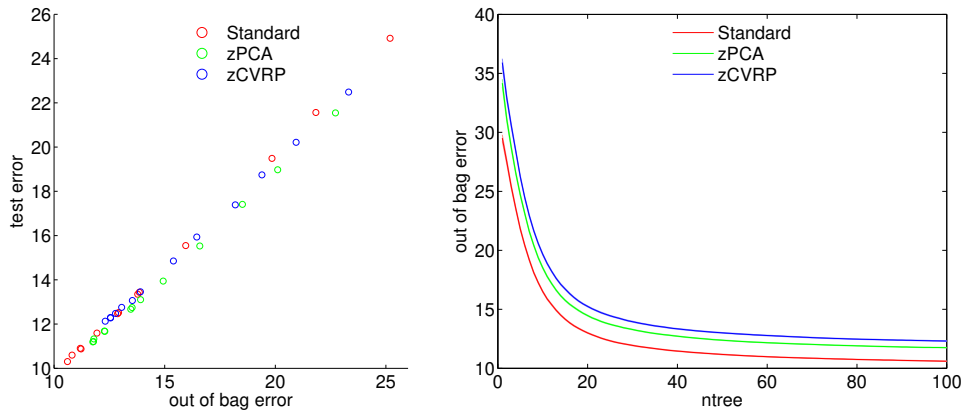
|   |                            |                            |                            |  |
|---|----------------------------|----------------------------|----------------------------|--|
| Set: Concrete (1000 runs) d=8, c=0, tot=1030, n=103 |                            |                            |                            |  |
| Coordinate system: Standard d=8                     |                            |                            |                            |  |
|   | $d_{trial} = 1$            | $d_{trial} = 2$            | $d_{trial} = 2$            | $d_{trial} = 4$                        |
| $G_{varr}$  | $56.347 \pm 3.71e-1^{***}$ | $31.960 \pm 2.98e-1^{***}$ | $32.019 \pm 2.99e-1^{***}$ | $27.344 \pm 2.69e-1$                   |
| $G_{varf}$  | $74.746 \pm 4.43e-1^{***}$ | $53.293 \pm 3.88e-1^{***}$ | $53.082 \pm 3.94e-1^{***}$ | $32.661 \pm 3.15e-1^{***}$             |
| $G_{rs}$  | $63.871 \pm 3.96e-1^{***}$ | $36.519 \pm 3.24e-1^{***}$ | $36.630 \pm 3.22e-1^{***}$ | <b><math>27.276 \pm 2.75e-1</math></b> |
| Best OOB  | $27.278 \pm 2.73e-1$       |                            |                            |  |
| Coordinate system: zPCA d=16                        |                            |                            |                            |  |
|   | $d_{trial} = 1$            | $d_{trial} = 2$            | $d_{trial} = 4$            | $d_{trial} = 5$                        |
| $G_{varr}$  | $53.309 \pm 3.71e-1^{***}$ | $37.843 \pm 3.11e-1^{***}$ | $30.420 \pm 2.71e-1^{***}$ | <b><math>29.248 \pm 2.63e-1</math></b> |
| $G_{varf}$  | $66.242 \pm 4.27e-1^{***}$ | $57.656 \pm 4.06e-1^{***}$ | $52.727 \pm 3.83e-1^{***}$ | $52.144 \pm 3.88e-1^{***}$             |
| $G_{rs}$  | $58.757 \pm 3.95e-1^{***}$ | $44.464 \pm 3.43e-1^{***}$ | $35.996 \pm 3.03e-1^{***}$ | $33.831 \pm 2.99e-1^{***}$             |
| Best OOB  | $29.457 \pm 2.65e-1$       |                            |                            |  |
| Coordinate system: zCVRP d=32                       |                            |                            |                            |  |
|   | $d_{trial} = 1$            | $d_{trial} = 2$            | $d_{trial} = 5$            | $d_{trial} = 6$                        |
| $G_{varr}$  | $53.745 \pm 3.84e-1^{***}$ | $41.887 \pm 3.44e-1^{***}$ | $32.864 \pm 3.02e-1^{**}$  | <b><math>31.703 \pm 2.95e-1</math></b> |
| $G_{varf}$  | $64.317 \pm 4.17e-1^{***}$ | $55.422 \pm 4.06e-1^{***}$ | $46.068 \pm 3.97e-1^{***}$ | $44.758 \pm 4.00e-1^{***}$             |
| $G_{rs}$  | $58.361 \pm 4.00e-1^{***}$ | $47.028 \pm 3.67e-1^{***}$ | $37.327 \pm 3.34e-1^{***}$ | $35.799 \pm 3.27e-1^{***}$             |
| Best OOB  | $31.917 \pm 2.95e-1$       |                            |                            |  |

Table B.15: Result table of the Concrete data set



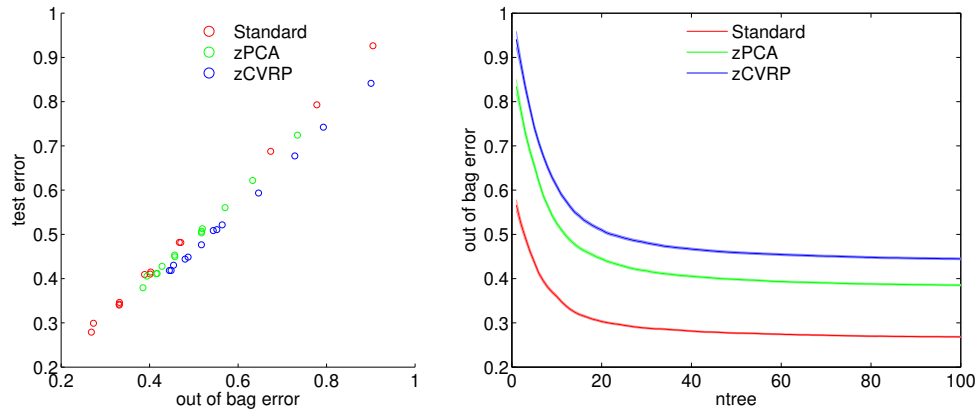
|   |                   |                   |                       |                       |
|---|-------------------|-------------------|-----------------------|-----------------------|
| Set: Housing (1000 runs) d=13, c=0, tot=506, n=51 |                   |                   |                       |                       |
| Coordinate system: Standard d=13                  |                   |                   |                       |                       |
|   | $d_{trial} = 1$   | $d_{trial} = 2$   | $d_{trial} = 3$       | $d_{trial} = 4$       |
| $G_{varr}$  | 19.490±2.93e-1*** | 12.470±2.16e-1*** | 10.903±1.89e-1**      | <b>10.303±1.68e-1</b> |
| $G_{varf}$  | 24.919±3.47e-1*** | 15.547±2.45e-1*** | 13.349±2.12e-1***     | 12.518±1.97e-1***     |
| $G_{rs}$  | 21.566±3.17e-1*** | 13.434±2.21e-1*** | 11.593±1.90e-1***     | 10.876±1.74e-1**      |
| Best OOB  | 10.595±1.76e-1    |                   |                       |                       |
| Coordinate system: zPCA d=26                      |                   |                   |                       |                       |
|   | $d_{trial} = 1$   | $d_{trial} = 2$   | $d_{trial} = 5$       | $d_{trial} = 5$       |
| $G_{varr}$  | 17.413±2.69e-1*** | 13.103±2.27e-1*** | <b>11.195±1.99e-1</b> | 11.208±2.00e-1        |
| $G_{varf}$  | 21.545±3.00e-1*** | 15.530±2.45e-1*** | 12.672±2.22e-1***     | 12.746±2.22e-1***     |
| $G_{rs}$  | 18.975±2.74e-1*** | 13.941±2.31e-1*** | 11.671±2.07e-1*       | 11.685±2.08e-1*       |
| Best OOB  | 11.325±2.01e-1    |                   |                       |                       |
| Coordinate system: zCVRP d=52                     |                   |                   |                       |                       |
|   | $d_{trial} = 1$   | $d_{trial} = 2$   | $d_{trial} = 6$       | $d_{trial} = 7$       |
| $G_{varr}$  | 18.743±2.67e-1*** | 14.851±2.31e-1*** | 12.276±2.07e-1        | <b>12.130±2.05e-1</b> |
| $G_{varf}$  | 22.489±3.01e-1*** | 17.390±2.49e-1*** | 13.458±2.14e-1***     | 13.063±2.10e-1***     |
| $G_{rs}$  | 20.215±2.80e-1*** | 15.934±2.41e-1*** | 12.755±2.11e-1*       | 12.481±2.08e-1        |
| Best OOB  | 12.296±2.07e-1    |                   |                       |                       |

Table B.16: Result table of the Housing data set



|  |                           |                           |                           |                                       |
|--|---------------------------|---------------------------|---------------------------|---------------------------------------|
| Set: Servo (1000 runs) d=4, c=0, tot=167, n=17 |                           |                           |                           |                                       |
| Coordinate system: Standard d=4                |                           |                           |                           |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 3$                       |
| $G_{varr}$                                     | $0.688 \pm 1.61e-2^{***}$ | $0.340 \pm 1.20e-2^{***}$ | $0.342 \pm 1.21e-2^{***}$ | <b><math>0.279 \pm 1.11e-2</math></b> |
| $G_{varf}$                                     | $0.926 \pm 2.04e-2^{***}$ | $0.482 \pm 1.55e-2^{***}$ | $0.482 \pm 1.55e-2^{***}$ | $0.409 \pm 1.63e-2^{***}$             |
| $G_{rs}$                                       | $0.793 \pm 1.81e-2^{***}$ | $0.410 \pm 1.40e-2^{***}$ | $0.415 \pm 1.43e-2^{***}$ | $0.346 \pm 1.33e-2^{***}$             |
| Best OOB E                                     | $0.299 \pm 1.19e-2$       |                           |                           |                                       |
| Coordinate system: zPCA d=8                    |                           |                           |                           |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 2$           | $d_{trial} = 4$                       |
| $G_{varr}$                                     | $0.560 \pm 1.46e-2^{***}$ | $0.412 \pm 1.35e-2^*$     | $0.411 \pm 1.34e-2^*$     | <b><math>0.379 \pm 1.36e-2</math></b> |
| $G_{varf}$                                     | $0.724 \pm 1.83e-2^{***}$ | $0.503 \pm 1.64e-2^{***}$ | $0.507 \pm 1.66e-2^{***}$ | $0.513 \pm 1.70e-2^{***}$             |
| $G_{rs}$                                       | $0.622 \pm 1.60e-2^{***}$ | $0.453 \pm 1.48e-2^{***}$ | $0.449 \pm 1.47e-2^{***}$ | $0.428 \pm 1.50e-2^{**}$              |
| Best OOB E                                     | $0.406 \pm 1.42e-2$       |                           |                           |                                       |
| Coordinate system: zCVRP d=16                  |                           |                           |                           |                                       |
|  | $d_{trial} = 1$           | $d_{trial} = 2$           | $d_{trial} = 4$           | $d_{trial} = 5$                       |
| $G_{varr}$                                     | $0.677 \pm 1.76e-2^{***}$ | $0.477 \pm 1.56e-2^{**}$  | $0.418 \pm 1.53e-2$       | <b><math>0.418 \pm 1.57e-2</math></b> |
| $G_{varf}$                                     | $0.842 \pm 2.04e-2^{***}$ | $0.593 \pm 1.81e-2^{***}$ | $0.511 \pm 1.76e-2^{***}$ | $0.509 \pm 1.77e-2^{***}$             |
| $G_{rs}$                                       | $0.742 \pm 1.88e-2^{***}$ | $0.521 \pm 1.67e-2^{***}$ | $0.449 \pm 1.61e-2$       | $0.444 \pm 1.63e-2$                   |
| Best OOB E                                     | $0.430 \pm 1.61e-2$       |                           |                           |                                       |

Table B.17: Result table of the Servo data set



# Appendix C

## Appendix C - Image Segmentation Examples

### C.1 EM Segmentation Results

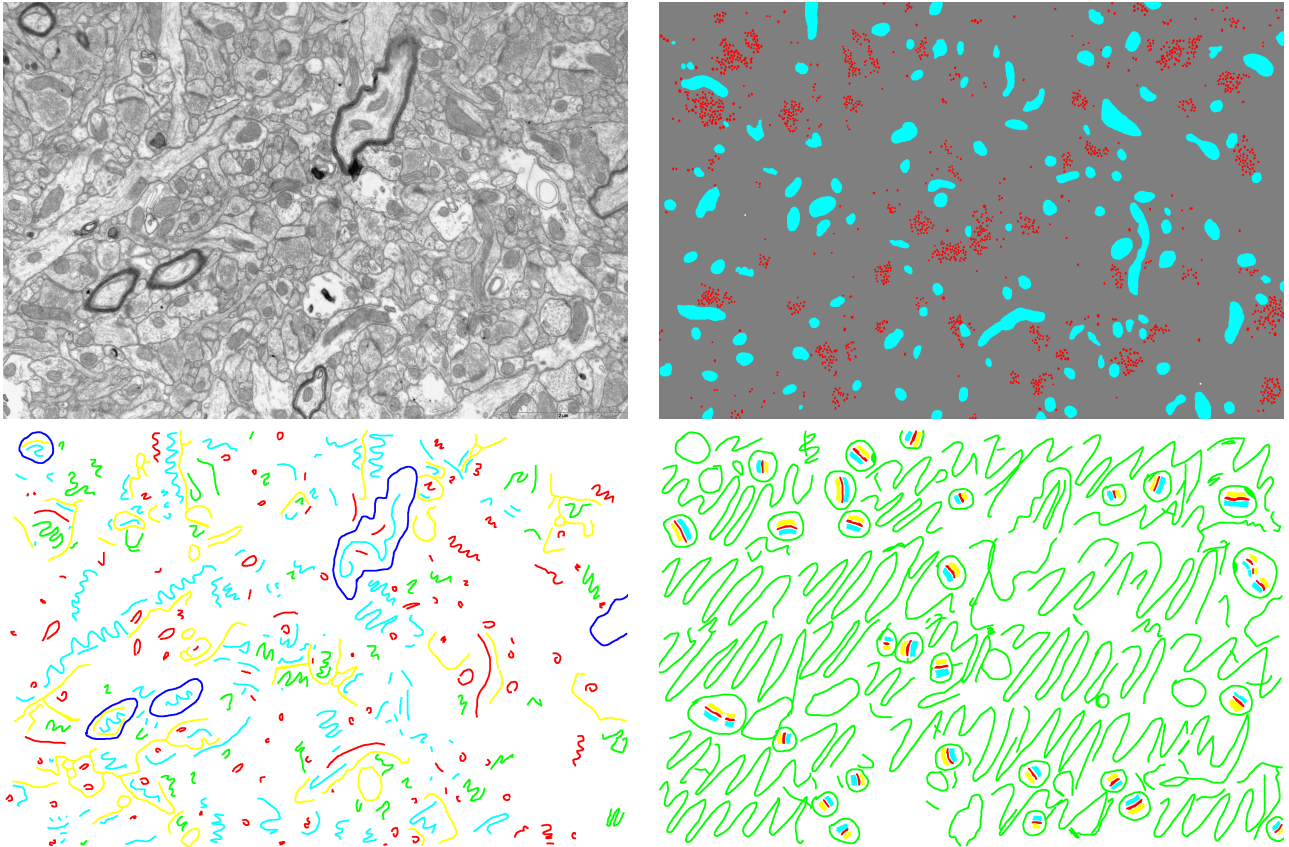


Figure C.1: Exclusive classification of image B1. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

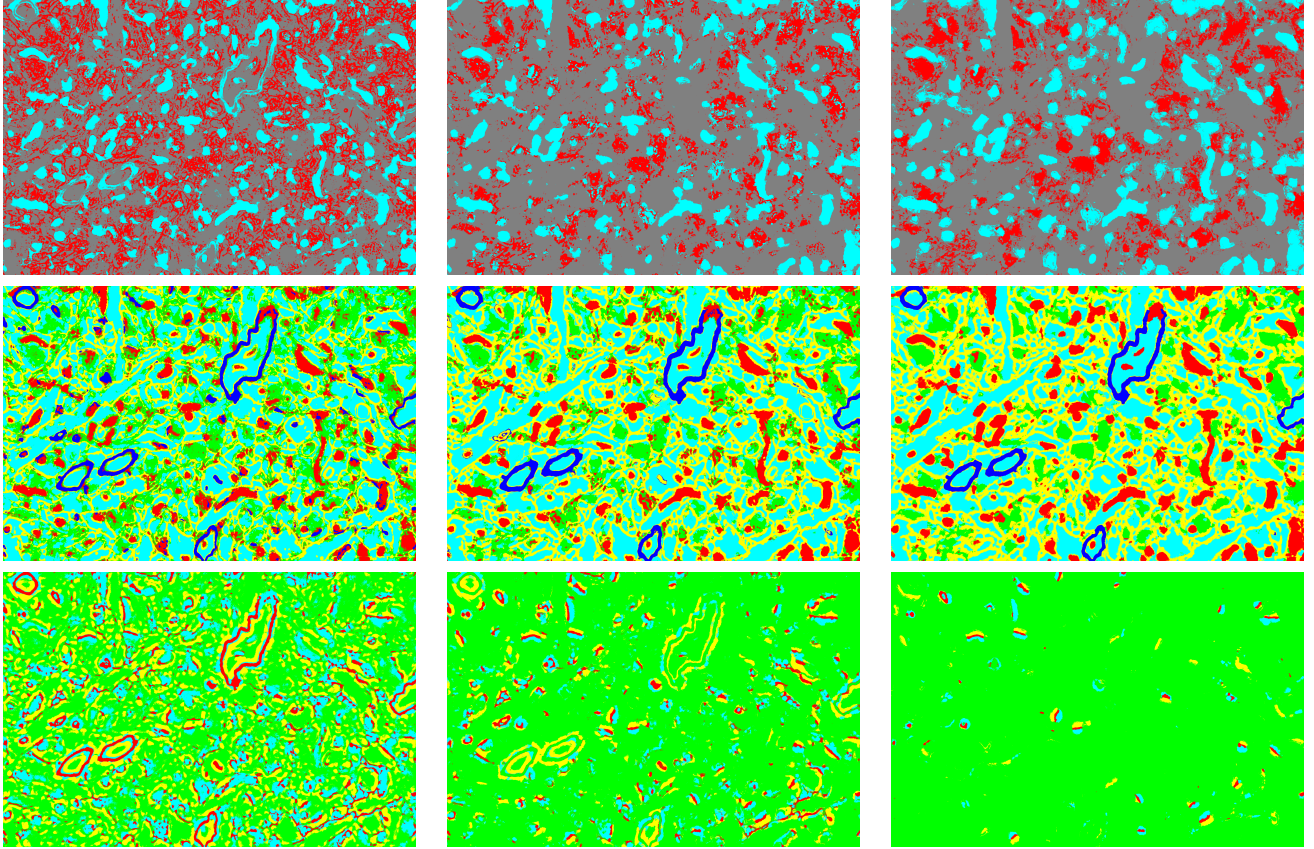


Figure C.2: Classification results for B1. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.068, 0.992, 0.348                 | 0.619, 0.756, 0.792, 0.942, 0.927       | 0.964, 0.184, 0.186, 0.204  |
| 1     | 0.123, 0.989, 0.418                 | 0.771, 0.921, 0.845, 0.940, 0.938       | 0.955, 0.392, 0.377, 0.396  |
| 2     | 0.124, 0.989, 0.409                 | 0.853, 0.942, 0.867, 0.907, 0.962       | 0.931, 0.698, 0.748, 0.742  |

Table C.1: Precision values for image B1

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.773, 0.610, 0.870                 | 0.761, 0.963, 0.863, 0.773, 0.877       | 0.548, 0.879, 0.783, 0.720  |
| 1     | 0.807, 0.762, 0.868                 | 0.849, 0.971, 0.877, 0.879, 0.918       | 0.855, 0.842, 0.573, 0.678  |
| 2     | 0.781, 0.757, 0.910                 | 0.890, 0.969, 0.907, 0.908, 0.909       | 0.979, 0.519, 0.381, 0.504  |

Table C.2: Recall values for image B1



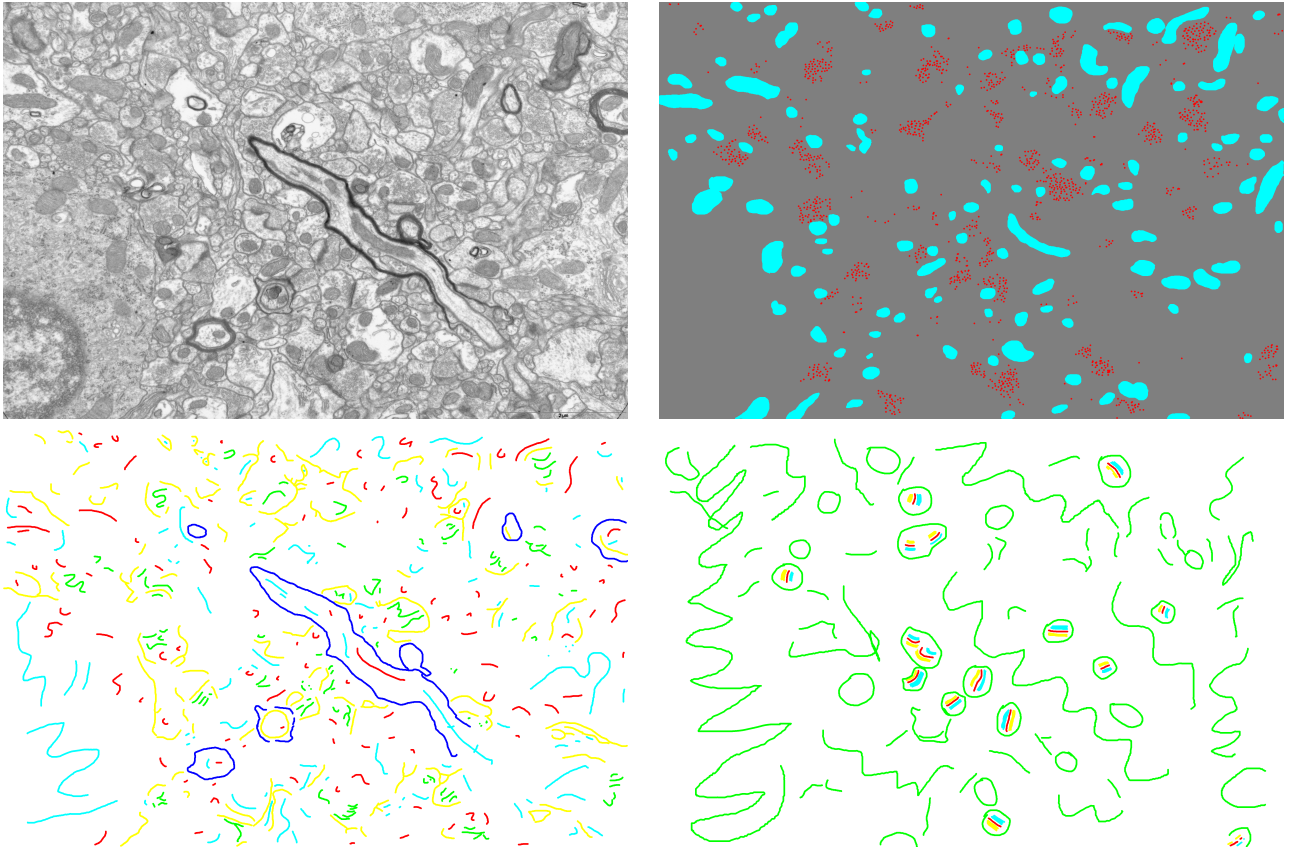


Figure C.3: Exclusive classification of image D3. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

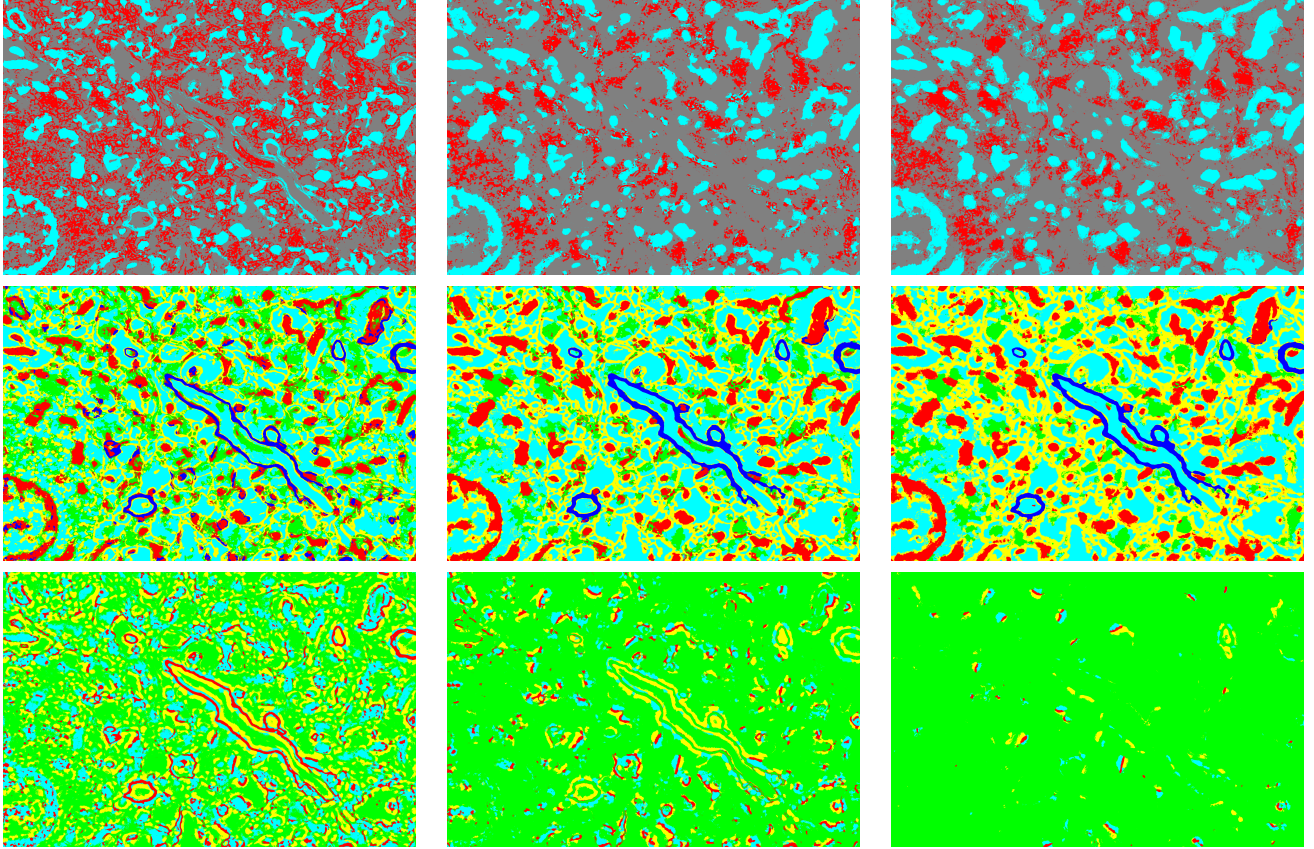


Figure C.4: Classification results for D3. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.036, 0.993, 0.323                 | 0.510, 0.850, 0.740, 0.881, 0.800       | 0.959, 0.203, 0.192, 0.174  |
| 1     | 0.070, 0.992, 0.385                 | 0.708, 0.966, 0.769, 0.891, 0.778       | 0.952, 0.282, 0.385, 0.358  |
| 2     | 0.066, 0.988, 0.355                 | 0.761, 0.964, 0.770, 0.887, 0.832       | 0.920, 0.746, 0.659, 0.699  |

Table C.3: Precision values for image D3

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.755, 0.574, 0.808                 | 0.698, 0.926, 0.835, 0.745, 0.731       | 0.505, 0.814, 0.778, 0.690  |
| 1     | 0.728, 0.734, 0.911                 | 0.702, 0.898, 0.904, 0.838, 0.808       | 0.803, 0.718, 0.754, 0.613  |
| 2     | 0.667, 0.715, 0.922                 | 0.735, 0.898, 0.955, 0.866, 0.788       | 0.975, 0.463, 0.439, 0.404  |

Table C.4: Recall values for image D3

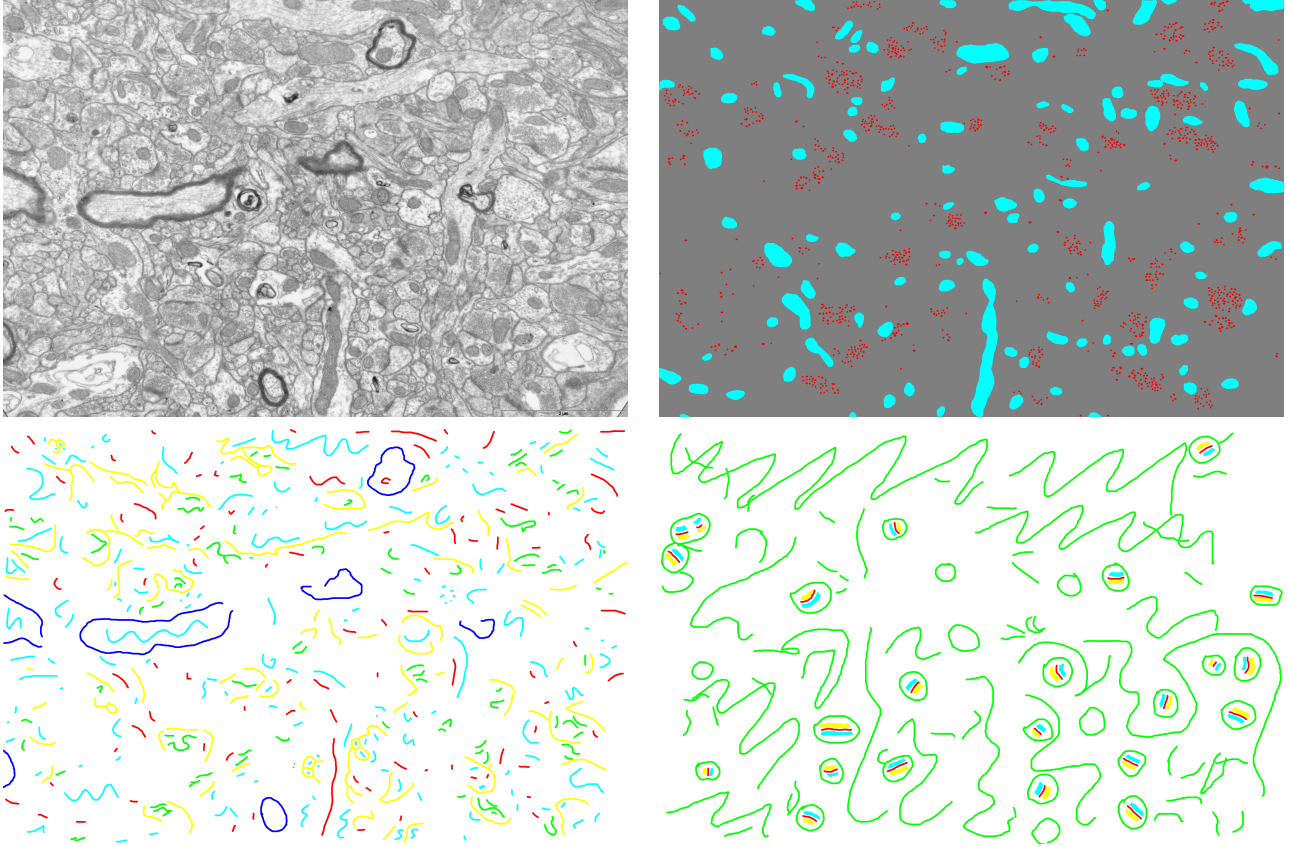


Figure C.5: Exclusive classification of image J5. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

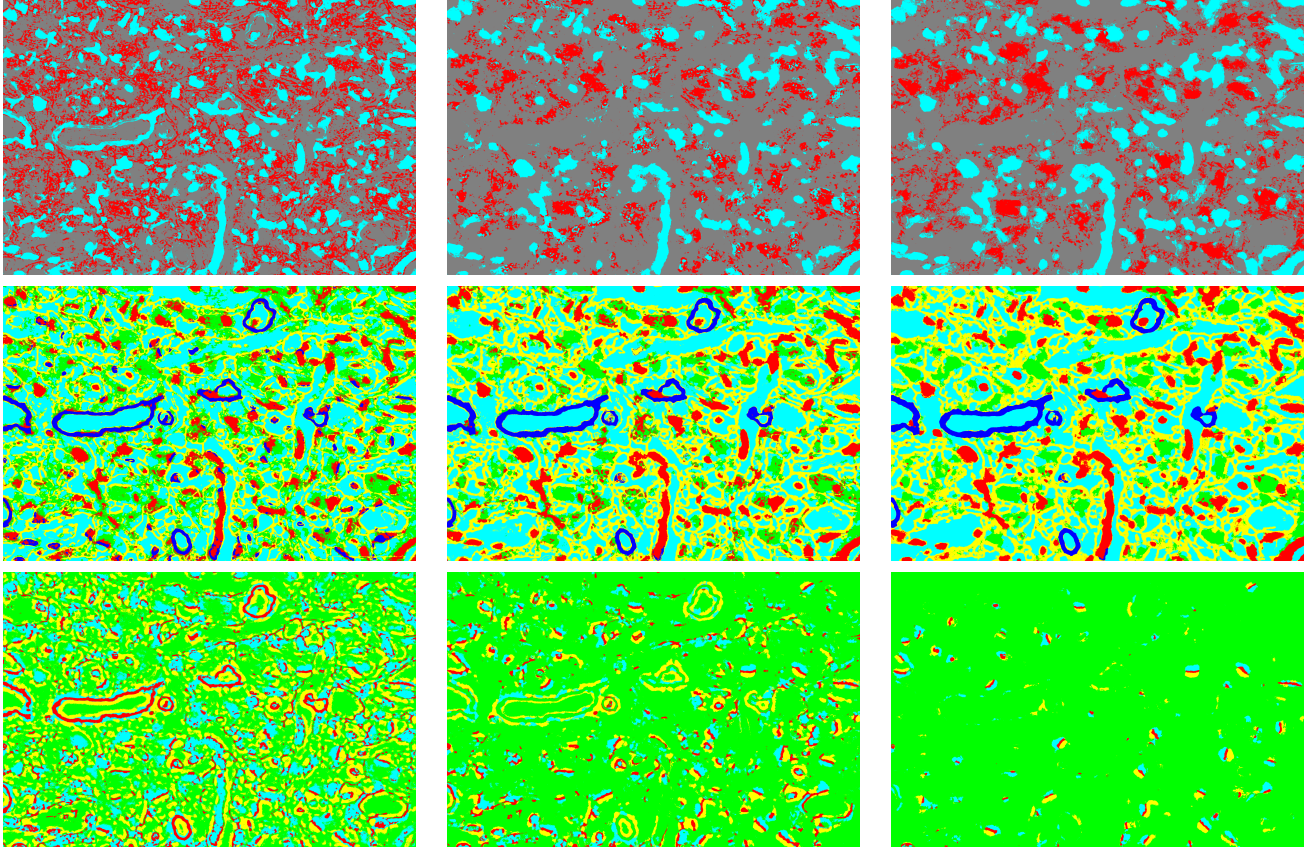


Figure C.6: Classification results for J5. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.037, 0.995, 0.314                 | 0.600, 0.821, 0.725, 0.927, 0.937       | 0.982, 0.255, 0.249, 0.271  |
| 1     | 0.070, 0.993, 0.434                 | 0.810, 0.999, 0.822, 0.960, 0.951       | 0.987, 0.361, 0.486, 0.491  |
| 2     | 0.063, 0.992, 0.403                 | 0.840, 1.000, 0.863, 0.960, 0.955       | 0.957, 0.687, 0.828, 0.781  |

Table C.5: Precision values for image J5

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.719, 0.608, 0.867                 | 0.718, 0.999, 0.891, 0.769, 0.897       | 0.506, 0.957, 0.786, 0.890  |
| 1     | 0.785, 0.773, 0.880                 | 0.830, 0.999, 0.949, 0.897, 0.959       | 0.793, 0.973, 0.836, 0.879  |
| 2     | 0.778, 0.741, 0.911                 | 0.866, 1.000, 0.947, 0.909, 0.967       | 0.964, 0.796, 0.664, 0.801  |

Table C.6: Recall values for image J5

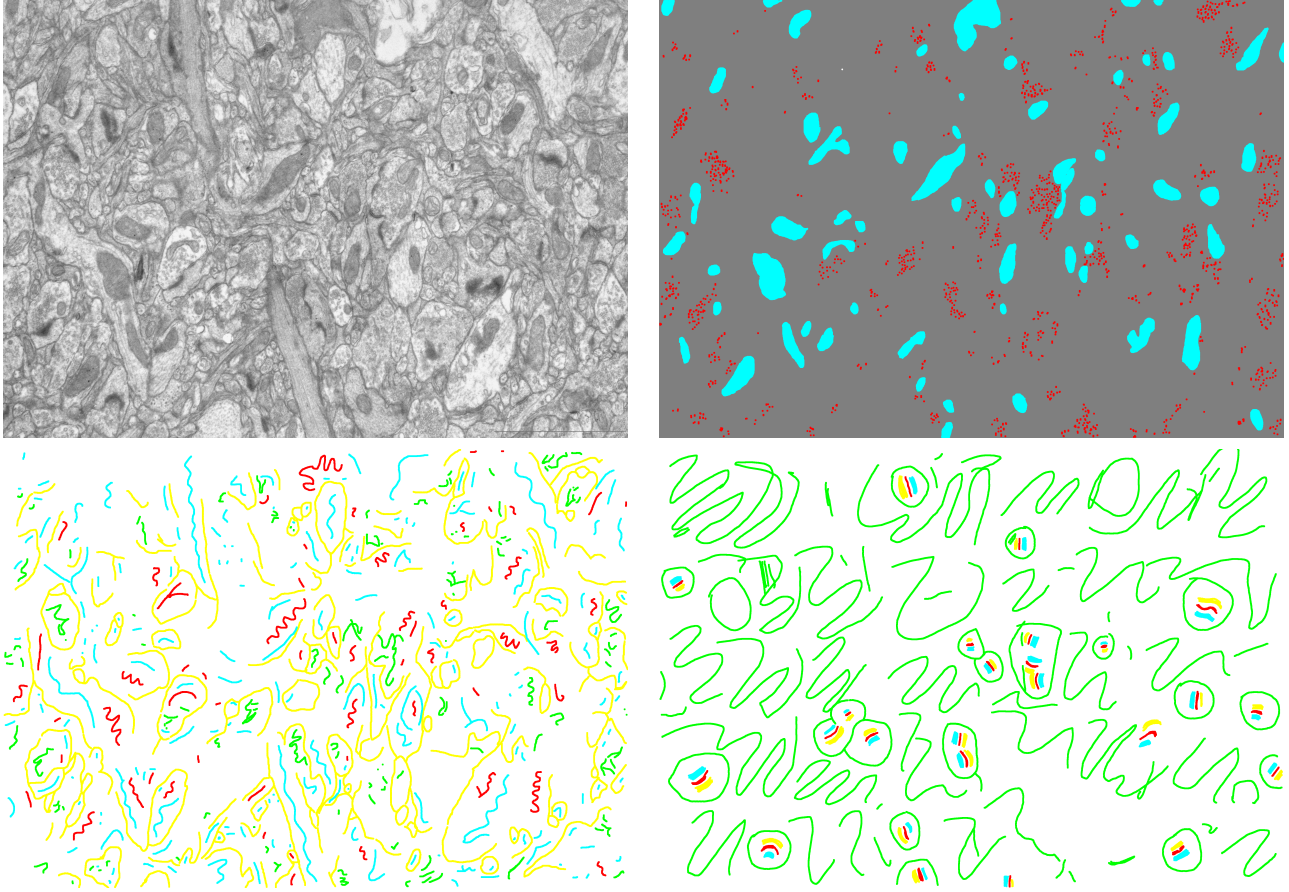


Figure C.7: Exclusive classification of image C2. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

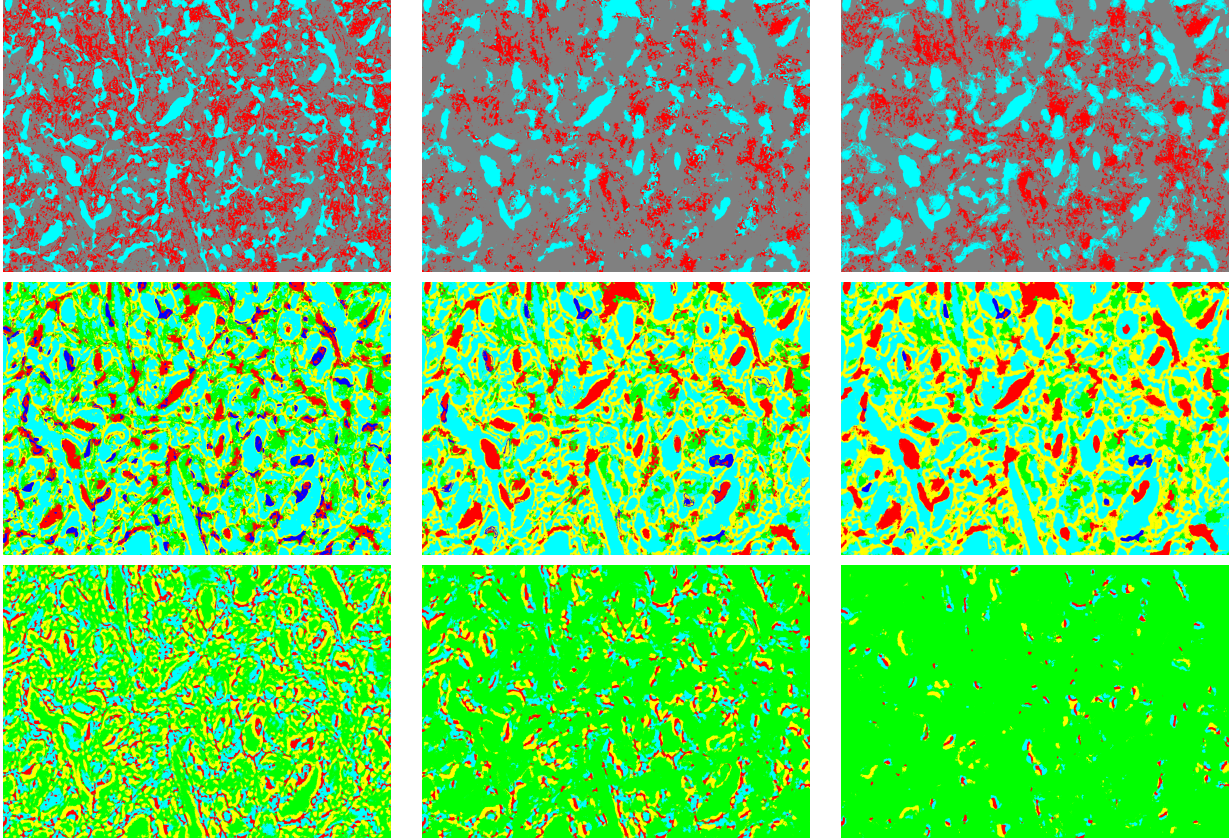


Figure C.8: Classification results for C2. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.051, 0.985, 0.227                 | 0.402, 0.000, 0.551, 0.931, 0.731       | 0.983, 0.246, 0.183, 0.175  |
| 1     | 0.080, 0.982, 0.324                 | 0.575, 0.000, 0.664, 0.955, 0.698       | 0.972, 0.282, 0.351, 0.282  |
| 2     | 0.073, 0.979, 0.292                 | 0.654, 0.000, 0.619, 0.952, 0.693       | 0.934, 0.579, 0.698, 0.625  |

Table C.7: Precision values for image C2

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.760, 0.592, 0.722                 | 0.570, 0.000, 0.618, 0.633, 0.895       | 0.441, 0.981, 0.832, 0.634  |
| 1     | 0.640, 0.760, 0.786                 | 0.537, 0.000, 0.797, 0.800, 0.926       | 0.736, 0.783, 0.855, 0.566  |
| 2     | 0.674, 0.714, 0.805                 | 0.618, 0.000, 0.807, 0.790, 0.912       | 0.955, 0.601, 0.622, 0.476  |

Table C.8: Recall values for image C2

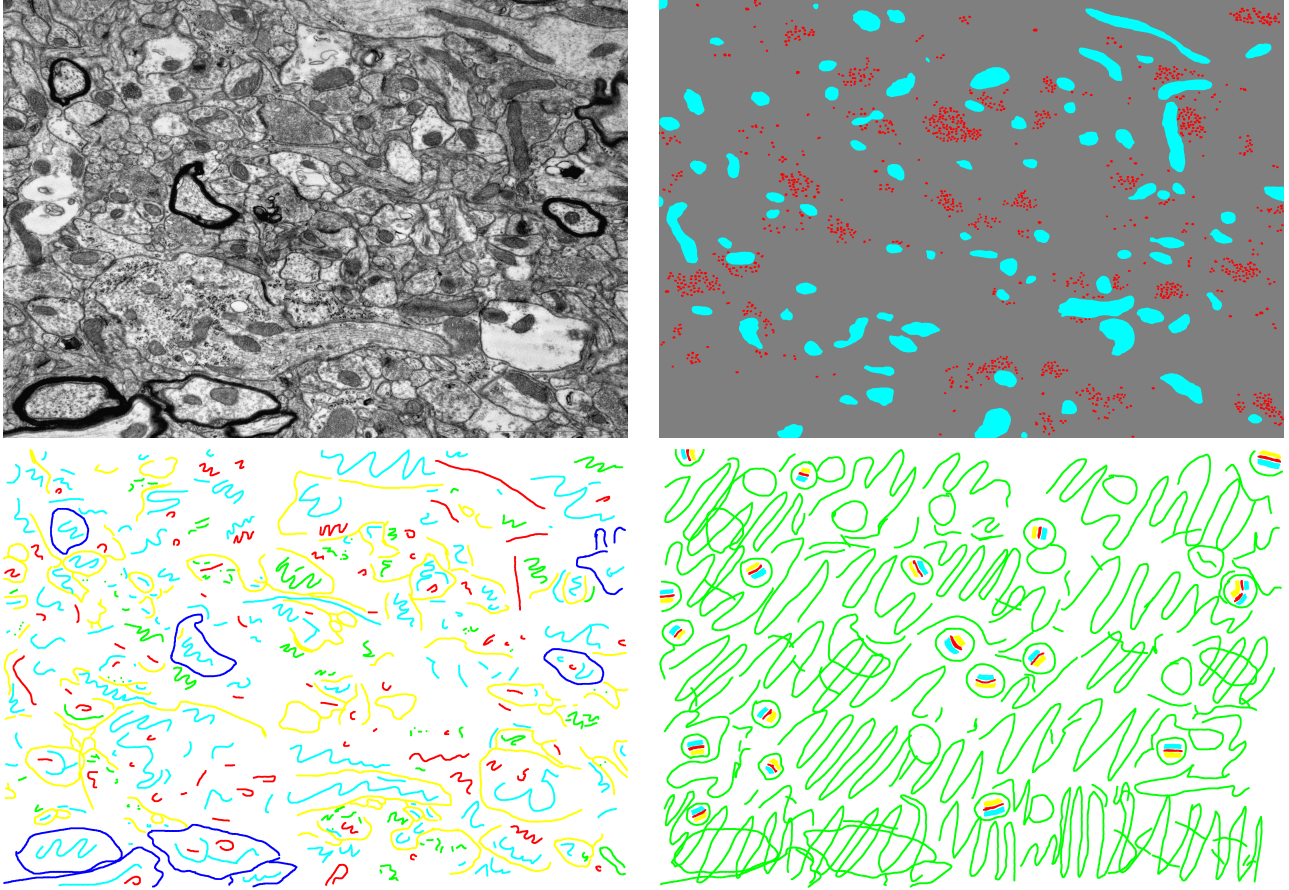


Figure C.9: Exclusive classification of image K4. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

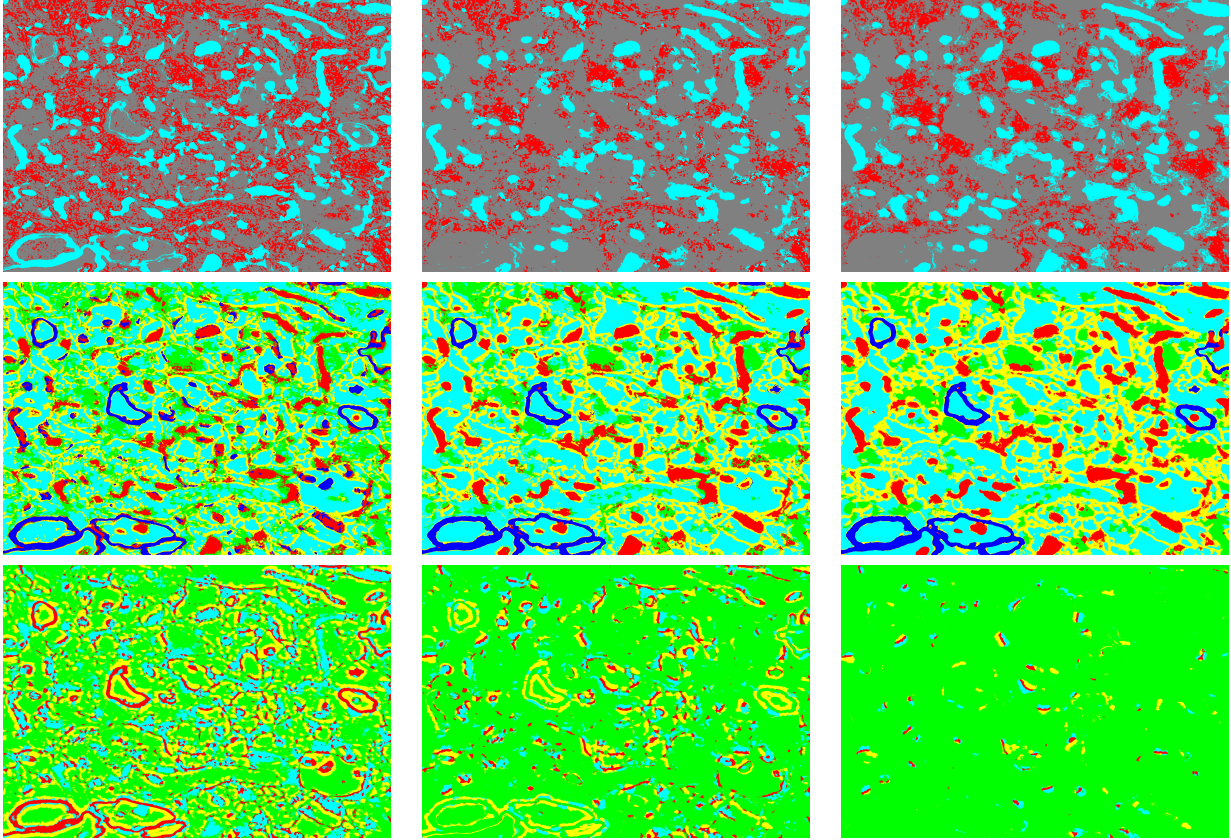


Figure C.10: Classification results for K4. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.063, 0.992, 0.350                 | 0.464, 0.723, 0.750, 0.864, 0.897       | 0.989, 0.104, 0.103, 0.135  |
| 1     | 0.102, 0.989, 0.457                 | 0.664, 0.992, 0.798, 0.897, 0.892       | 0.987, 0.303, 0.195, 0.282  |
| 2     | 0.091, 0.989, 0.422                 | 0.722, 0.993, 0.790, 0.891, 0.921       | 0.970, 0.788, 0.482, 0.688  |

Table C.9: Precision values for image K4

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.767, 0.616, 0.879                 | 0.696, 0.977, 0.834, 0.739, 0.798       | 0.484, 0.957, 0.780, 0.749  |
| 1     | 0.695, 0.775, 0.919                 | 0.705, 0.973, 0.940, 0.858, 0.875       | 0.817, 0.970, 0.713, 0.695  |
| 2     | 0.654, 0.744, 0.961                 | 0.698, 0.965, 0.957, 0.887, 0.878       | 0.977, 0.791, 0.411, 0.611  |

Table C.10: Recall values for image K4



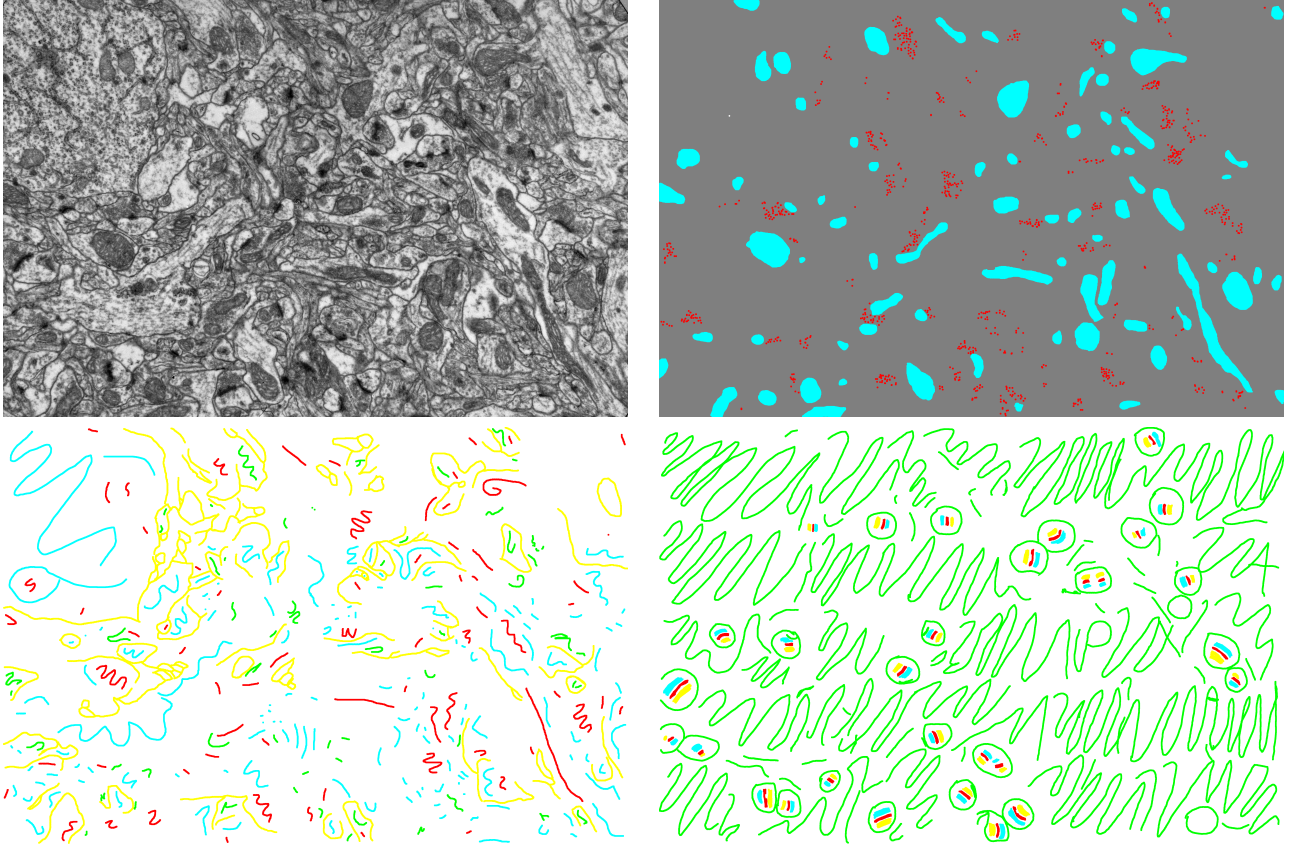


Figure C.11: Exclusive classification of image C5. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

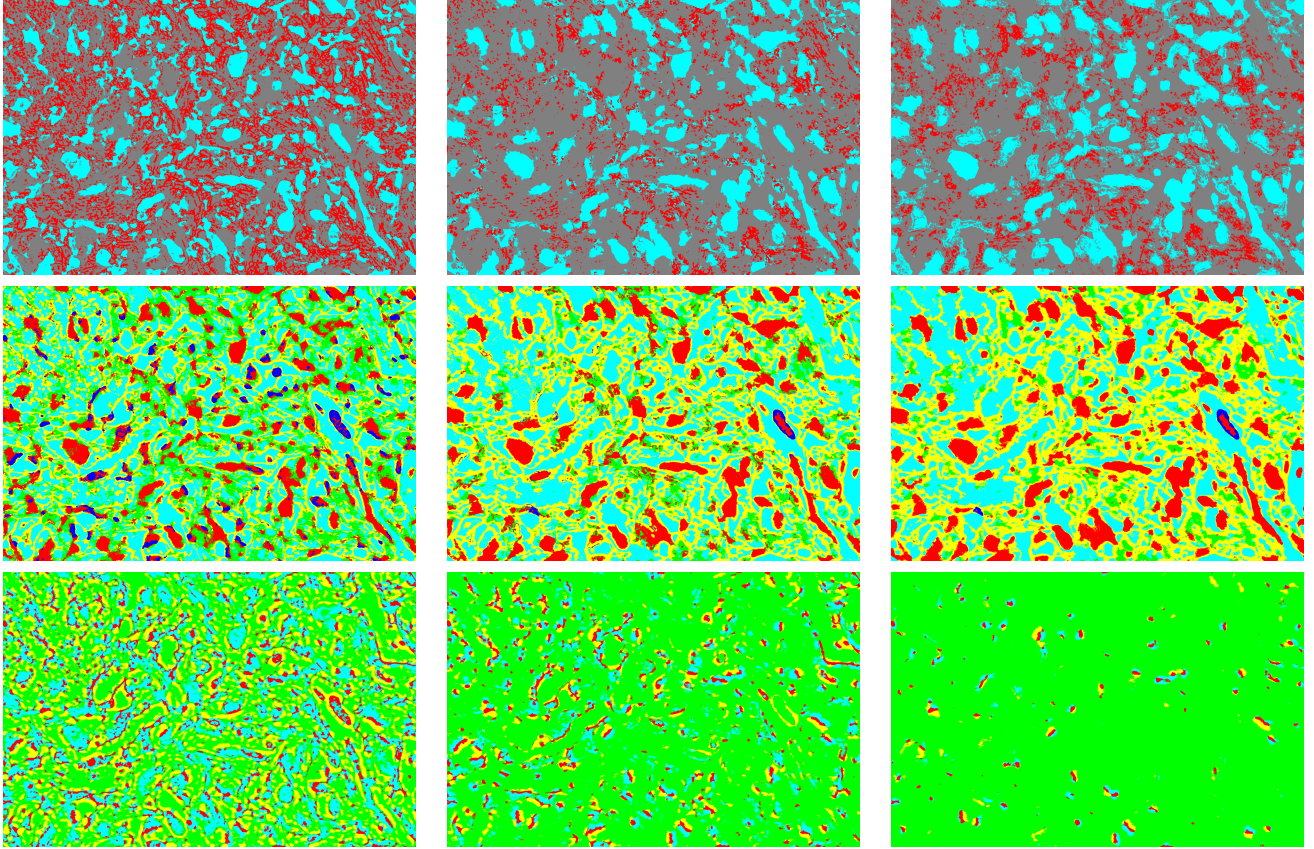


Figure C.12: Classification results for C5. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.016, 0.995, 0.284                 | 0.211, 0.000, 0.595, 0.894, 0.843       | 0.993, 0.232, 0.130, 0.141  |
| 1     | 0.027, 0.994, 0.332                 | 0.467, 0.000, 0.656, 0.897, 0.890       | 0.986, 0.360, 0.332, 0.314  |
| 2     | 0.020, 0.992, 0.312                 | 0.424, 0.000, 0.630, 0.892, 0.920       | 0.950, 0.682, 0.734, 0.654  |

Table C.11: Precision values for image C5

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.435, 0.561, 0.890                 | 0.312, 0.000, 0.839, 0.688, 0.746       | 0.452, 0.972, 0.743, 0.803  |
| 1     | 0.319, 0.732, 0.960                 | 0.314, 0.000, 0.948, 0.870, 0.824       | 0.801, 0.933, 0.783, 0.799  |
| 2     | 0.224, 0.717, 0.964                 | 0.231, 0.000, 0.966, 0.880, 0.825       | 0.975, 0.735, 0.533, 0.431  |

Table C.12: Recall values for image C5

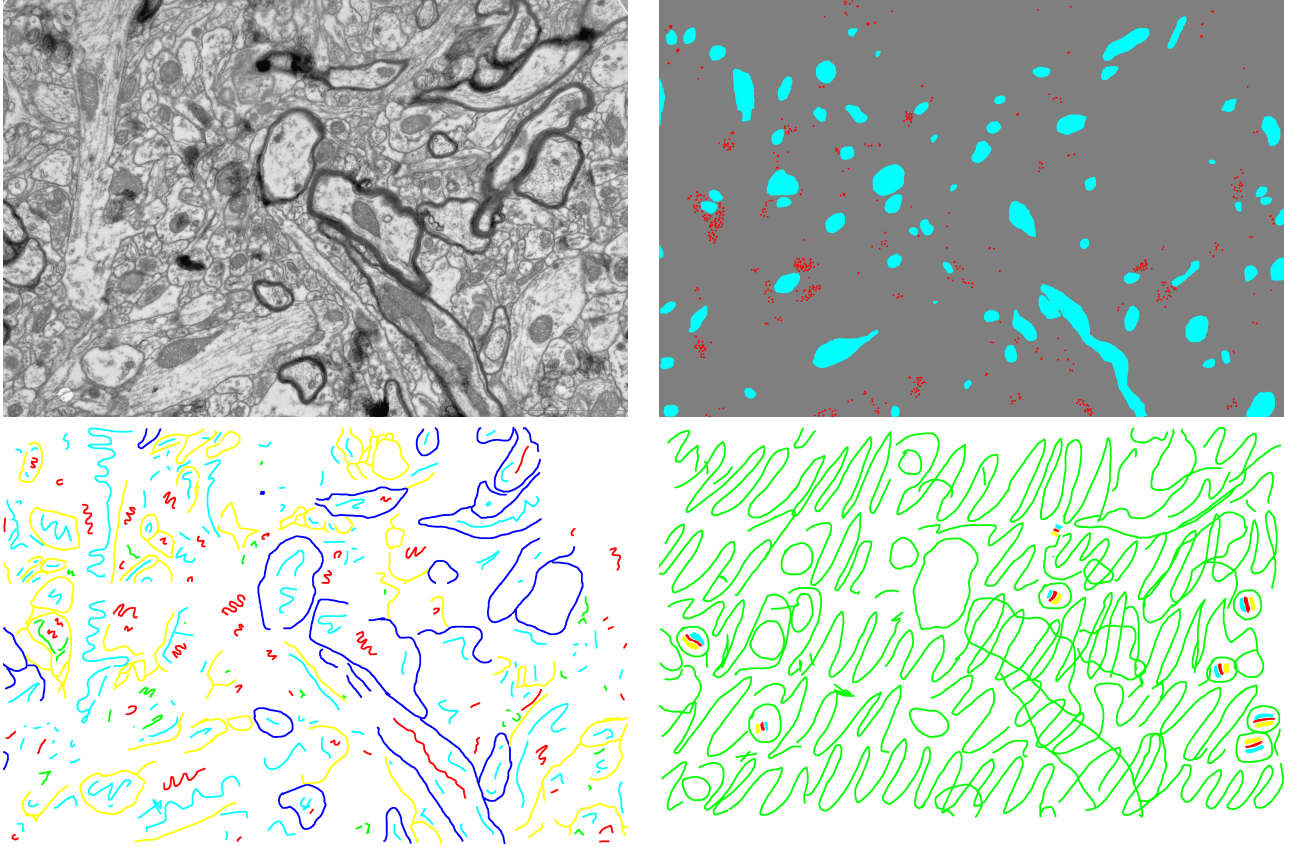


Figure C.13: Exclusive classification of image D1. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

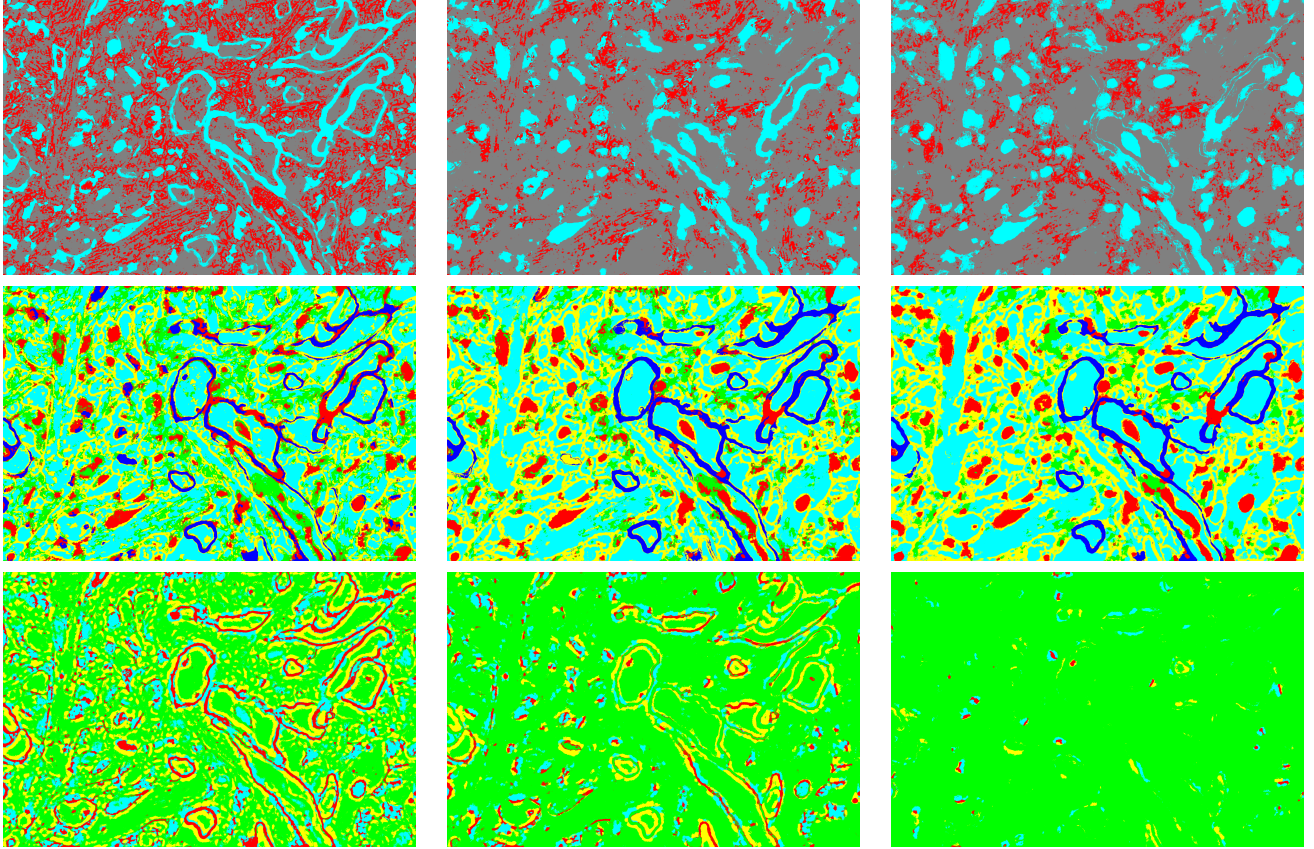


Figure C.14: Classification results for D1. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.015, 0.991, 0.228                 | 0.107, 0.948, 0.553, 0.762, 0.897       | 0.993, 0.040, 0.049, 0.051  |
| 1     | 0.023, 0.990, 0.365                 | 0.218, 0.982, 0.745, 0.787, 0.903       | 0.992, 0.097, 0.103, 0.094  |
| 2     | 0.024, 0.987, 0.361                 | 0.252, 0.987, 0.712, 0.785, 0.889       | 0.982, 0.670, 0.314, 0.323  |

Table C.13: Precision values for image D1

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.586, 0.604, 0.651                 | 0.427, 0.674, 0.673, 0.728, 0.792       | 0.473, 0.860, 0.688, 0.650  |
| 1     | 0.438, 0.782, 0.824                 | 0.435, 0.743, 0.860, 0.824, 0.887       | 0.780, 0.793, 0.640, 0.644  |
| 2     | 0.416, 0.785, 0.840                 | 0.422, 0.719, 0.865, 0.826, 0.900       | 0.979, 0.539, 0.272, 0.499  |

Table C.14: Recall values for image D1

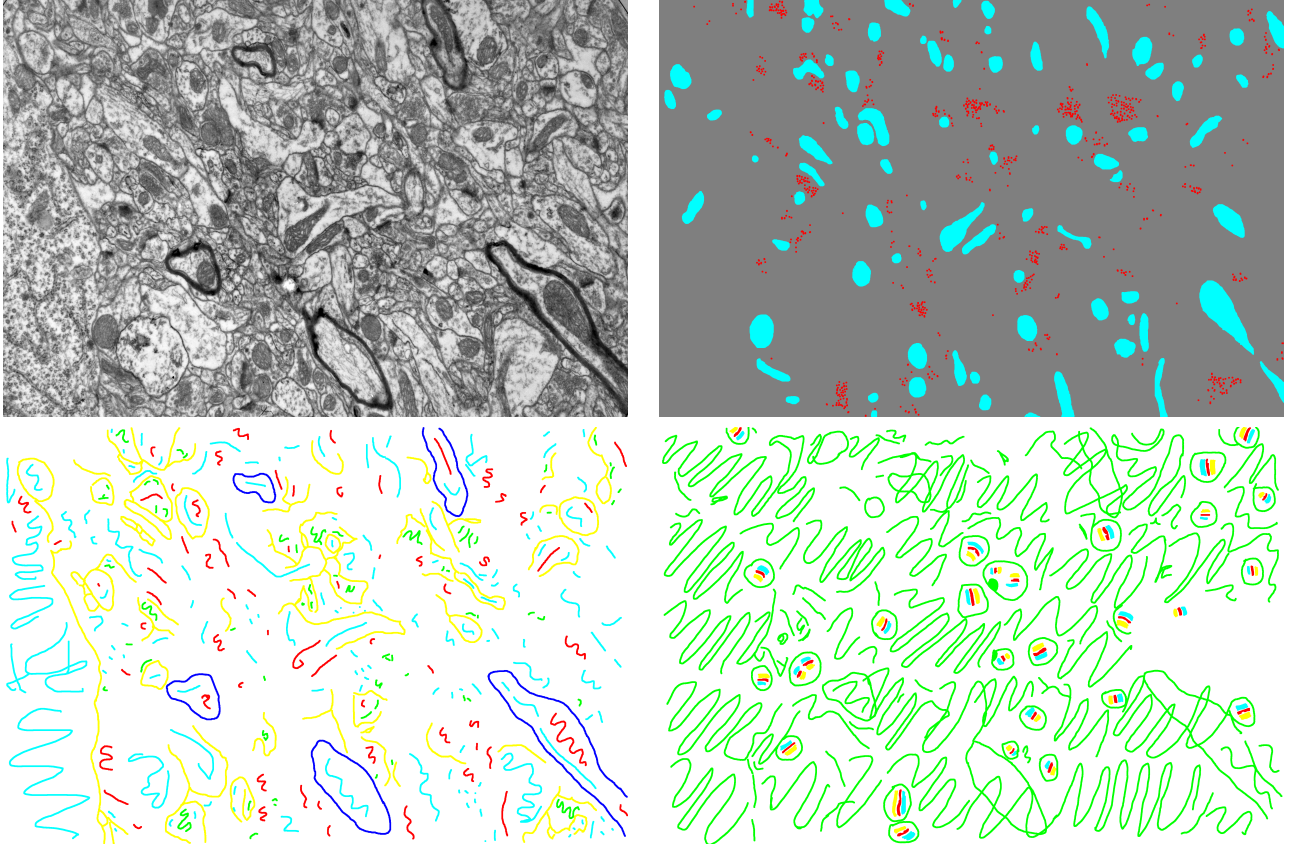


Figure C.15: Exclusive classification of image G1. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

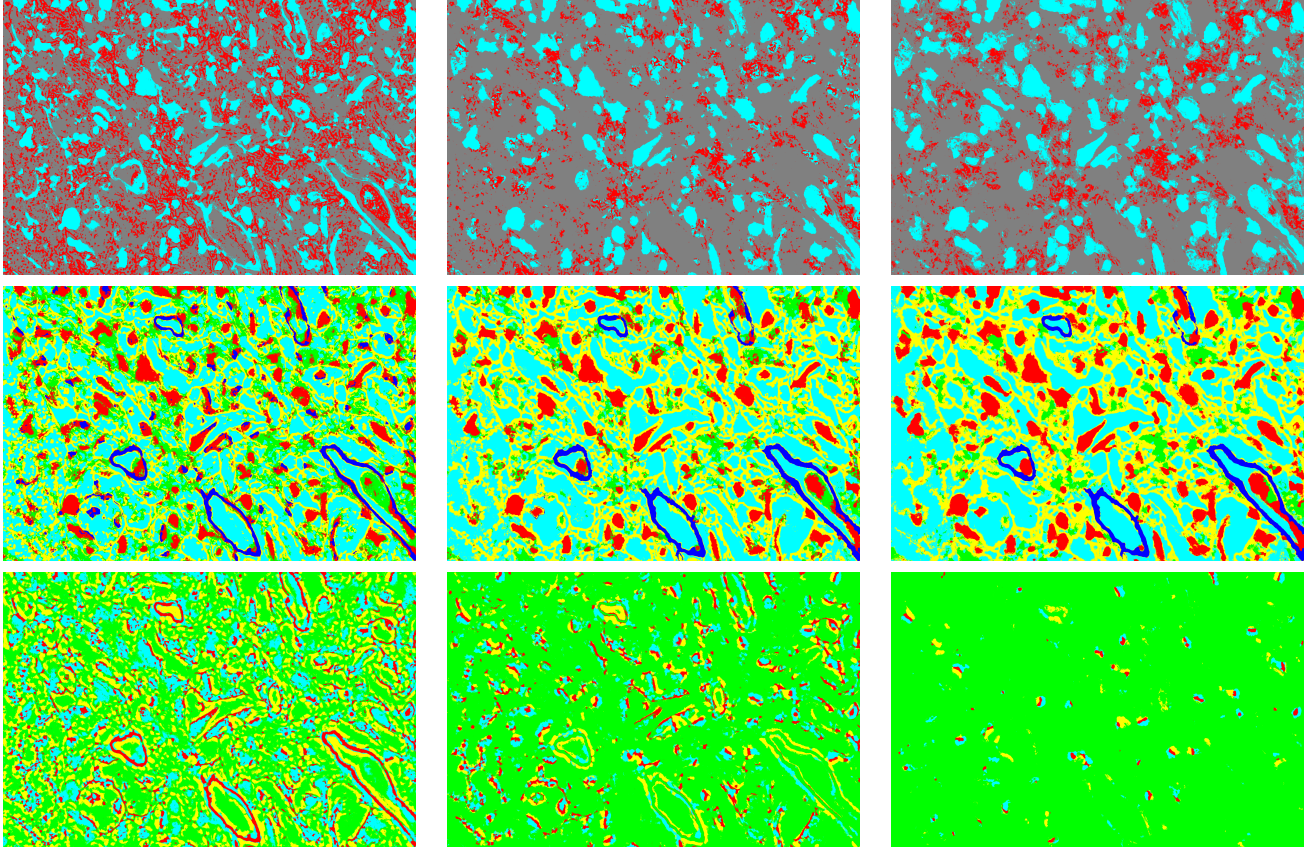


Figure C.16: Classification results for G1. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.018, 0.994, 0.278                 | 0.152, 0.770, 0.587, 0.829, 0.888       | 0.989, 0.143, 0.106, 0.133  |
| 1     | 0.033, 0.994, 0.378                 | 0.365, 0.994, 0.643, 0.867, 0.913       | 0.987, 0.267, 0.290, 0.290  |
| 2     | 0.026, 0.993, 0.338                 | 0.344, 0.995, 0.592, 0.884, 0.919       | 0.955, 0.678, 0.661, 0.593  |

Table C.15: Precision values for image G1

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.416, 0.603, 0.828                 | 0.278, 0.943, 0.827, 0.695, 0.735       | 0.452, 0.939, 0.694, 0.817  |
| 1     | 0.342, 0.782, 0.917                 | 0.301, 0.936, 0.919, 0.838, 0.850       | 0.803, 0.914, 0.740, 0.794  |
| 2     | 0.260, 0.758, 0.947                 | 0.250, 0.930, 0.965, 0.806, 0.870       | 0.976, 0.620, 0.480, 0.442  |

Table C.16: Recall values for image G1

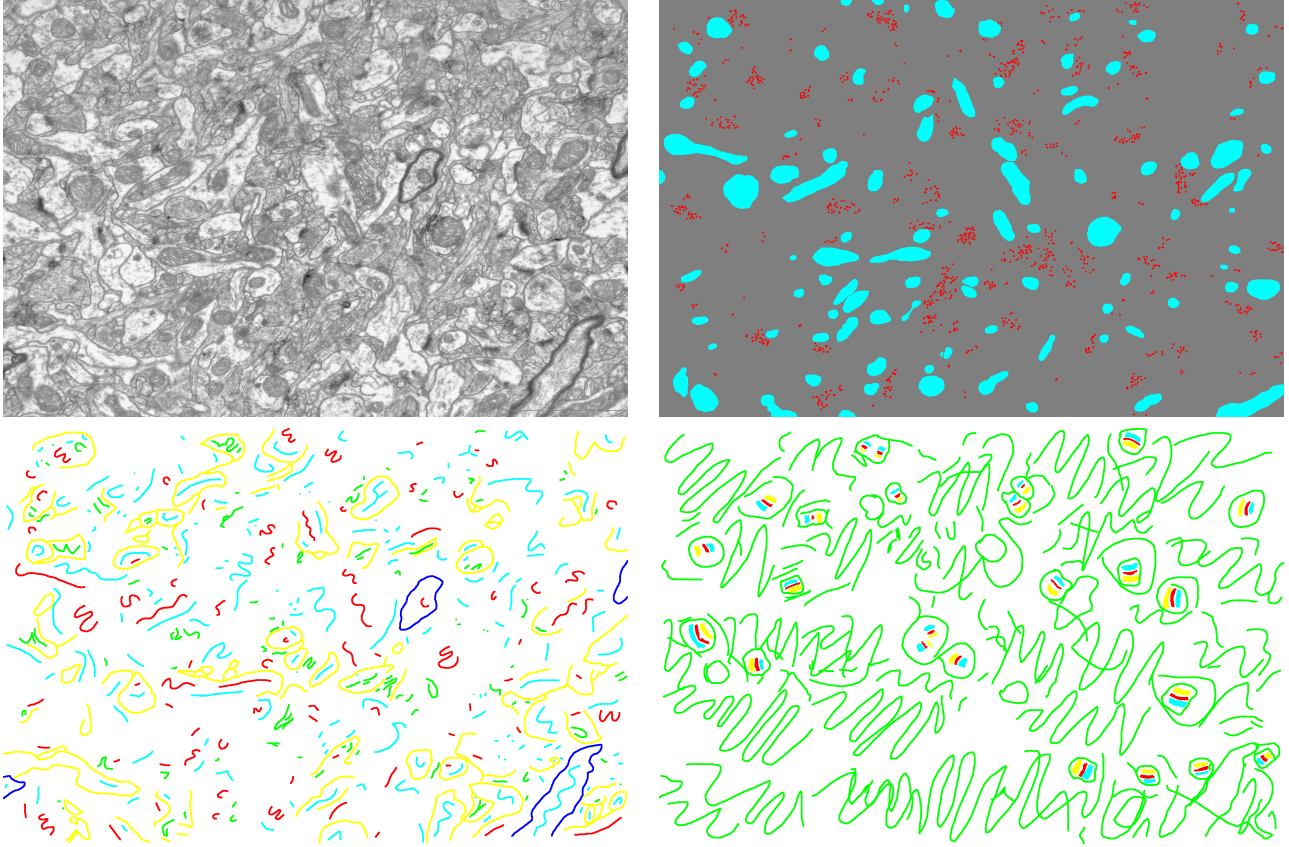


Figure C.17: Exclusive classification of image H5. Top left: Raw image; top right: vesicle-mitochondria labels; bottom left: sparse organelle labels; bottom right: synapse labels

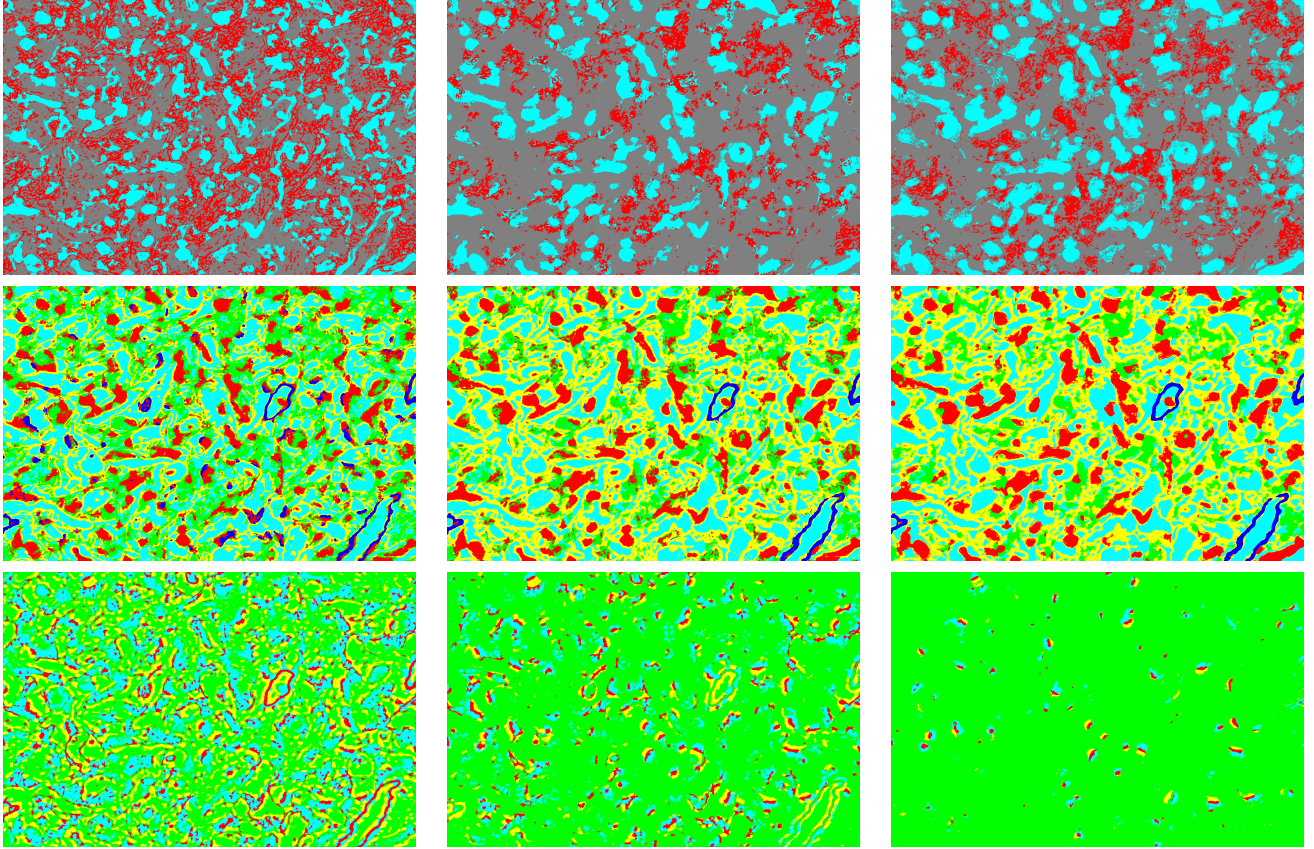


Figure C.18: Classification results for H5. Classification iterations from left to right. Top row: vesicle-mitochondria; middle row: sparse organelles; bottom row: synapses

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.030, 0.988, 0.299                 | 0.310, 0.592, 0.624, 0.918, 0.849       | 0.980, 0.183, 0.156, 0.135  |
| 1     | 0.052, 0.989, 0.381                 | 0.488, 0.984, 0.681, 0.922, 0.919       | 0.971, 0.344, 0.370, 0.254  |
| 2     | 0.044, 0.985, 0.372                 | 0.519, 0.987, 0.671, 0.917, 0.912       | 0.942, 0.675, 0.770, 0.521  |

Table C.17: Precision values for image H5

| Iter. | Vesicle-Mitochondria<br>ves,oth,mit | Sparse Organelle<br>ves,mye,mit,mem,int | Synapses<br>oth,syn,pos,pre |
|-------|-------------------------------------|---|-----------------------------|
| 0     | 0.603, 0.562, 0.803                 | 0.505, 0.947, 0.785, 0.653, 0.883       | 0.494, 0.892, 0.782, 0.835  |
| 1     | 0.579, 0.722, 0.889                 | 0.539, 0.985, 0.889, 0.821, 0.912       | 0.822, 0.833, 0.686, 0.724  |
| 2     | 0.495, 0.706, 0.905                 | 0.489, 0.987, 0.911, 0.826, 0.924       | 0.973, 0.598, 0.332, 0.454  |

Table C.18: Recall values for image H5

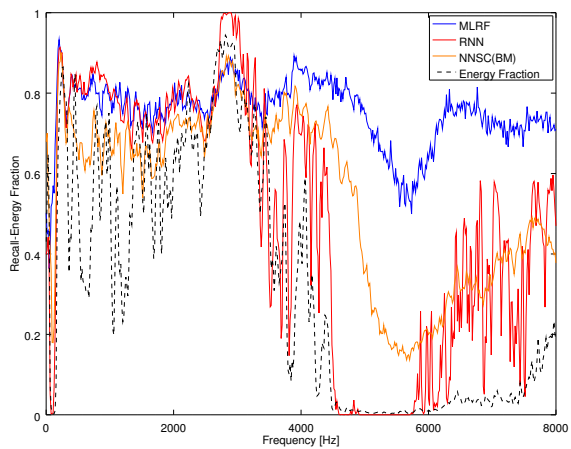
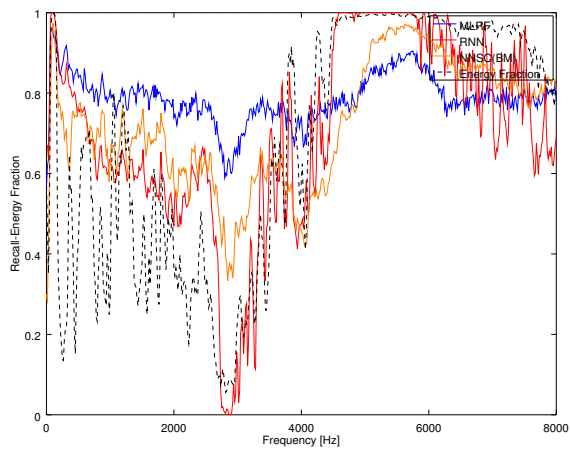
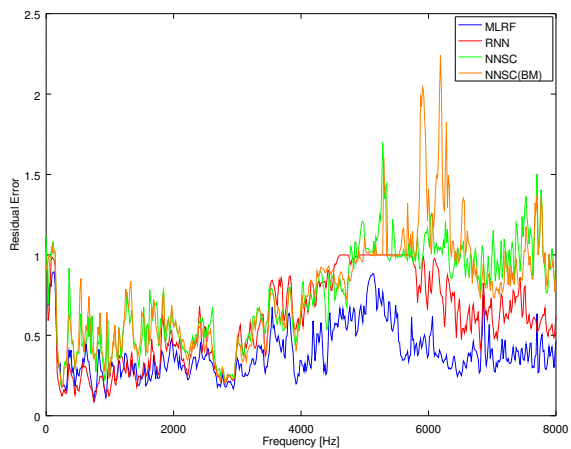
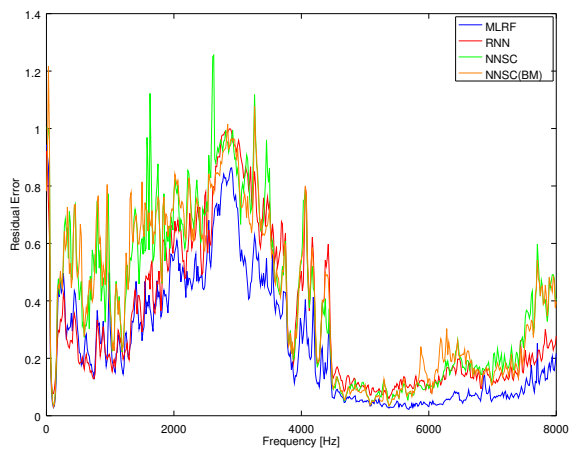


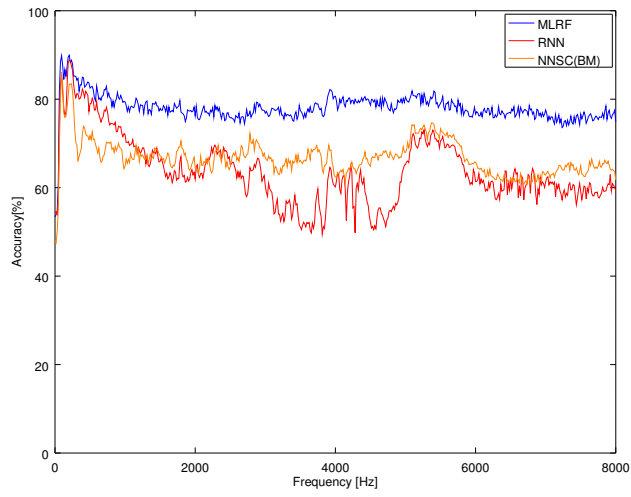
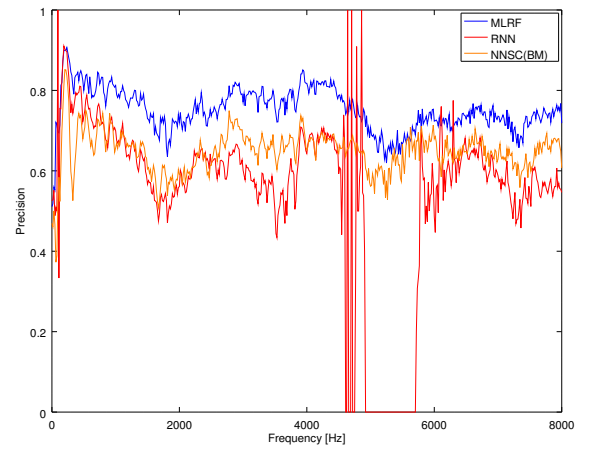
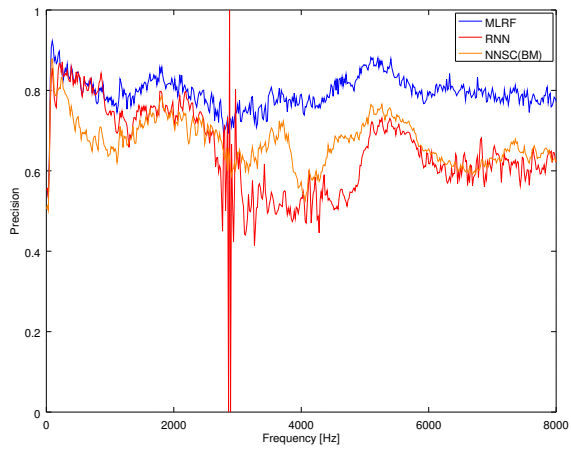
# Appendix D

## Appendix D - Random Forest Source Separation

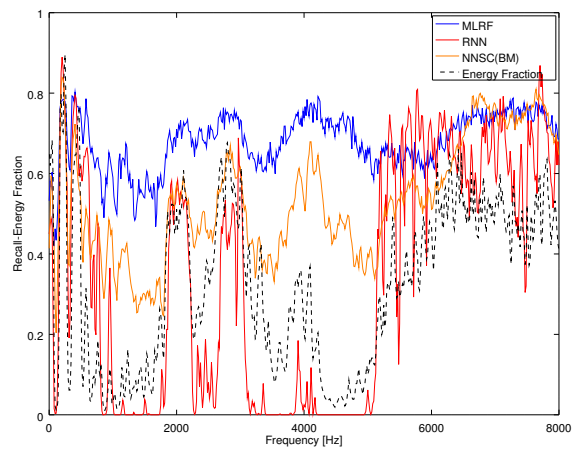
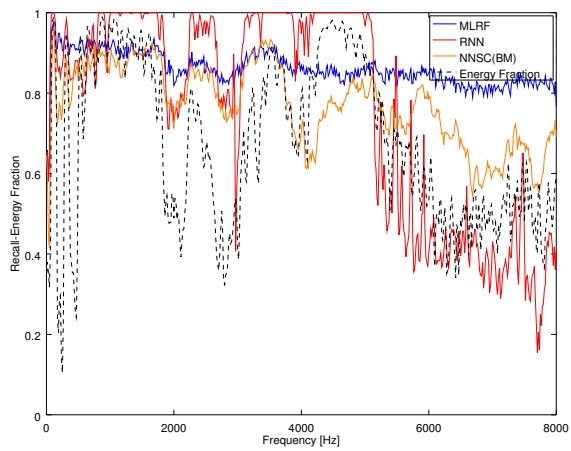
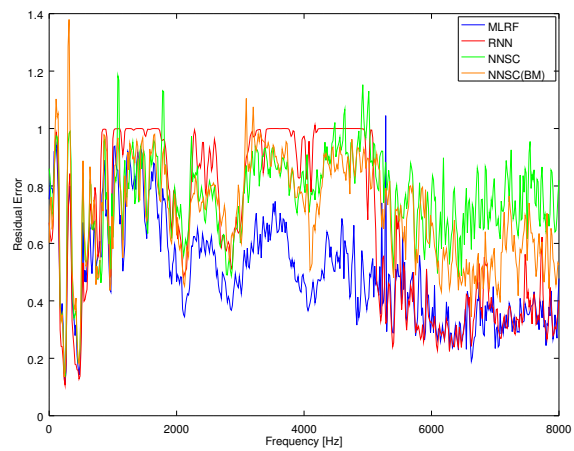
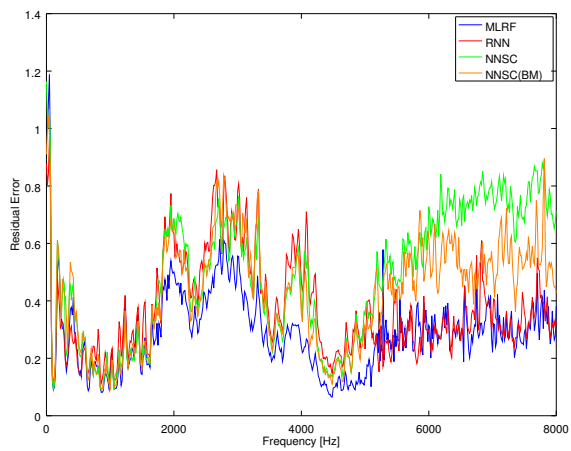
### D.1 Random Forest Source Separation Frequency Analysis

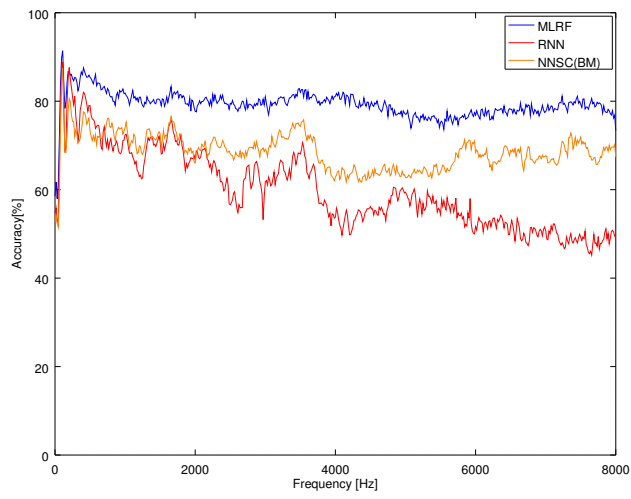
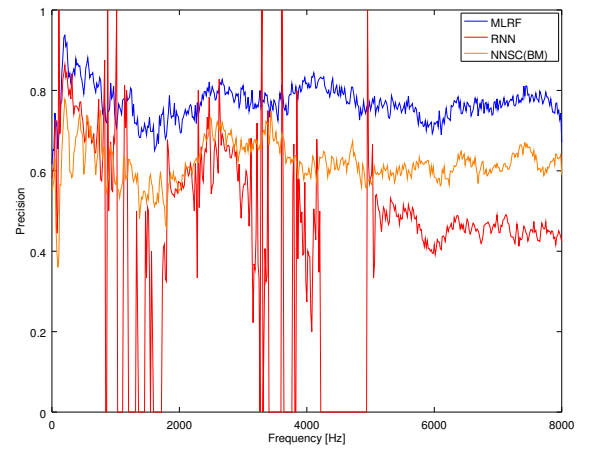
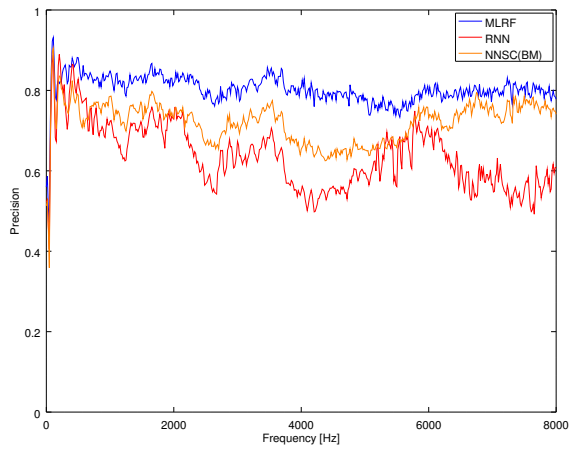
# Male-Female: Pair 1



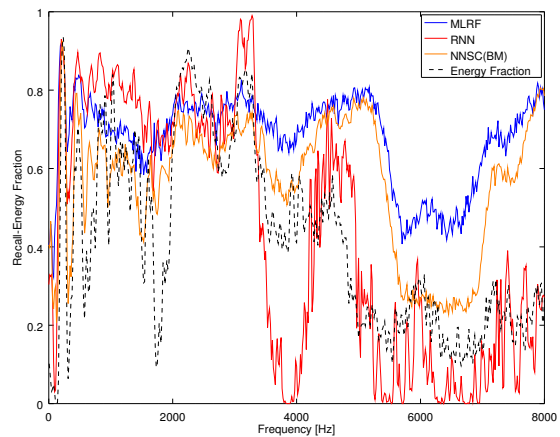
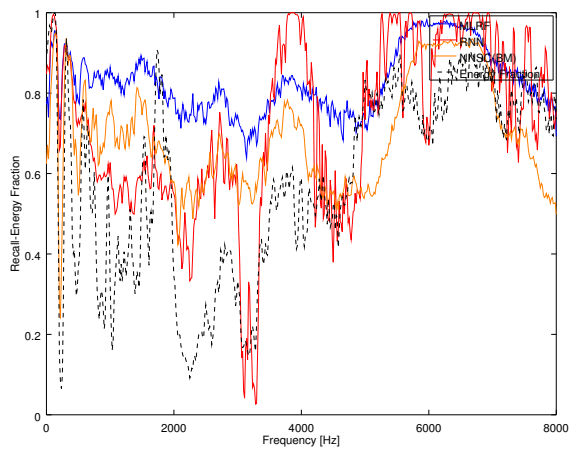
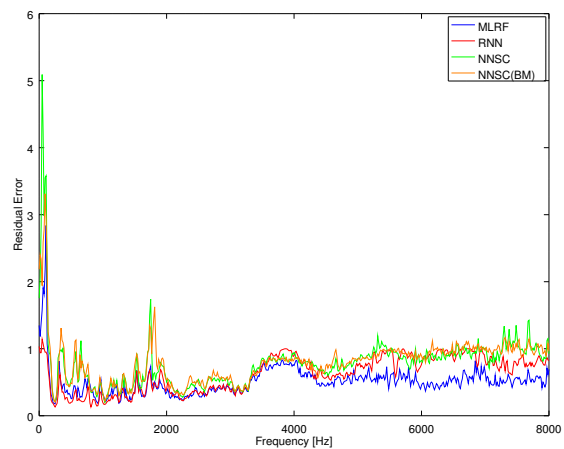
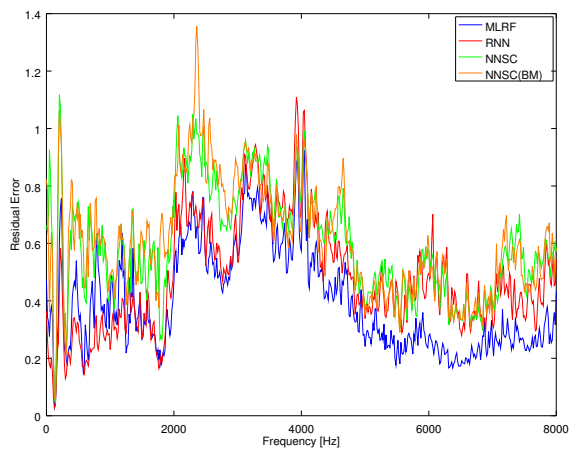


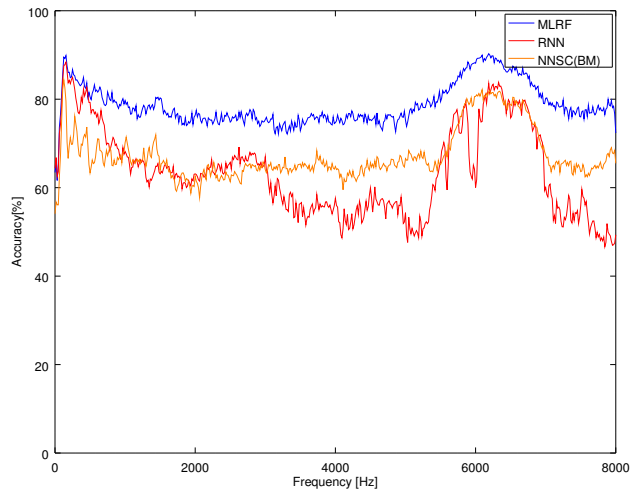
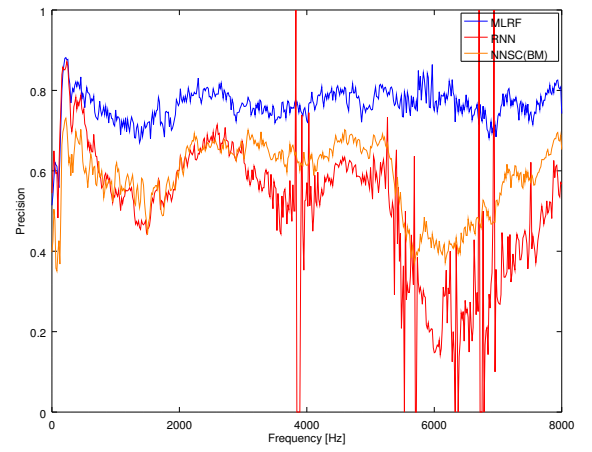
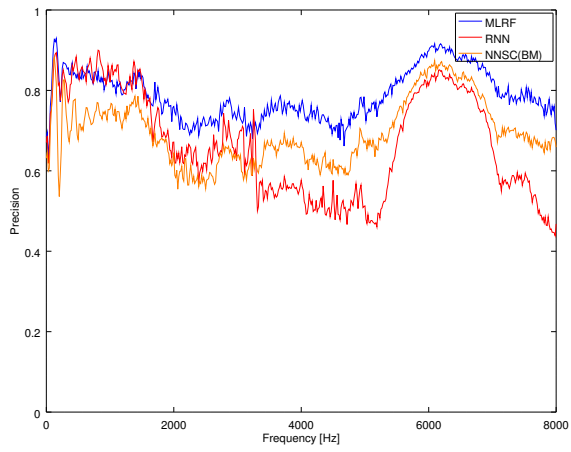
## Male-Female: Pair 2



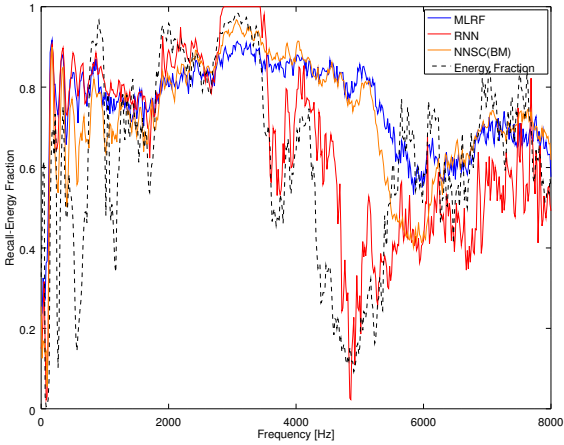
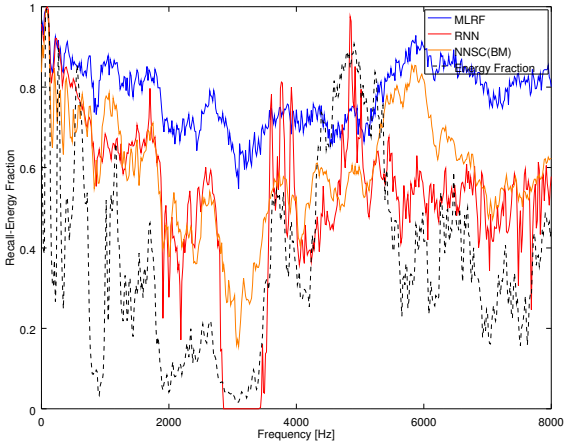
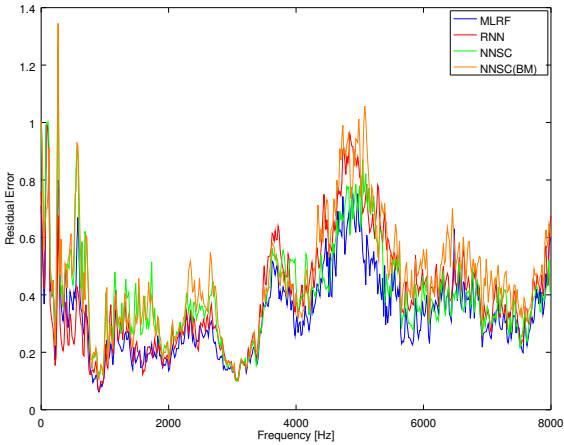
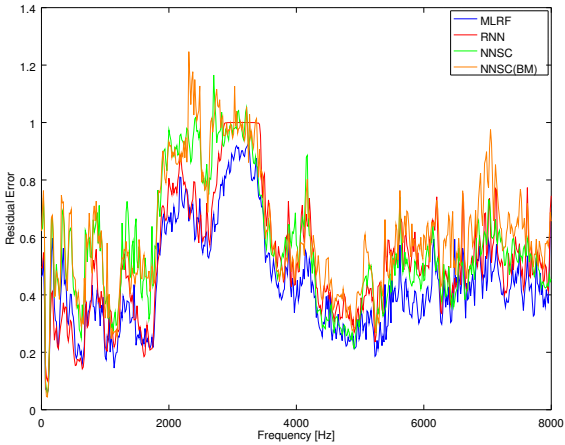


# Male-Female: Pair 3

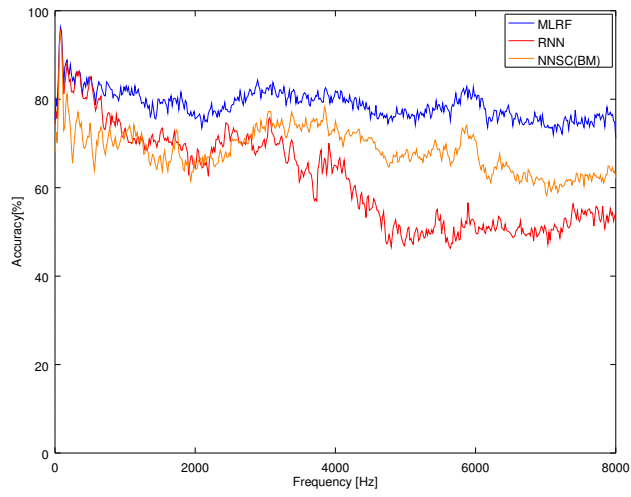
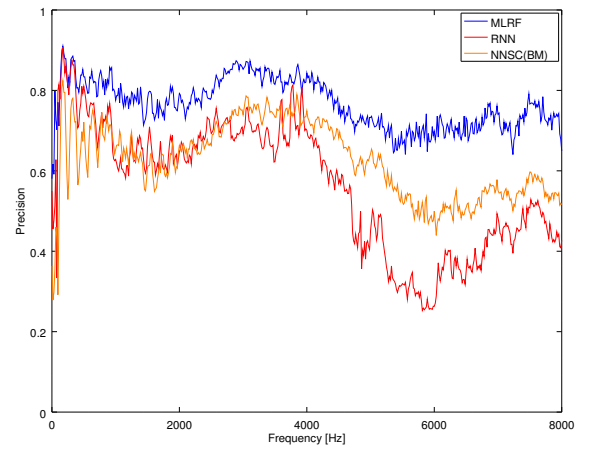
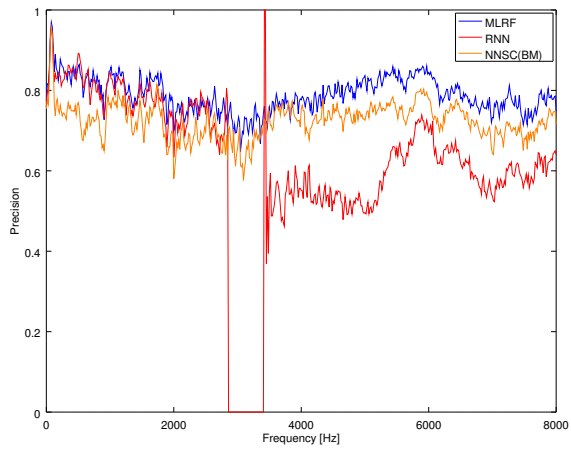




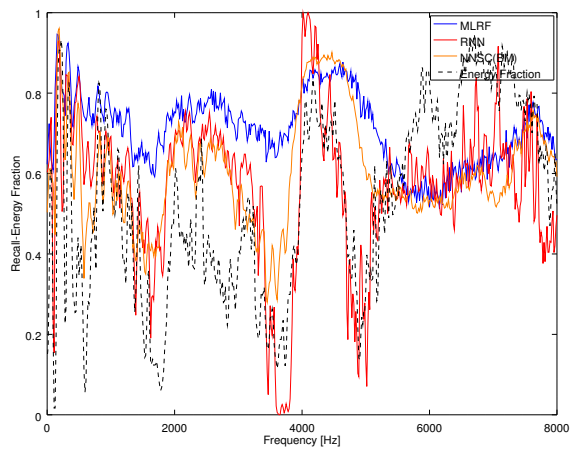
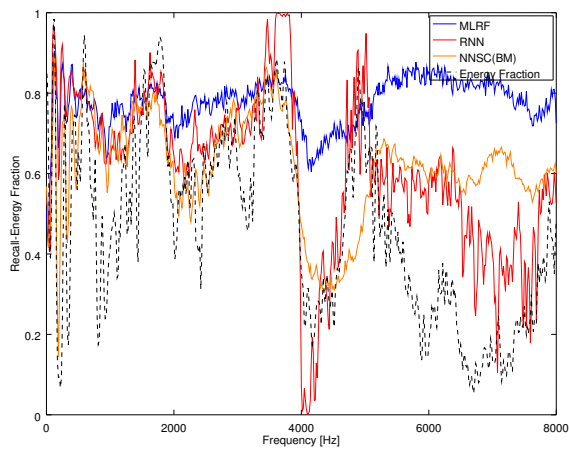
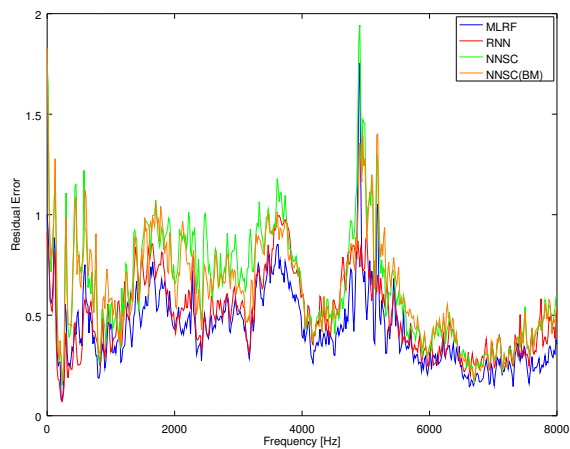
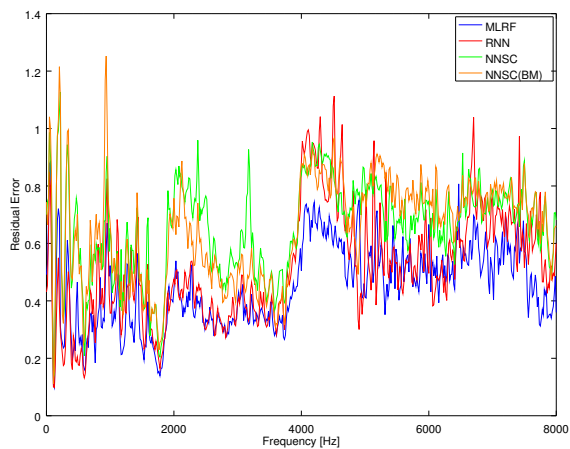
# Male-Female: Pair 4

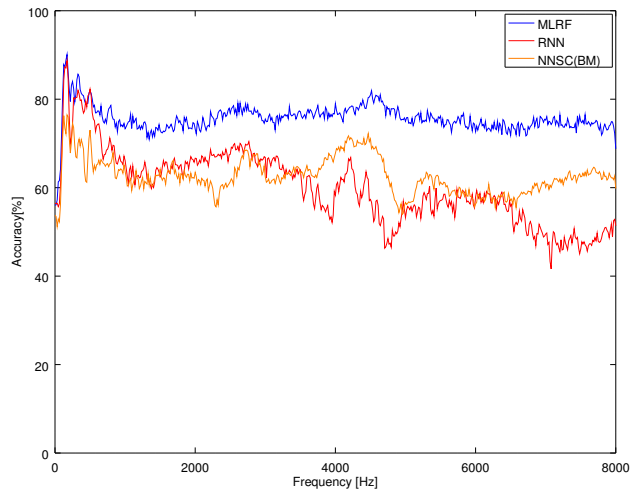
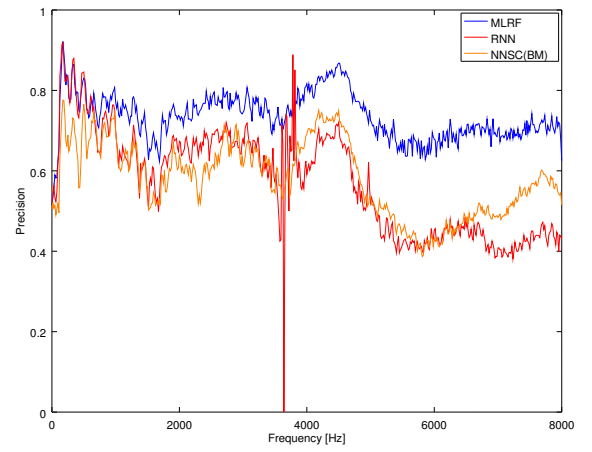
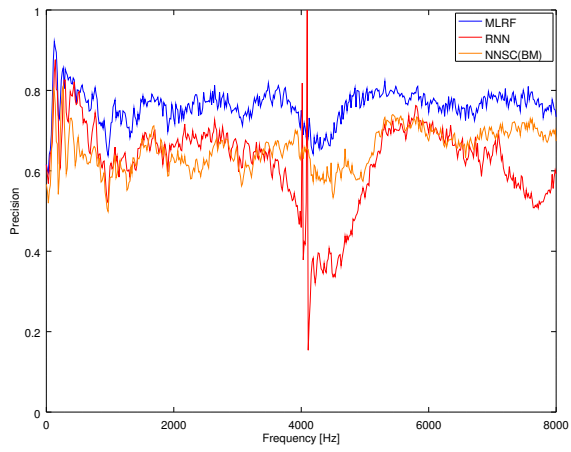




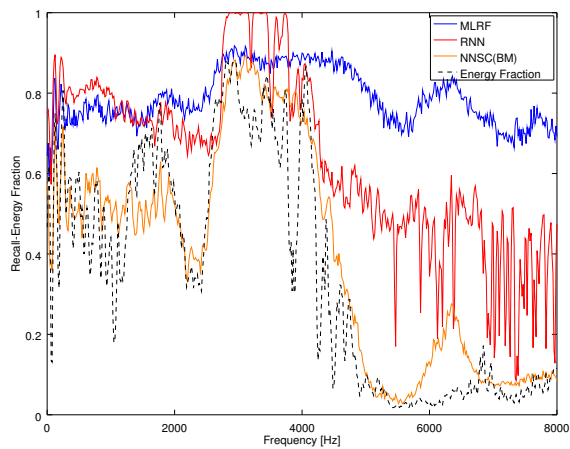
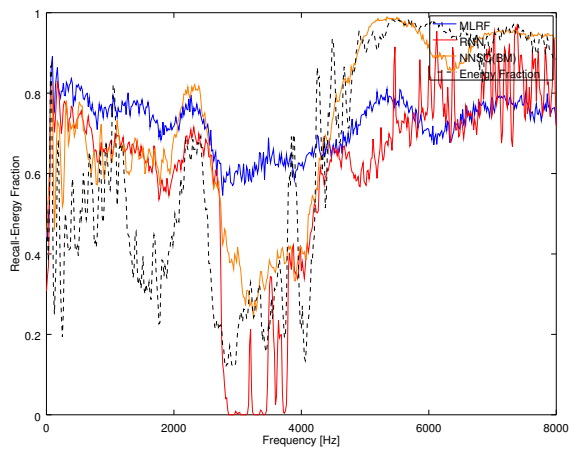
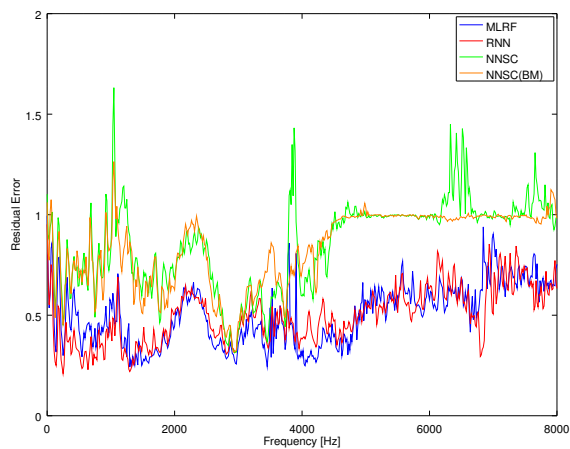
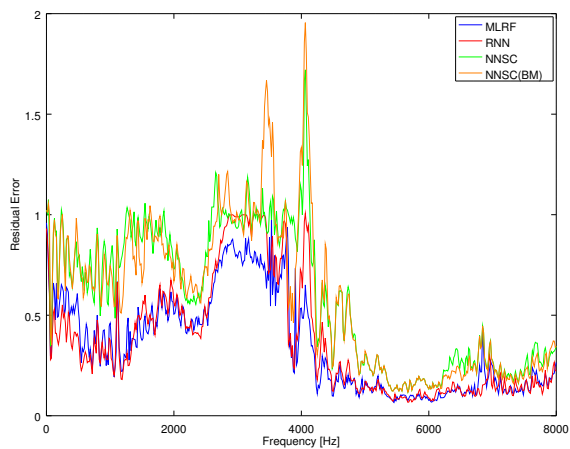


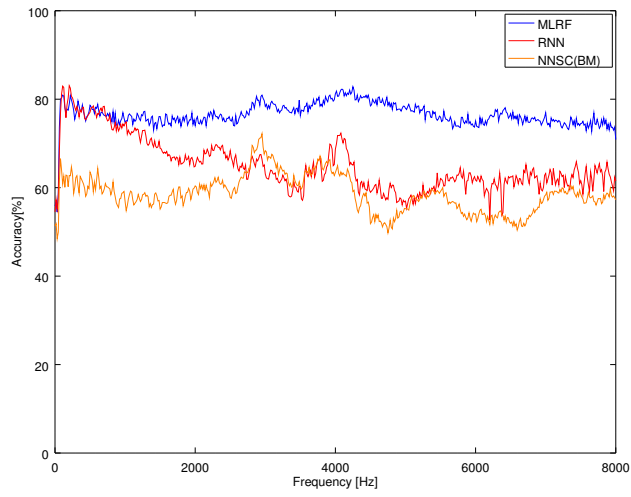
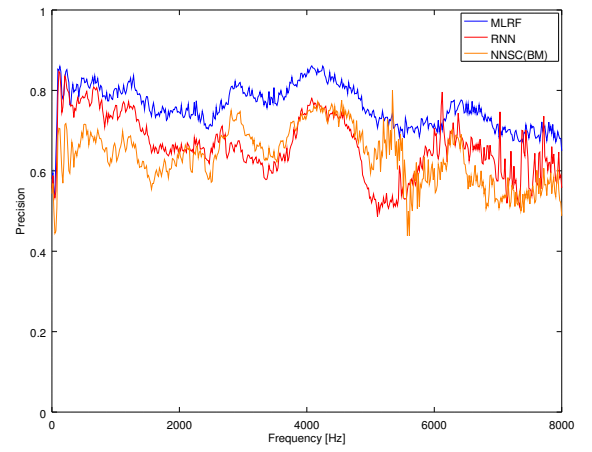
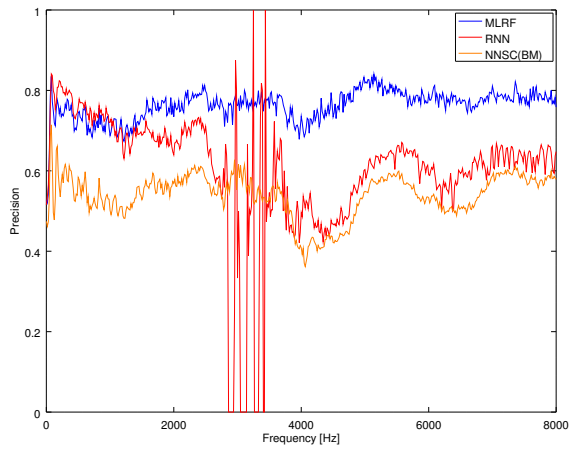
# Male-Female: Pair 5



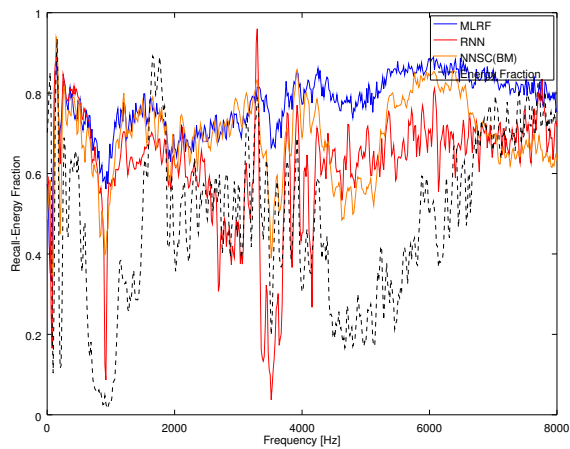
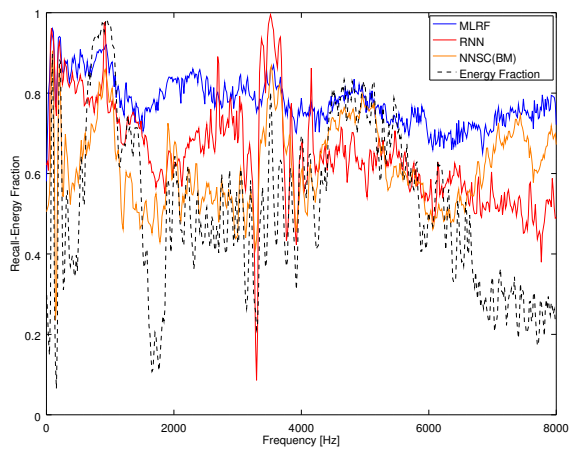
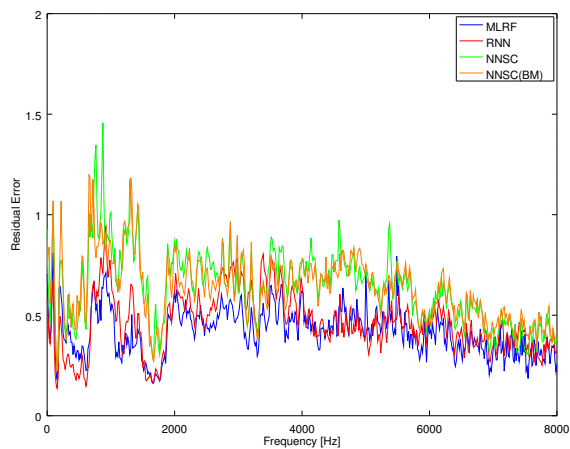
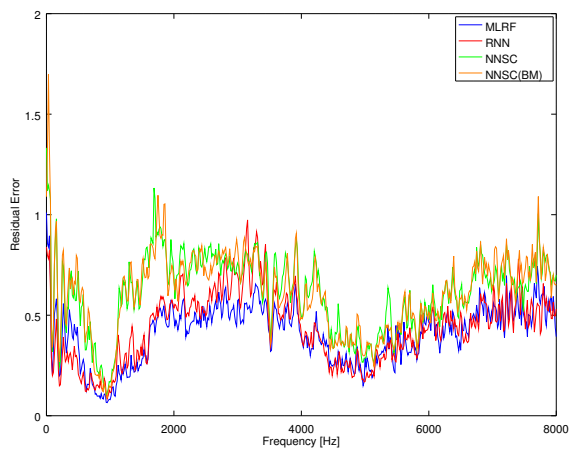


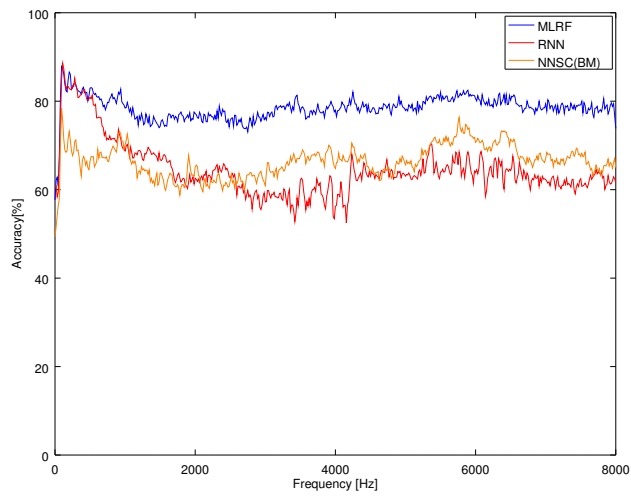
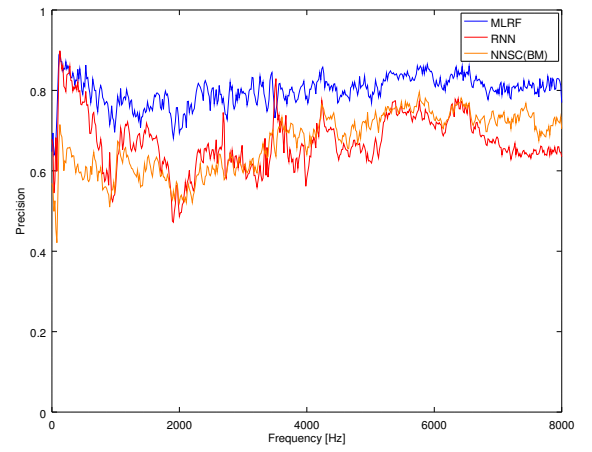
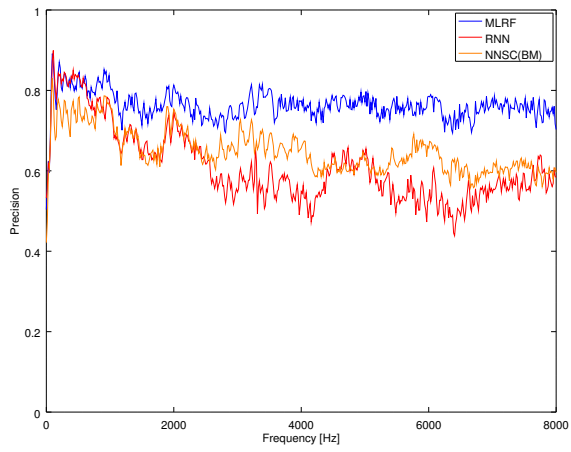
# Male-Male: Pair 6



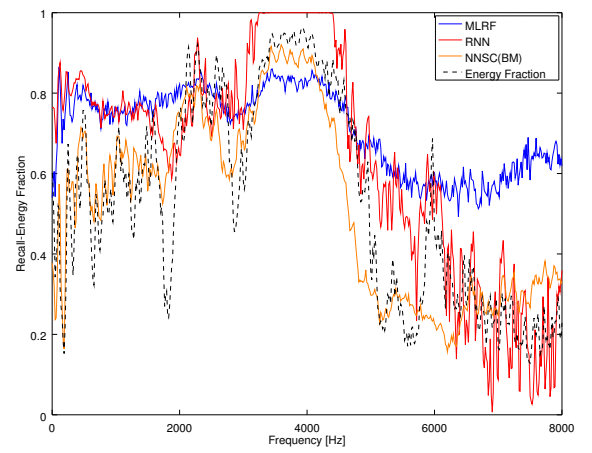
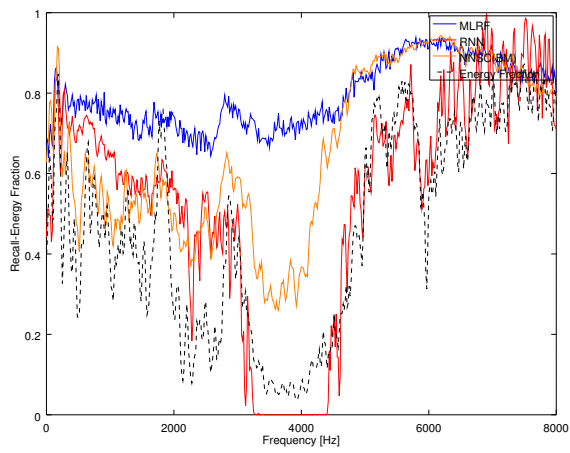
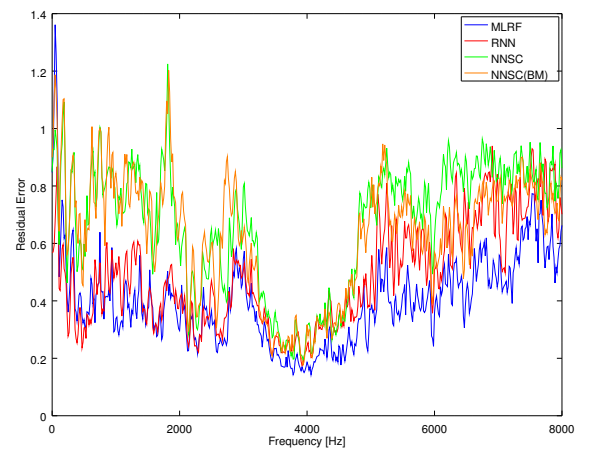
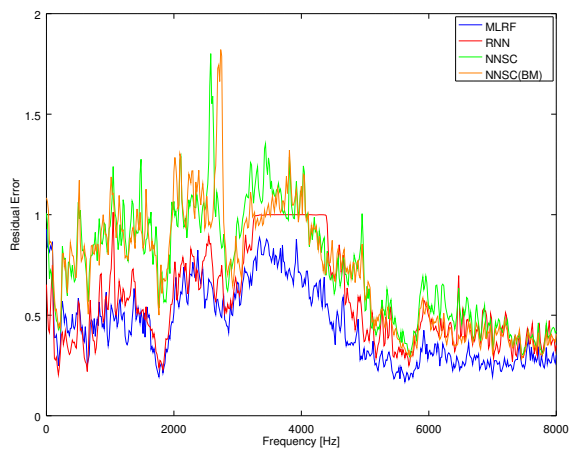


# Male-Male: Pair 7

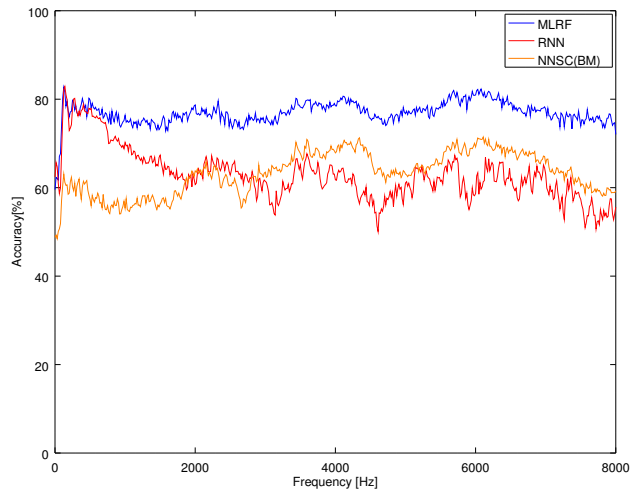
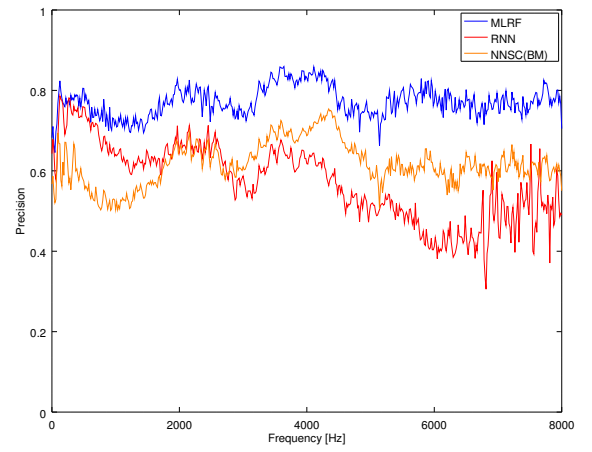
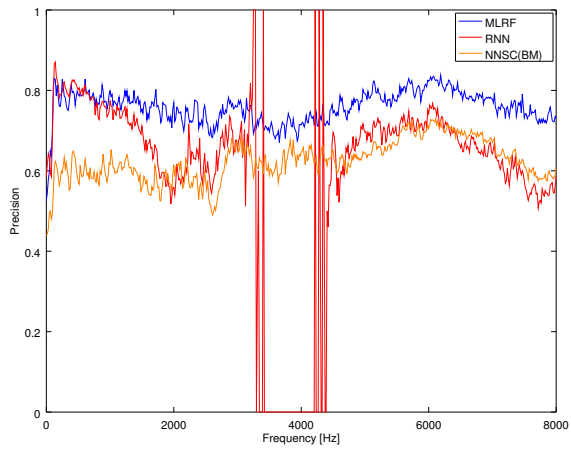




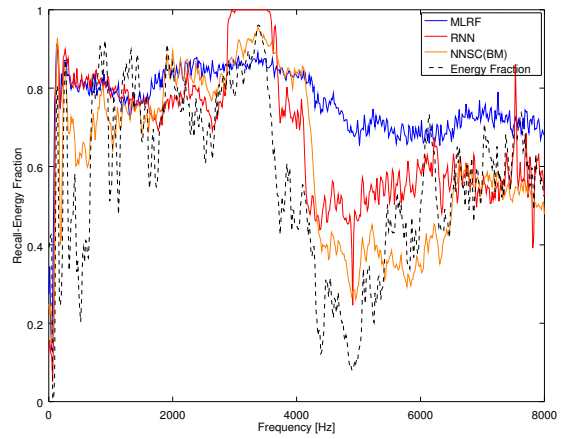
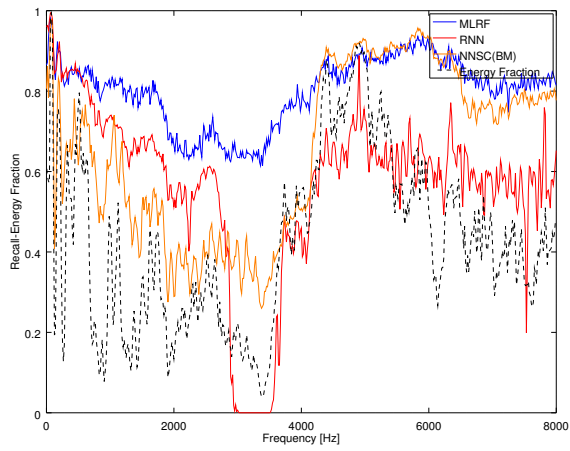
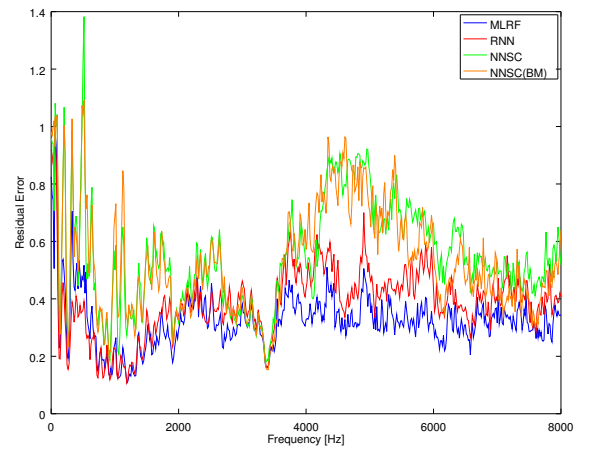
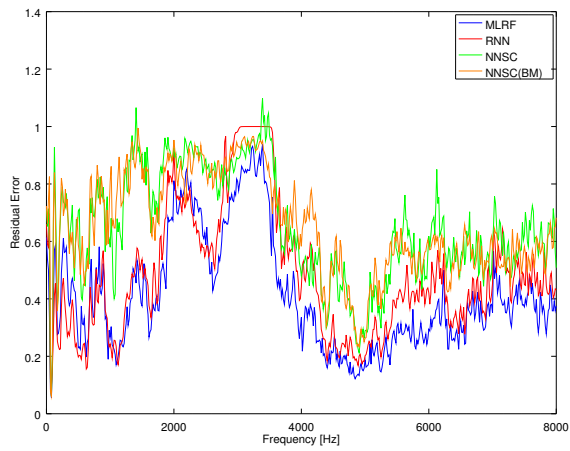
# Male-Male: Pair 8

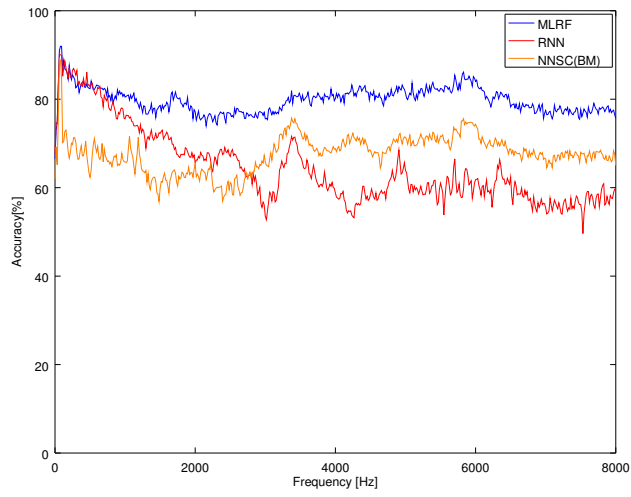
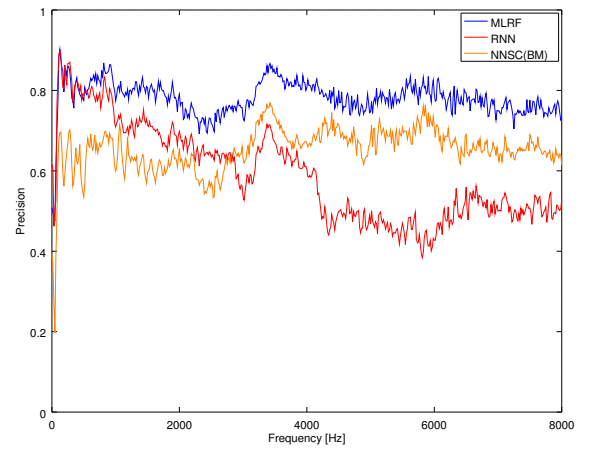
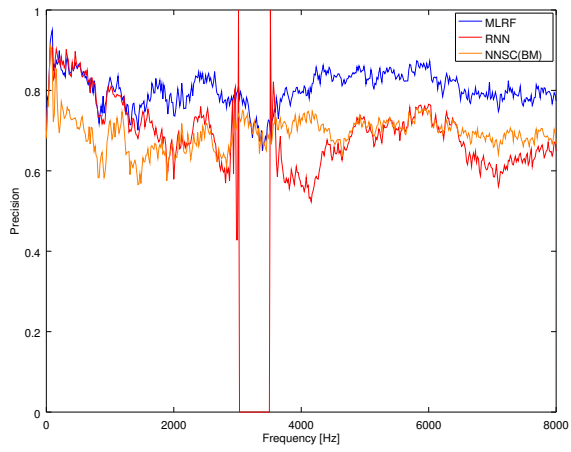




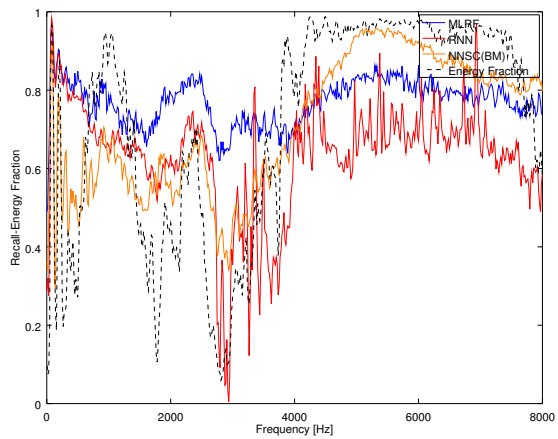
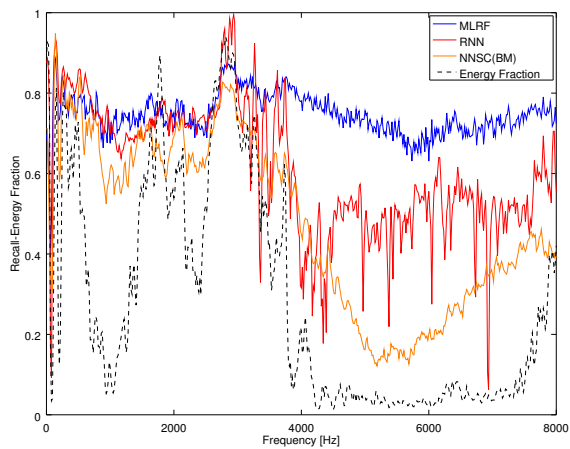
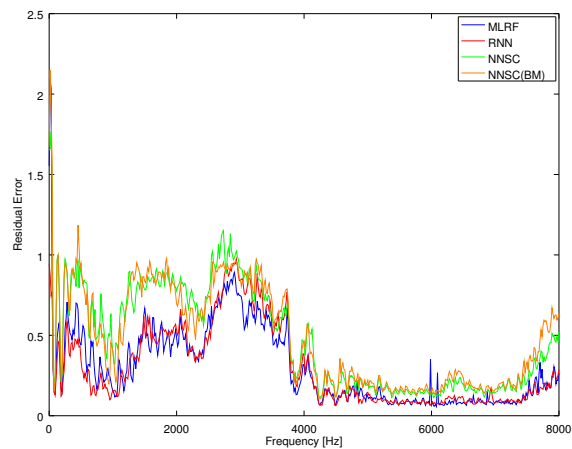
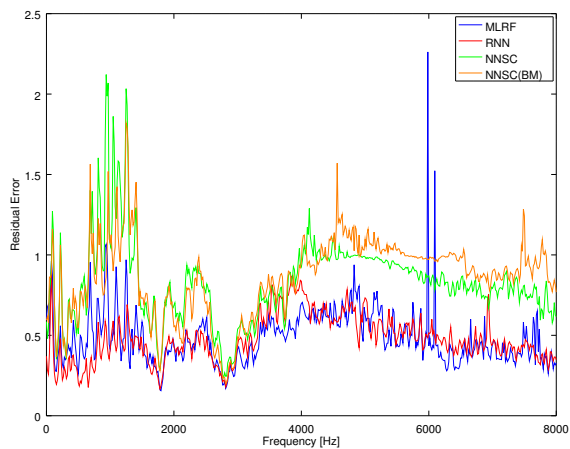


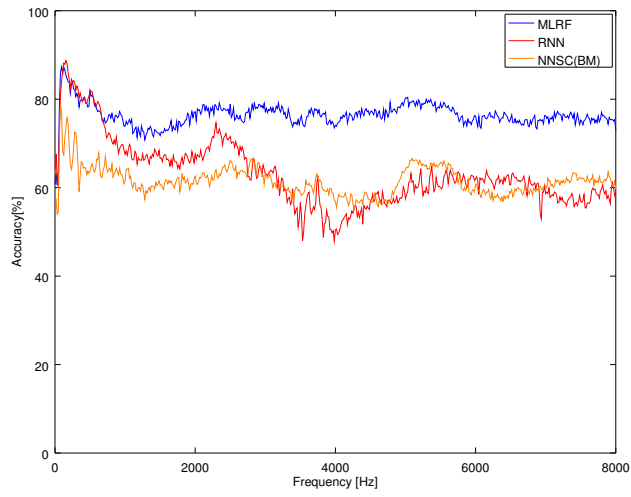
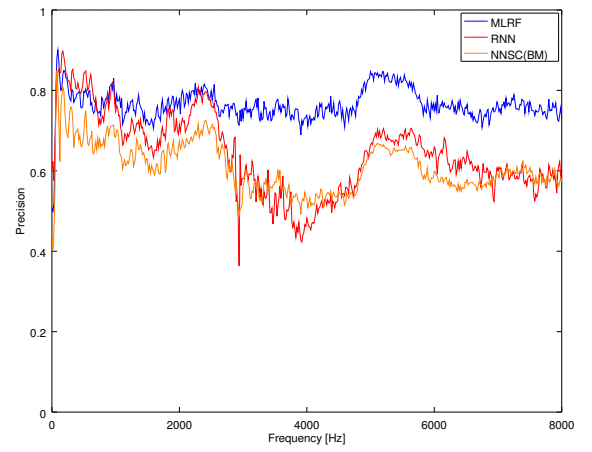
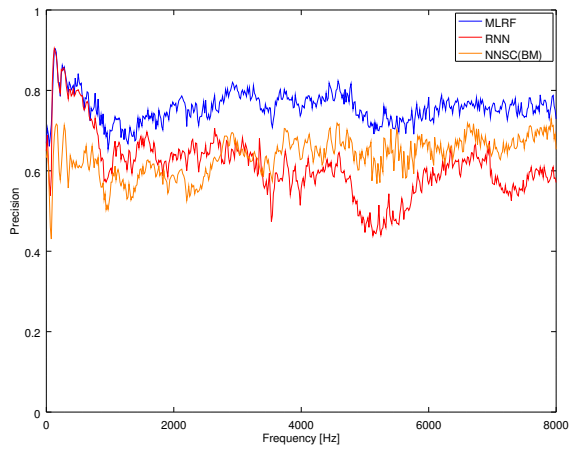
# Male-Male: Pair 9



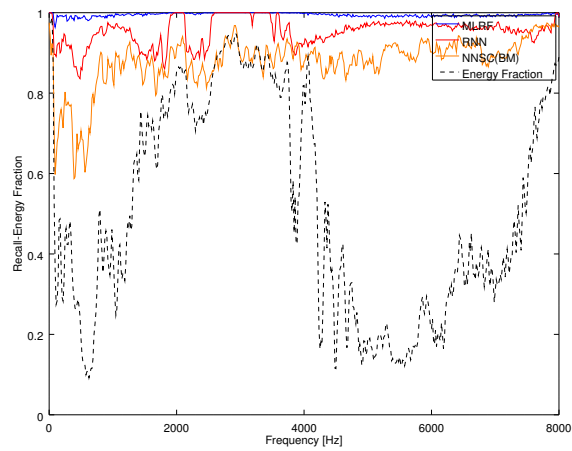
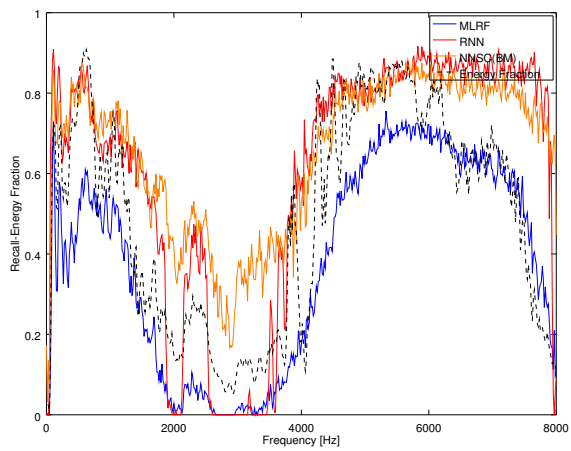
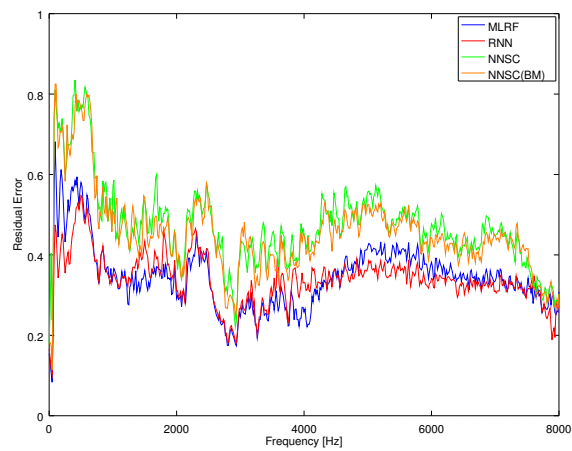
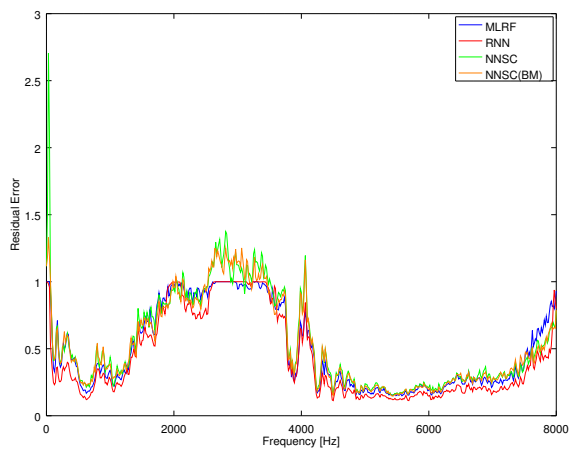


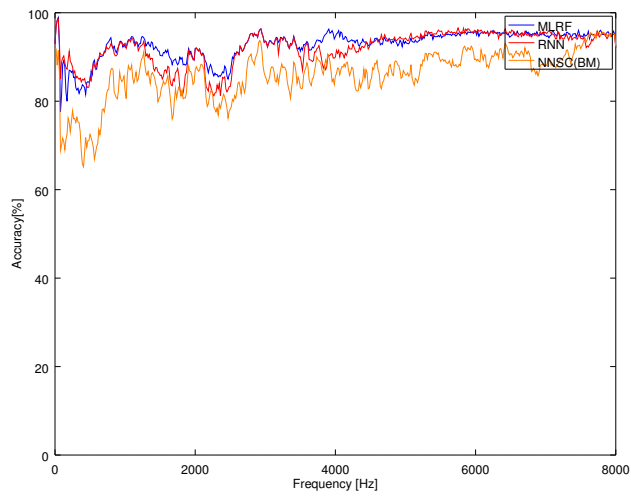
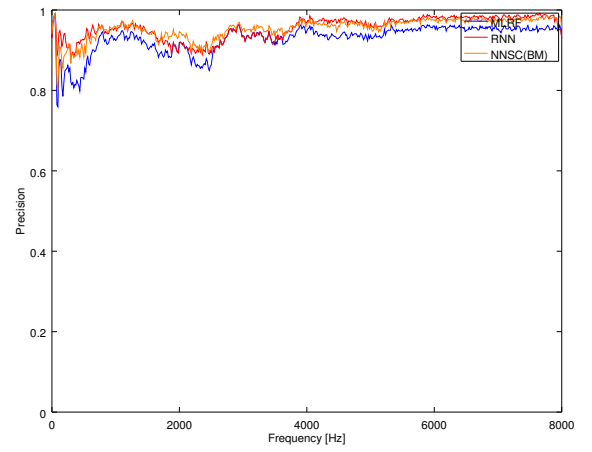
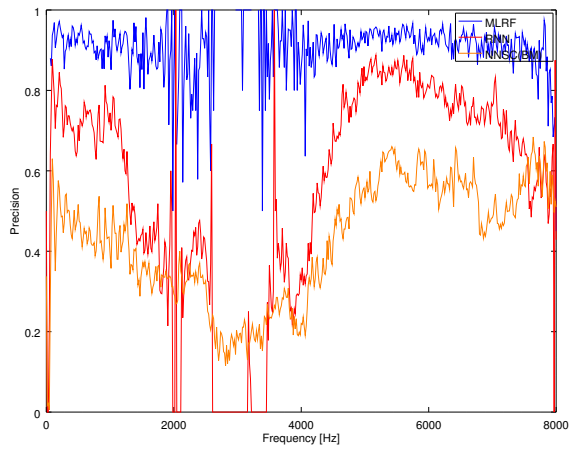
# Male-Male: Pair 10



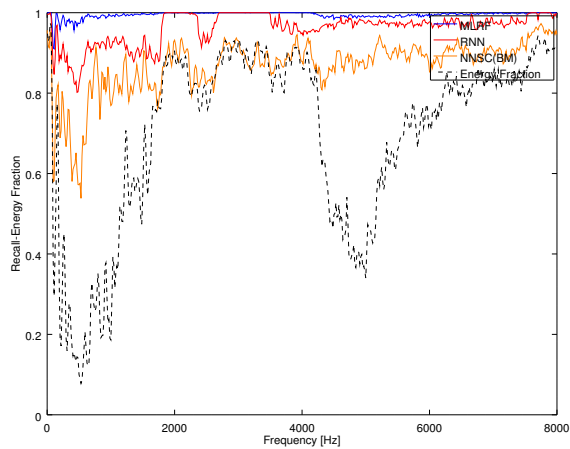
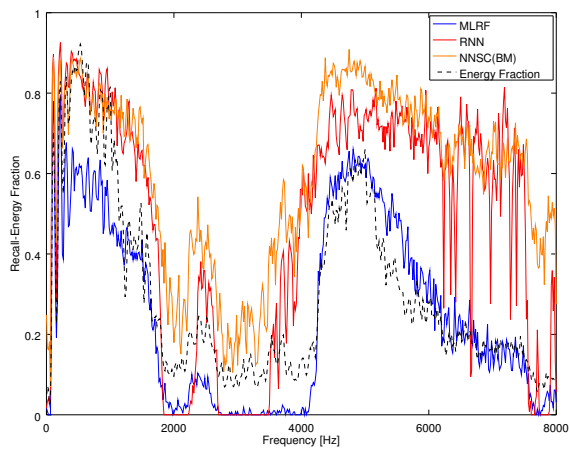
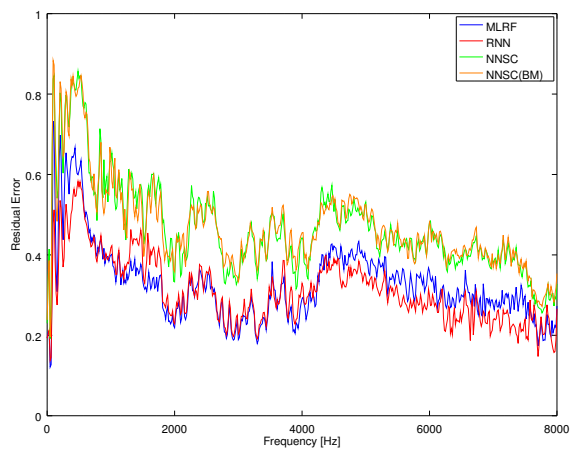
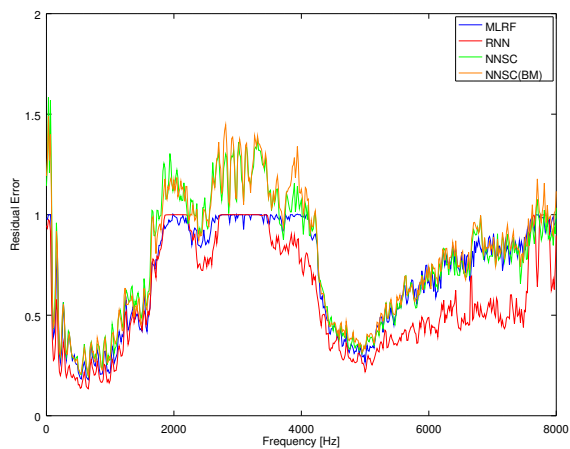


# Male-Pink Noise: Pair 11

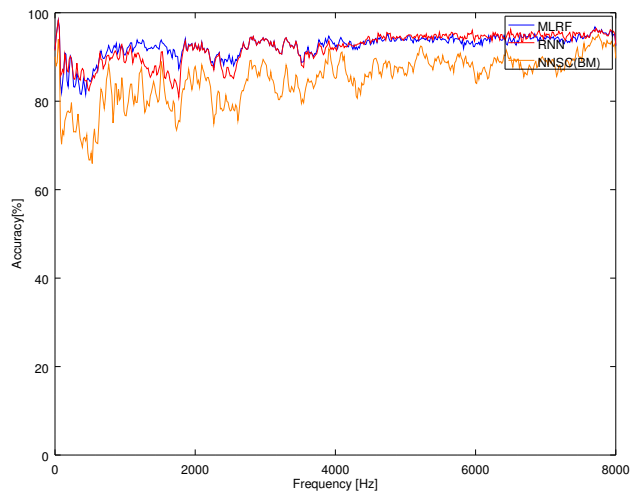
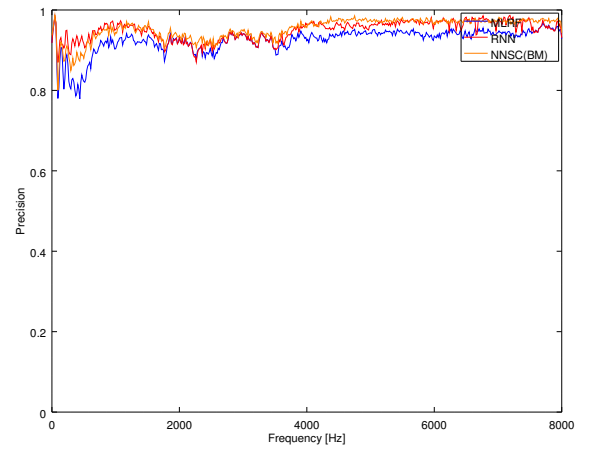
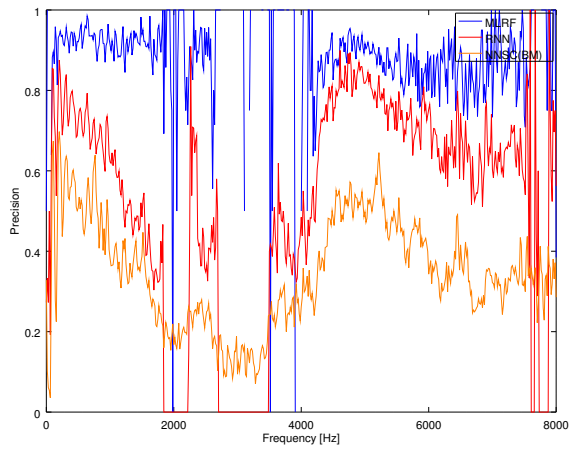




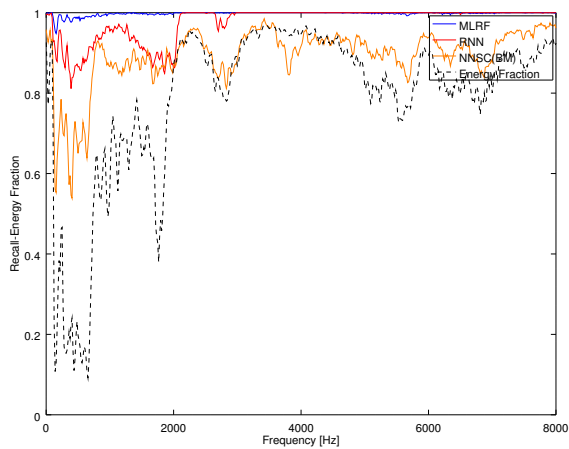
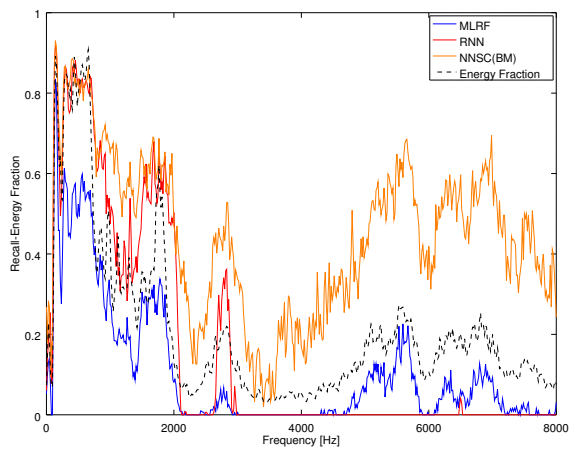
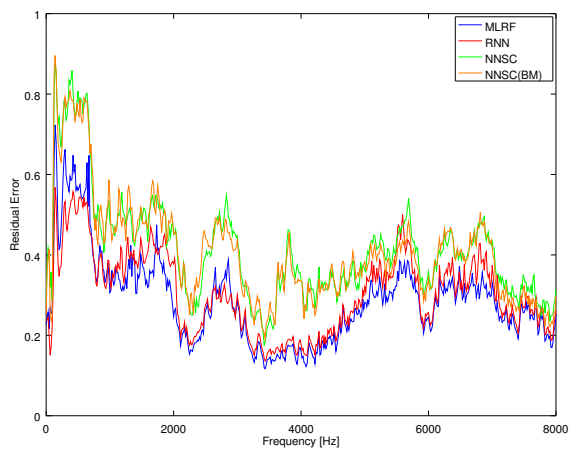
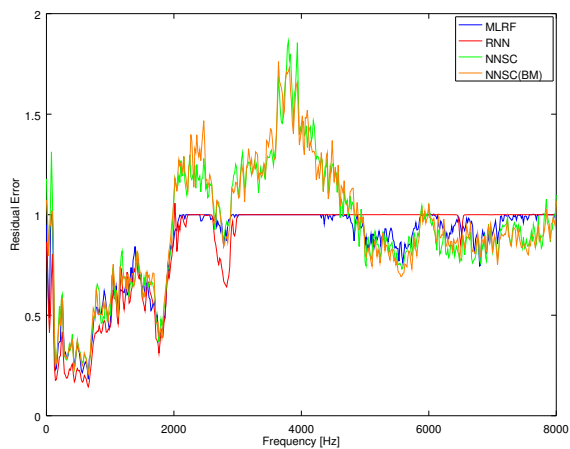
# Male-Pink Noise: Pair 12

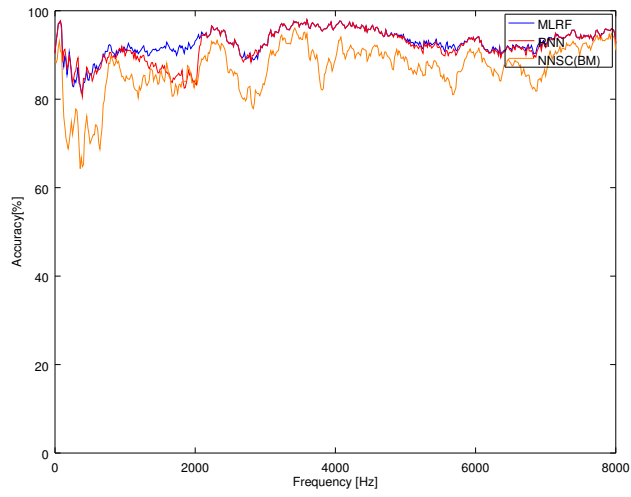
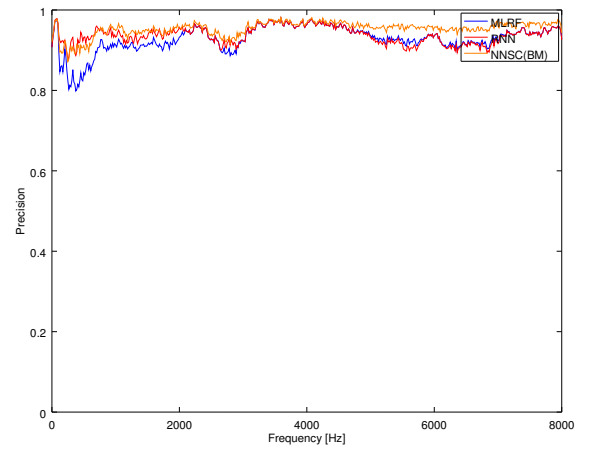
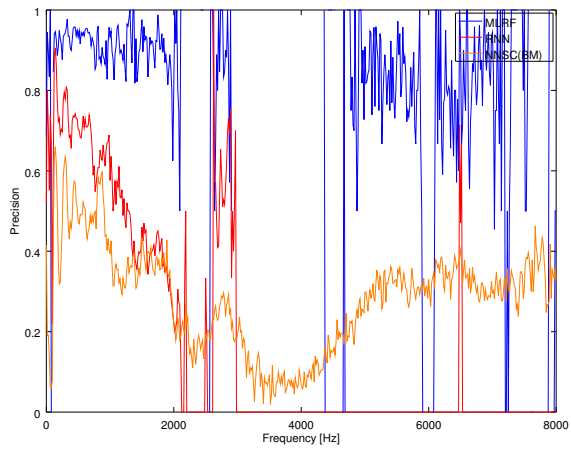




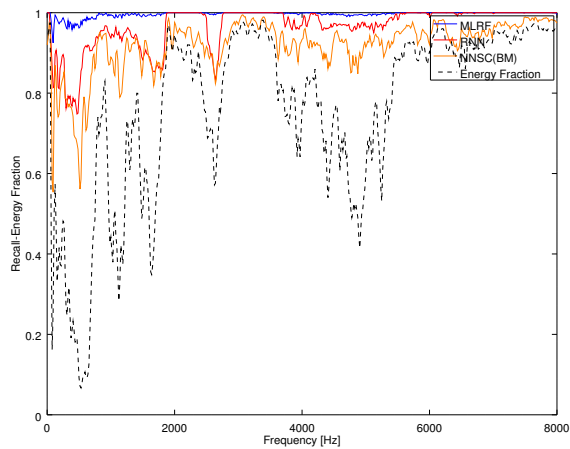
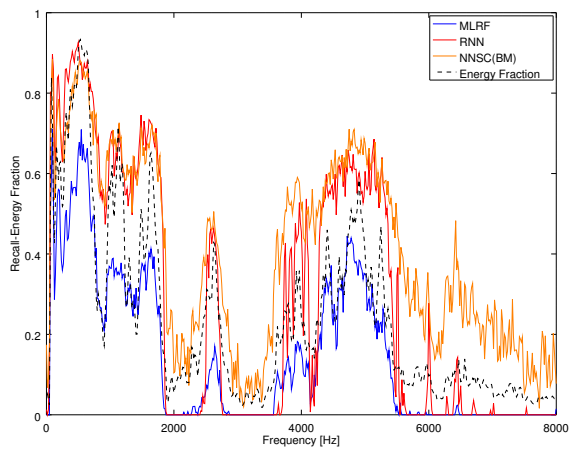
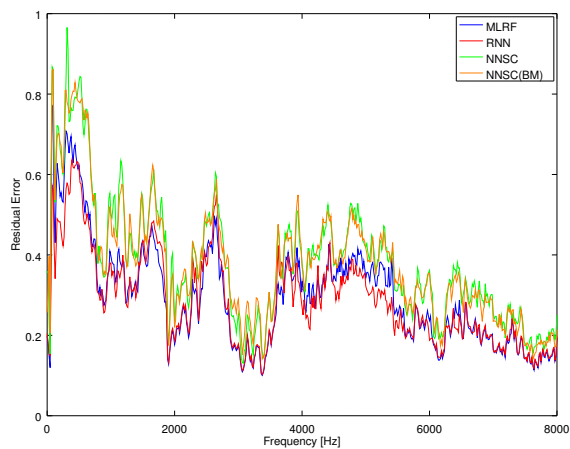
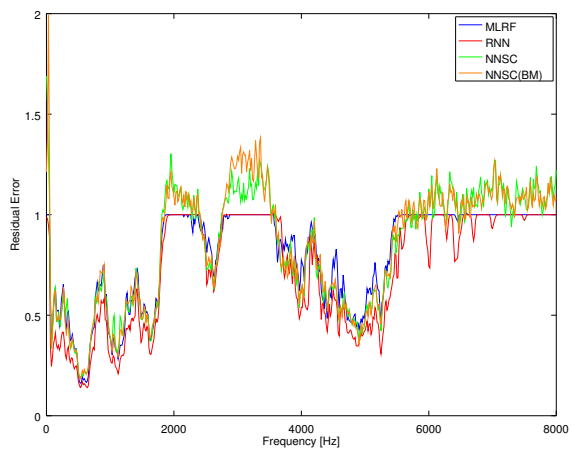


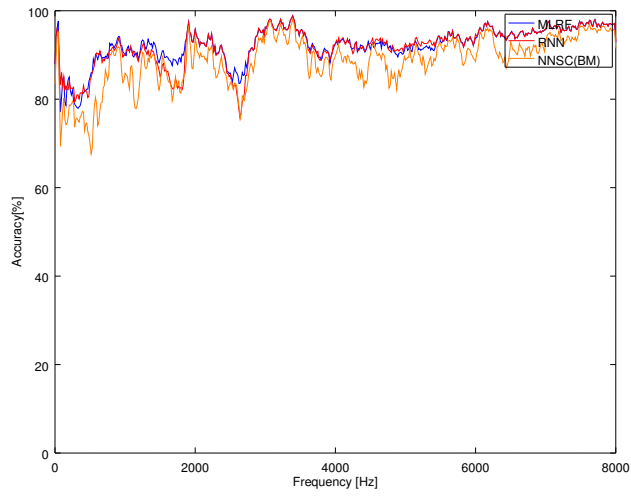
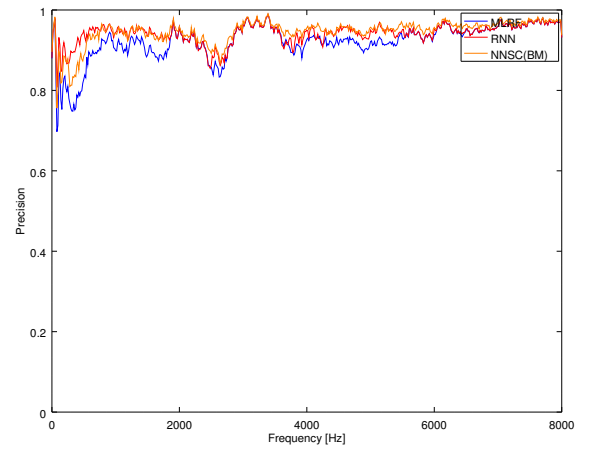
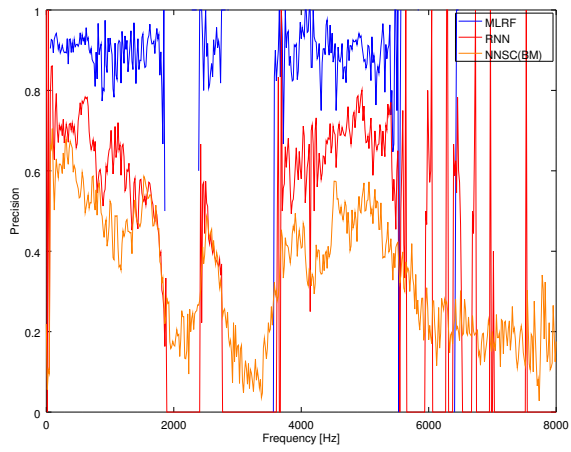
# Male-Pink Noise: Pair 13



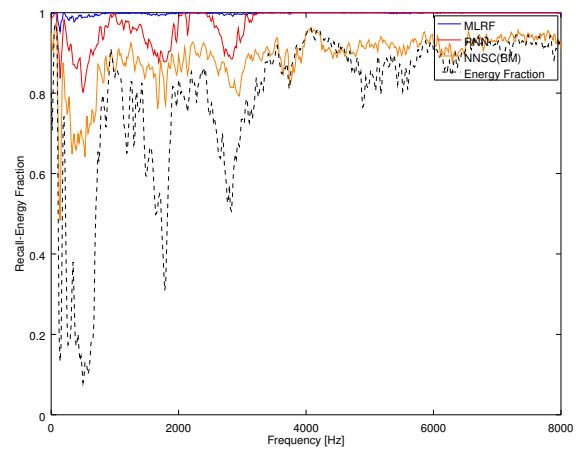
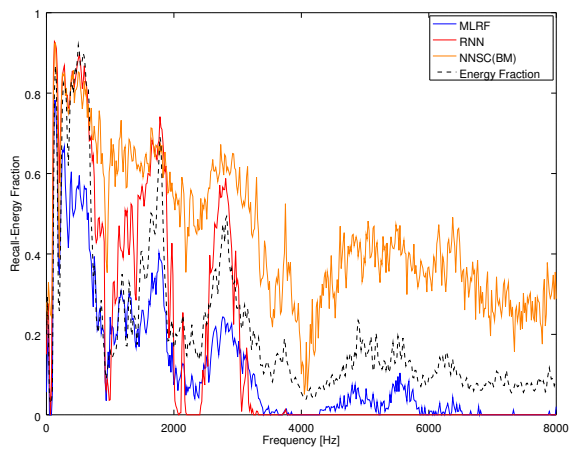
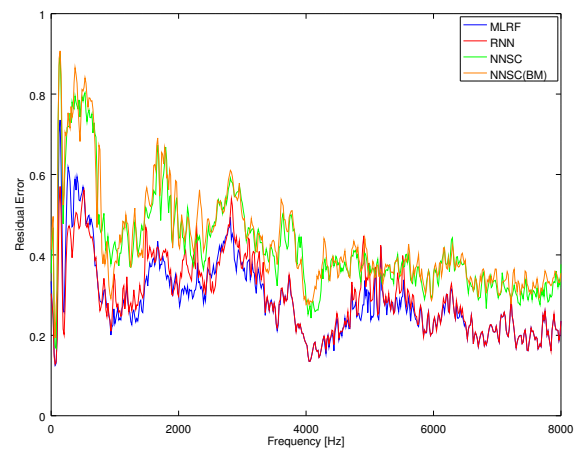
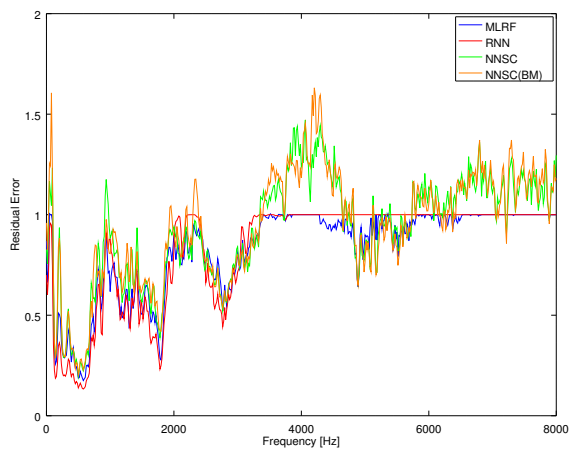


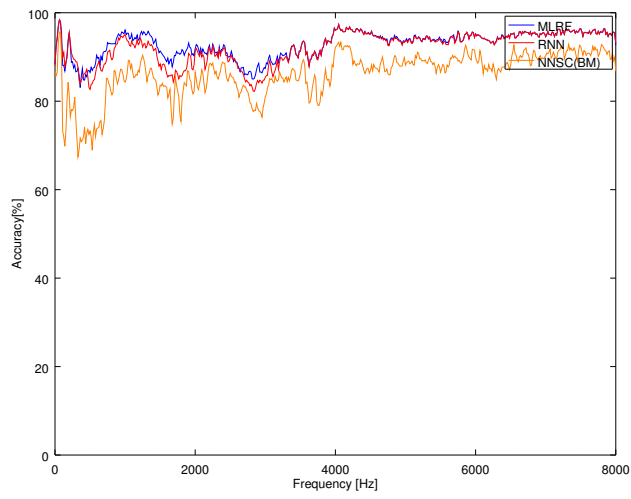
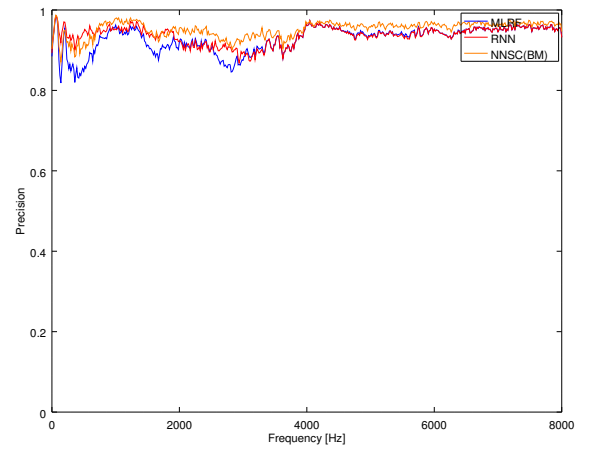
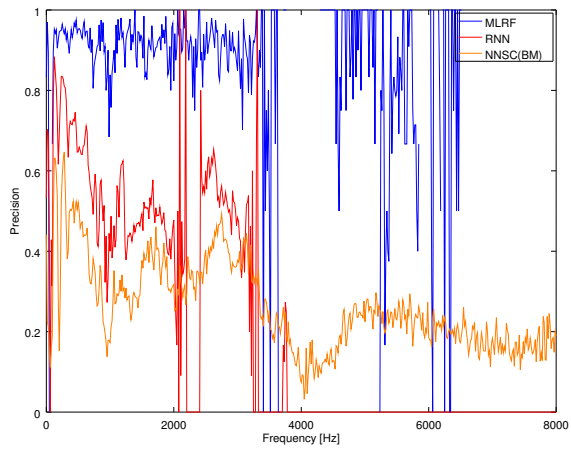
# Male-Pink Noise: Pair 14





# Male-Pink Noise: Pair 15





# Bibliography

- [1] R. Aebersold, L. E. Hood, and J. D. Watts. Equipping scientists for the new biology. *Nature Biotechnology*, 18(4):359, 2000.
- [2] L. Alonso-Nanclares, J. Gonzalez-Soriano, J. R. Rodriguez, and J. DeFelipe. Gender Differences in Human Cortical Synaptic Density. *PNAS*, 105(38):14615–14619, Sept. 2008.
- [3] L. Alonso-Nanclares, J.-R. Rodríguez, J. DeFelipe, Á. Rodríguez, and Á. Merchán-Pérez. ESPINA: a tool for the automated segmentation and counting of synapses in large stacks of electron microscopy images. *Front. Neuroanat.*, 5:18, 2011.
- [4] M. Aoki, M. Okamoto, S. Aoki, H. Matsui, T. Sakurai, and Y. Kaneda. Sound source segregation based on estimating incident angle of each frequency component of input signals acquired by multiple microphones. *Acoustical Science and Technology*, 22(2):149–157, 2001.
- [5] C. Beaulieu. Numerical data on neocortical neurons in adult rat, with special reference to the GABA population. *Brain Research*, 609(1–2):284–292, Apr. 1993.
- [6] C. Beaulieu and M. Colonnier. A comparison of the number of neurons in individual laminae of cortical areas 17, 18 and posteromedial suprasylvian (PMLS) area in the cat. *Brain Research*, 339(1):166–170, July 1985.
- [7] C. Beaulieu and M. Colonnier. A laminar analysis of the number of round-asymmetrical and flat-symmetrical synapses on spines, dendritic trunks, and cell bodies in area 17 of the cat. *The Journal of Comparative Neurology*, 231(2):180–189, Jan. 1985.
- [8] C. Beaulieu and M. Colonnier. Number and size of neurons and synapses in the motor cortex of cats raised in different environmental complexities. *The Journal of Comparative Neurology*, 289(1):178–187, Nov. 1989.
- [9] C. Beaulieu and M. Colonnier. The number of neurons in the different laminae of the binocular and monocular regions of area 17 in the cat. *The Journal of Comparative Neurology*, 217(3):337–344, Oct. 2004.
- [10] C. Becker, K. Ali, G. Knott, and P. Fua. Learning Context Cues for Synapse Segmentation. *IEEE Transactions on Medical Imaging*, 32(10):1864–1877, Oct. 2013.



- [11] M. Berning, K. M. Boergens, and M. Helmstaedter. SegEM: Efficient Image Analysis for High-Resolution Connectomics. *Neuron*, 87(6):1193–1206, Sept. 2015.
- [12] T. Binzegger, R. J. Douglas, and K. A. C. Martin. A Quantitative Map of the Circuit of Cat Primary Visual Cortex. *J. Neurosci.*, 24(39):8441–8453, Sept. 2004.
- [13] J. A. Blackard and D. J. Dean. Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. *Computers and Electronics in Agriculture*, 24(3):131–151, Dec. 1999.
- [14] K. M. Boergens, M. Berning, T. Bocklisch, D. Bräunlein, F. Drawitsch, J. Frohnhofen, T. Herold, P. Otto, N. Rzepka, T. Werkmeister, and others. webKnossos: efficient online 3d data annotation for connectomics. *nature methods*, 14(7):691–694, 2017.
- [15] V. Braitenberg and A. Schüz. *Anatomy of the cortex: Statistics and geometry*. Springer-Verlag Publishing, New York, NY, US, 1991.
- [16] L. Breiman. Random Forests. In *Machine Learning*, pages 5–32, 2001.
- [17] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. Classification and regression trees, 1984.
- [18] L. G. Brock, J. S. Coombs, and J. C. Eccles. The recording of potentials from motoneurons with an intracellular electrode. *The Journal of Physiology*, 117(4):431–460, Aug. 1952.
- [19] K. Brodmann. *Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues*. Barth, 1909.
- [20] T. F. Brooks, D. S. Pope, and M. A. Marcolini. Airfoil self-noise and prediction. Technical report, July 1989.
- [21] N. Brunel and V. Hakim. Fast Global Oscillations in Networks of Integrate-and-Fire Neurons with Low Firing Rates. *Neural Computation*, 11(7):1621–1671, Oct. 1999.
- [22] N. Brunel and X.-J. Wang. What Determines the Frequency of Fast Network Oscillations With Irregular Neural Discharges? I. Synaptic Dynamics and Excitation-Inhibition Balance. *Journal of Neurophysiology*, 90(1):415–430, July 2003.
- [23] D. S. Brungart, P. S. Chang, B. D. Simpson, and D. Wang. Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *The Journal of the Acoustical Society of America*, 120(6):4007–4018, Dec. 2006.
- [24] V. S. Caviness. Architectonic map of neocortex of the normal mouse. *The Journal of Comparative Neurology*, 164(2):247–263, 1975.

- [25] J. R. Christensen, K. B. Larsen, S. H. Lisanby, J. Scalia, V. Arango, A. J. Dwork, and B. Pakkenberg. Neocortical and hippocampal neuron and glial cell numbers in the rhesus monkey. *The Anatomical Record: Advances in Integrative Anatomy and Evolutionary Biology*, 290(3):330–340, Feb. 2007.
- [26] D. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3642–3649, June 2012.
- [27] M. Colonnier. Synaptic patterns on different cell types in the different laminae of the cat visual cortex. An electron microscope study. *Brain Research*, 9(2):268–287, July 1968.
- [28] M. Cooke, J. Barker, S. Cunningham, and X. Shao. An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424, Oct. 2006.
- [29] B. G. Cragg. The density of synapses and neurones in the motor and visual areas of the cerebral cortex. *J Anat*, 101(Pt 4):639–654, Sept. 1967.
- [30] D. K. Cullen, M. E. Gilroy, H. R. Irons, and M. C. LaPlaca. Synapse-to-neuron ratio is inversely related to neuronal density in mature neuronal cultures. *Brain Research*, 1359(0):44–55, Nov. 2010.
- [31] E. M. Curtis, M. G. Stewart, and T. S. King. Quantitation of synaptic, neuronal and glial development in the intermediate and medial hyperstriatum ventrale (IMHV) of the chick *Gallus domesticus*, pre- and post-hatch. *Developmental Brain Research*, 48(1):105–118, July 1989.
- [32] N. M. da Costa, K. Hepp, and K. A. Martin. A systematic random sampling scheme optimized to detect the proportion of rare synapses in the neuropil. *Journal of Neuroscience Methods*, 180(1):77–81, May 2009.
- [33] J. DeFelipe, R. D. Fields, P. R. Hof, M. Hoistad, I. Kostovic, G. Meyer, and K. S. Rockland. Cortical White Matter: Beyond the Pale Remarks, Main Conclusions and Discussion. *Front Neuroanat*, 4, Mar. 2010.
- [34] J. DeFelipe, P. Marco, I. Busturia, and A. Merchán-Pérez. Estimation of the Number of Synapses in the Cerebral Cortex: Methodological Considerations. *Cereb. Cortex*, 9(7):722–732, Oct. 1999.
- [35] D. Deterding. UCI Machine Learning Repository: Connectionist Bench (Vowel Recognition - Deterding Data) Data Set.
- [36] I. S. Dhillon and S. Sra. Modeling data using directional distributions. Technical report, Technical Report TR-03-06, Department of Computer Sciences, The University

- of Texas at Austin. URL <ftp://ftp.cs.utexas.edu/pub/techreports/tr03-06>. ps. gz, 2003.
- [37] S. Dorkenwald, P. J. Schubert, M. F. Killinger, G. Urban, S. Mikula, F. Svara, and J. Kornfeld. Automated synaptic connectivity inference for volume electron microscopy. *Nat Meth*, 14(4):435–442, Apr. 2017.
- [38] R. J. Douglas, K. A. Martin, and D. Whitteridge. A Canonical Microcircuit for Neocortex. *Neural Computation*, 1(4):480–488, 1989.
- [39] A. L. Eberle, R. Schalek, J. W. Lichtman, M. Malloy, B. Thiel, and D. Zeidler. Multiple-Beam Scanning Electron Microscopy. *Microscopy Today*, 23(2):12–19, Mar. 2015.
- [40] R. Egger, V. J. Dercksen, D. Udvary, H.-C. Hege, and M. Oberlaender. Generation of dense statistical connectomes from sparse morphological data. *Front. Neuroanat*, 8:129, 2014.
- [41] N. Eriksen and B. Pakkenberg. Total neocortical cell number in the mysticete brain. *The Anatomical Record: Advances in Integrative Anatomy and Evolutionary Biology*, 290(1):83–95, Jan. 2007.
- [42] I. W. Evett and E. J. Spiehler. UCI Machine Learning Repository: Glass Identification Data Set.
- [43] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, 4(12):2379–2394, Dec. 1987.
- [44] R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(2):179–188, Sept. 1936.
- [45] M. Forina. UCI Machine Learning Repository: Wine Data Set.
- [46] R. S. Forsyth. UCI Machine Learning Repository: Liver Disorders Data Set, May 1990.
- [47] P. Gabbott and M. Stewart. Distribution of neurons and glia in the visual cortex (area 17) of the adult albino rat: A quantitative description. *Neuroscience*, 21(3):833–845, June 1987.
- [48] D. Gabor. Theory of communication. Part 1: The analysis of information. *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering*, 93(26):429–441, Nov. 1946.
- [49] L. Garey, J. Takács, A. Revishchin, and J. Hámori. Quantitative distribution of GABA-immunoreactive neurons in cetacean visual cortex is similar to that in land mammals. *Brain Research*, 485(2):278–284, Apr. 1989.

- [50] L. J. Garey and G. Leuba. A quantitative study of neuronal and glial numerical density in the visual cortex of the bottlenose dolphin: Evidence for a specialized subarea and changes with age. *The Journal of Comparative Neurology*, 247(4):491–496, May 1986.
- [51] C. Genoud, G. W. Knott, K. Sakata, B. Lu, and E. Welker. Altered Synapse Formation in the Adult Somatosensory Cortex of Brain-Derived Neurotrophic Factor Heterozygote Mice. *J. Neurosci.*, 24(10):2394–2400, Mar. 2004.
- [52] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Mach Learn*, 63(1):3–42, Apr. 2006.
- [53] I. I. Glezer and P. J. Morgane. Ultrastructure of synapses and golgi analysis of neurons in neocortex of the lateral gyrus (visual cortex) of the dolphin and pilot whale. *Brain Research Bulletin*, 24(3):401–427, Mar. 1990.
- [54] C. Golgi. Sulla struttura della sostanza grigia del cervello. *Gazzetta Medica Italiana. Lombardia*, 33:244–246, 1873.
- [55] R. P. Gorman and T. J. Sejnowski. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1(1):75–89, Jan. 1988.
- [56] E. G. Gray. Axo-somatic and axo-dendritic synapses of the cerebral cortex. *J Anat*, 93(Pt 4):420–433, Oct. 1959.
- [57] C. G. Gross. Early history of neuroscience. *Encyclopedia of neuroscience*, 2:843–846, 1987.
- [58] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405(6789):947–951, June 2000.
- [59] R. H. R. Hahnloser, H. S. Seung, and J. J. Slotine. Permitted and Forbidden Sets in Symmetric Threshold-Linear Networks. *Neural Computation*, 15(3):621–638, Mar. 2003.
- [60] K. Han and D. Wang. A classification based approach to speech segregation. *The Journal of the Acoustical Society of America*, 132(5):3475–3483, Nov. 2012.
- [61] D. Harrison and D. L. Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5(1):81–102, Mar. 1978.
- [62] M. Hassiotis, G. Paxinos, and K. Ashwell. The anatomy of the cerebral cortex of the echidna (*Tachyglossus aculeatus*). *Comparative Biochemistry and Physiology - Part A: Molecular & Integrative Physiology*, 136(4):827–850, Dec. 2003.
- [63] E. W. Healy, S. E. Yoho, Y. Wang, and D. Wang. An algorithm to improve speech recognition in noise for hearing-impaired listeners. *The Journal of the Acoustical Society of America*, 134(4):3029–3038, Oct. 2013.

- [64] D. O. Hebb. *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.
- [65] M. Helmstaedter, K. L. Briggman, S. C. Turaga, V. Jain, H. S. Seung, and W. Denk. Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature*, 500(7461):168–174, Aug. 2013.
- [66] S. H. Hendry, H. D. Schwark, E. G. Jones, and J. Yan. Numbers and Proportions of GABA-Immunoreactive Neurons in Different Areas of Monkey Cerebral Cortex. *J. Neurosci.*, 7(5):1503–1519, May 1987.
- [67] S. Herculano-Houzel, C. E. Collins, P. Wong, and J. H. Kaas. Cellular Scaling Rules for Primate Brains. *PNAS*, 104(9):3562–3567, Feb. 2007.
- [68] S. Herculano-Houzel, C. E. Collins, P. Wong, J. H. Kaas, and R. Lent. The Basic Nonuniformity of the Cerebral Cortex. *PNAS*, 105(34):12593–12598, Aug. 2008.
- [69] S. Herculano-Houzel, B. Mota, and R. Lent. Cellular Scaling Rules for Rodent Brains. *PNAS*, 103(32):12138–12143, Aug. 2006.
- [70] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997.
- [71] A. L. Hodgkin and A. F. Huxley. Propagation of electrical signals along giant nerve fibres. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, pages 177–183, 1952.
- [72] P. R. Hof, R. Chavis, and L. Marino. Cortical complexity in cetacean brains. *Anat. Rec.*, 287A(1):1142–1152, Nov. 2005.
- [73] H. Hofmann. UCI Machine Learning Repository: Statlog (German Credit Data) Data Set, Nov. 1994.
- [74] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *PNAS*, 79(8):2554–2558, Apr. 1982.
- [75] J. J. Hopfield and A. V. Herz. Rapid local synchronization of action potentials: toward computation with coupled integrate-and-fire neurons. *Proc Natl Acad Sci U S A*, 92(15):6655–6662, July 1995.
- [76] Y. Hu and P. C. Loizou. Evaluation of Objective Quality Measures for Speech Enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):229–238, Jan. 2008.
- [77] G. B. Huang, L. K. Scheffer, and S. M. Plaza. Fully-Automatic Synapse Prediction and Validation on a Large Data Set. *arXiv:1604.03075 [cs]*, Apr. 2016. arXiv: 1604.03075.

- [78] P. S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis. Deep learning for monaural speech separation. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1562–1566, May 2014.
- [79] D. H. Hubel and T. N. Wiesel. Receptive fields of single neurones in the cat’s striate cortex. *The Journal of physiology*, 148(3):574–591, 1959.
- [80] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, Apr. 2004.
- [81] H. Ishwaran. The effect of splitting on random forests. *Mach Learn*, 99(1):75–118, Apr. 2015.
- [82] M. Jasinska, A. Grzegorzczak, E. Jasek, J. A. Litwin, M. Kossut, G. Barbacka-Surowiak, and E. Pyza. Daily rhythm of synapse turnover in mouse somatosensory cortex. *Acta Neurobiol Exp (Wars)*, 74(1):104–110, 2014.
- [83] M. Jasinska, A. Grzegorzczak, O. Woznicka, E. Jasek, M. Kossut, G. Barbacka-Surowiak, J. A. Litwin, and E. Pyza. Circadian rhythmicity of synapses in mouse somatosensory cortex. *Eur J Neurosci*, 42(8):2585–2594, Oct. 2015.
- [84] S. Jinno and T. Kosaka. Stereological estimation of numerical densities of glutamatergic principal neurons in the mouse hippocampus. *Hippocampus*, 20(7):829–840, 2010.
- [85] L. A. Jorgenson, W. T. Newsome, D. J. Anderson, C. I. Bargmann, E. N. Brown, K. Deisseroth, J. P. Donoghue, K. L. Hudson, G. S. F. Ling, P. R. MacLeish, E. Marder, R. A. Normann, J. R. Sanes, M. J. Schnitzer, T. J. Sejnowski, D. W. Tank, R. Y. Tsien, K. Ugurbil, and J. C. Wingfield. The BRAIN Initiative: developing technology to catalyse neuroscience discovery. *Phil. Trans. R. Soc. B*, 370(1668):20140164, May 2015.
- [86] A. Jourjine, S. Rickard, and O. Yilmaz. Blind separation of disjoint orthogonal signals: demixing N sources from 2 mixtures. In *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, volume 5, pages 2985–2988 vol.5, 2000.
- [87] T.-P. Jung, C. Humphries, T.-W. Lee, S. Makeig, M. J. McKeown, V. Iragui, and T. J. Sejnowski. Extended ICA Removes Artifacts from Electroencephalographic Recordings. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 894–900. MIT Press, 1998.
- [88] N. Kasthuri, K. J. Hayworth, D. R. Berger, R. L. Schalek, J. A. Conchello, S. Knowles-Barley, D. Lee, A. Vázquez-Reina, V. Kaynig, T. R. Jones, M. Roberts, J. L. Morgan, J. C. Tapia, H. S. Seung, W. G. Roncal, J. T. Vogelstein, R. Burns, D. L. Sussman, C. E. Priebe, H. Pfister, and J. W. Lichtman. Saturated Reconstruction of a Volume of Neocortex. *Cell*, 162(3):648–661, July 2015.

- [89] V. Kaynig, T. J. Fuchs, and J. M. Buhmann. Geometrical Consistent 3d Tracing of Neuronal Processes in ssTEM Data. In *Proceedings of the 13th International Conference on Medical Image Computing and Computer-assisted Intervention: Part II, MICCAI'10*, pages 209–216, Berlin, Heidelberg, 2010. Springer-Verlag.
- [90] J. P. Kelly and D. C. Van Essen. Cell structure and function in the visual cortex of the cat. *The Journal of Physiology*, 238(3):515–547, May 1974.
- [91] M. A. Kennard. Age and other factors in motor recovery from precentral lesions in monkeys. *American Journal of Physiology–Legacy Content*, 115(1):138–146, 1936.
- [92] S. T. Kitai and G. A. Bishop. Horseradish Peroxidase. In *Neuroanatomical Tract-Tracing Methods*, pages 263–277. Springer, Boston, MA, 1981.
- [93] S. Knowles-Barley, N. J. Butcher, I. A. Meinertzhagen, and J. D. Armstrong. Biologically inspired EM image alignment and neural reconstruction. *Bioinformatics*, 27(16):2216–2223, Aug. 2011.
- [94] K. Kokkinakis and P. C. Loizou. Using blind source separation techniques to improve speech recognition in bilateral cochlear implant patients. *J Acoust Soc Am*, 123(4):2379–2390, Apr. 2008.
- [95] N. Korogod, C. C. Petersen, and G. W. Knott. Ultrastructural analysis of adult mouse neocortex comparing aldehyde perfusion with cryo fixation. *eLife*, 4.
- [96] A. Kreshuk, U. Koethe, E. Pax, D. D. Bock, and F. A. Hamprecht. Automated Detection of Synapses in Serial Section Transmission Electron Microscopy Image Stacks. *PLoS ONE*, 9(2):e87351, Feb. 2014.
- [97] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics*, 2(1-2):83–97, Mar. 1955.
- [98] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, R. Funke, D. Gage, K. Harris, A. Heaford, J. Howland, L. Kann, J. Lehoczyk, R. LeVine, P. McEwan, K. McKernan, J. Meldrim, J. P. Mesirov, C. Miranda, W. Morris, J. Naylor, C. Raymond, M. Rosetti, R. Santos, A. Sheridan, C. Sougnez, N. Stange-Thomann, N. Stojanovic, A. Subramanian, D. Wyman, J. Rogers, J. Sulston, R. Ainscough, S. Beck, D. Bentley, J. Burton, C. Clee, N. Carter, A. Coulson, R. Deadman, P. Deloukas, A. Dunham, I. Dunham, R. Durbin, L. French, D. Grafham, S. Gregory, T. Hubbard, S. Humphray, A. Hunt, M. Jones, C. Lloyd, A. McMurray, L. Matthews, S. Mercer, S. Milne, J. C. Mullikin, A. Mungall, R. Plumb, M. Ross, R. Shownkeen, S. Sims, R. H. Waterston, R. K. Wilson, L. W. Hillier, J. D. McPherson, M. A. Marra, E. R. Mardis, L. A. Fulton, A. T. Chinwalla, K. H. Pepin, W. R. Gish, S. L. Chissoe, M. C. Wendl, K. D. Delehaunty, T. L. Miner, A. Delehaunty, J. B. Kramer, L. L. Cook, R. S. Fulton, D. L.

Johnson, P. J. Minx, S. W. Clifton, T. Hawkins, E. Branscomb, P. Predki, P. Richardson, S. Wenning, T. Slezak, N. Doggett, J.-F. Cheng, A. Olsen, S. Lucas, C. Elkin, E. Uberbacher, M. Frazier, R. A. Gibbs, D. M. Muzny, S. E. Scherer, J. B. Bouck, E. J. Sodergren, K. C. Worley, C. M. Rives, J. H. Gorrell, M. L. Metzker, S. L. Naylor, R. S. Kucherlapati, D. L. Nelson, G. M. Weinstock, Y. Sakaki, A. Fujiyama, M. Hattori, T. Yada, A. Toyoda, T. Itoh, C. Kawagoe, H. Watanabe, Y. Totoki, T. Taylor, J. Weissenbach, R. Heilig, W. Saurin, F. Artiguenave, P. Brottier, T. Bruls, E. Pelletier, C. Robert, P. Wincker, A. Rosenthal, M. Platzer, G. Nyakatura, S. Taudien, A. Rump, D. R. Smith, L. Doucette-Stamm, M. Rubenfield, K. Weinstock, H. M. Lee, J. Dubois, H. Yang, J. Yu, J. Wang, G. Huang, J. Gu, L. Hood, L. Rowen, A. Madan, S. Qin, R. W. Davis, N. A. Federspiel, A. P. Abola, M. J. Proctor, B. A. Roe, F. Chen, H. Pan, J. Ramser, H. Lehrach, R. Reinhardt, W. R. McCombie, M. d. I. Bastide, N. Dedhia, H. Blöcker, K. Hornischer, G. Nordsiek, R. Agarwala, L. Aravind, J. A. Bailey, A. Bateman, S. Batzoglou, E. Birney, P. Bork, D. G. Brown, C. B. Burge, L. Cerutti, H.-C. Chen, D. Church, M. Clamp, R. R. Copley, T. Doerks, S. R. Eddy, E. E. Eichler, T. S. Furey, J. Galagan, J. G. R. Gilbert, C. Harmon, Y. Hayashizaki, D. Haussler, H. Hermjakob, K. Hokamp, W. Jang, L. S. Johnson, T. A. Jones, S. Kasif, A. Kasprzyk, S. Kennedy, W. J. Kent, P. Kitts, E. V. Koonin, I. Korf, D. Kulp, D. Lancet, T. M. Lowe, A. McLysaght, T. Mikkelsen, J. V. Moran, N. Mulder, V. J. Pollara, C. P. Ponting, G. Schuler, J. Schultz, G. Slater, A. F. A. Smit, E. Stupka, J. Szustakowki, D. Thierry-Mieg, J. Thierry-Mieg, L. Wagner, J. Wallis, R. Wheeler, A. Williams, Y. I. Wolf, K. H. Wolfe, S.-P. Yang, R.-F. Yeh, F. Collins, M. S. Guyer, J. Peterson, A. Felsenfeld, K. A. Wetterstrand, R. M. Myers, J. Schmutz, M. Dickson, J. Grimwood, D. R. Cox, M. V. Olson, R. Kaul, C. Raymond, N. Shimizu, K. Kawasaki, S. Minoshima, G. A. Evans, M. Athanasiou, R. Schultz, A. Patrinos, and M. J. Morgan. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, Feb. 2001.

- [99] Y. Li and D. Wang. On the optimality of ideal binary time–frequency masks. *Speech Communication*, 51(3):230–239, Mar. 2009.
- [100] J. W. Lichtman, H. Pfister, and N. Shavit. The big data challenges of connectomics. *Nat Neurosci*, 17(11):1448–1454, Nov. 2014.
- [101] J. W. Lichtman and J. R. Sanes. Ome sweet ome: what can the genome tell us about the connectome? *Curr Opin Neurobiol*, 18(3):346–353, June 2008.
- [102] V. Lohweg and H. Doerksen. UCI Machine Learning Repository: banknote authentication Data Set, Apr. 2013.
- [103] J. C. Magee and E. P. Cook. Somatic EPSP amplitude is independent of synapse location in hippocampal pyramidal neurons. *Nature Neuroscience*, 3(9):895–903, Sept. 2000.



- [104] H. Markram. The human brain project. *Scientific American*, 306(EPFL-ARTICLE-183377):50–5, 2012.
- [105] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.
- [106] N. T. McMullen, C. B. Smelser, and R. K. de Venecia. A quantitative analysis of parvalbumin neurons in rabbit auditory neocortex. *The Journal of Comparative Neurology*, 349(4):493–511, Nov. 1994.
- [107] D. L. McNeill, K. Chung, C. E. Hulsebosch, R. P. Bolender, and R. E. Coggeshall. Numbers of synapses in laminae I–IV of the rat dorsal horn. *The Journal of Comparative Neurology*, 278(3):453–460, Dec. 1988.
- [108] I. A. Meinertzhagen. The organisation of invertebrate brains: cells, synapses and circuits. *Acta Zoologica*, 91(1):64–71, Jan. 2010.
- [109] I. A. Meinertzhagen. Connectome studies on *Drosophila*: a short perspective on a tiny brain. *Journal of Neurogenetics*, 30(2):62–68, Apr. 2016.
- [110] T. Meynert. *Der Bau der Gross-Hirnrinde: und seine örtlichen Verschiedenheiten, nebst einem pathologisch-anatomischen Corollarium*. Heuser, 1868.
- [111] T. Miki, Y. Fukui, M. Itoh, S. Hisano, Q. Xie, and Y. Takeuchi. Estimation of the numerical densities of neurons and synapses in cerebral cortex. *Brain Research Protocols*, 2(1):9–16, Dec. 1997.
- [112] M. Minsky and S. Papert. *Perceptrons*. 1969.
- [113] Y. Mishchenko. On Optical Detection of Densely Labeled Synapses in Neuropil and Mapping Connectivity with Combinatorially Multiplexed Fluorescent Synaptic Markers. *PLoS ONE*, 5(1):e8853, Jan. 2010.
- [114] Y. Mishchenko, T. Hu, J. Spacek, J. Mendenhall, K. M. Harris, and D. B. Chklovskii. Ultrastructural Analysis of Hippocampal Neuropil from the Connectomics Perspective. *Neuron*, 67(6):1009–1020, Sept. 2010.
- [115] J. L. Morgan and J. W. Lichtman. Why not connectomics? *Nat Methods*, 10(6):494–500, June 2013.
- [116] V. B. Mountcastle. Modality and topographic properties of single neurons of cat’s somatic sensory cortex. *J neurophysiol*, 20(4):408–434, 1957.
- [117] P. R. Mouton, D. L. Price, and L. C. Walker. Empirical assessment of synapse numbers in primate neocortex. *Journal of Neuroscience Methods*, 75(2):119–126, Aug. 1997.

- [118] J. Nagi, F. Ducatelle, G. Di Caro, D. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 342–347, 2011.
- [119] S. Navlakha, J. Suhan, A. L. Barth, and Z. Bar-Joseph. A high-throughput framework to detect synapses in electron microscopy images. *Bioinformatics*, 29(13):i9–i17, July 2013.
- [120] J. Ng, A. Browning, L. Lechner, M. Terada, G. Howard, and G. Jefferis. Genetically targeted 3d visualisation of *Drosophila* neurons under Electron Microscopy and X-Ray Microscopy using miniSOG. Dec. 2016.
- [121] L.-T. Nguyen, A. Belouchrani, K. Abed-Meraim, and B. Boashash. Separating more sources than sensors using time-frequency distributions. In *Proceedings of the Sixth International Symposium on Signal Processing and its Applications (Cat.No.01EX467)*, volume 2, pages 583–586 vol.2, 2001.
- [122] S. W. Oh, J. A. Harris, L. Ng, B. Winslow, N. Cain, S. Mihalas, Q. Wang, C. Lau, L. Kuan, A. M. Henry, M. T. Mortrud, B. Ouellette, T. N. Nguyen, S. A. Sorensen, C. R. Slaughterbeck, W. Wakeman, Y. Li, D. Feng, A. Ho, E. Nicholas, K. E. Hirokawa, P. Bohn, K. M. Joines, H. Peng, M. J. Hawrylycz, J. W. Phillips, J. G. Hohmann, P. Wohnoutka, C. R. Gerfen, C. Koch, A. Bernard, C. Dang, A. R. Jones, and H. Zeng. A mesoscale connectome of the mouse brain. *Nature*, 508(7495):207–214, Apr. 2014.
- [123] J. O’Kusky and M. Colonnier. A laminar analysis of the number of neurons, glia, and synapses in the visual cortex (area 17) of adult macaque monkeys. *The Journal of Comparative Neurology*, 210(3):278–290, Sept. 1982.
- [124] J. O’Kusky and M. Colonnier. Postnatal changes in the number of neurons and synapses in the visual cortex (area 17) of the macaque monkey: A stereological analysis in normal and monocularly deprived animals. *The Journal of Comparative Neurology*, 210(3):291–306, Sept. 1982.
- [125] J. R. O’Kusky. Postnatal changes in the numerical density and total number of asymmetric and symmetric synapses in the hypoglossal nucleus of the rat. *Developmental Brain Research*, 108(1–2):179–191, June 1998.
- [126] B. Pakkenberg and H. J. G. Gundersen. Neocortical neuron number in humans: Effect of sex and age. *The Journal of Comparative Neurology*, 384(2):312–320, 1997.
- [127] G. Paxinos and C. Watson. *The Rat Brain in Stereotaxic Coordinates: Hard Cover Edition*. Academic Press, Feb. 2007.
- [128] M. S. Pedersen. *Source separation for hearing aid applications*. IMM, Informatik og Matematisk Modelling, DTU, 2006.

- [129] C. Perez-Cruz, M. Simon, G. Flügge, E. Fuchs, and B. Czéh. Diurnal rhythm and stress regulate dendritic architecture and spine density of pyramidal neurons in the rat infralimbic cortex. *Behavioural Brain Research*, 205(2):406–413, Dec. 2009.
- [130] A. Peters and M. L. Feldman. The projection of the lateral geniculate nucleus to area 17 of the rat cerebral cortex. I. General description. *J Neurocytol*, 5(1):63–84, Feb. 1976.
- [131] S. M. Plaza, T. Parag, G. B. Huang, D. J. Olbris, M. A. Saunders, and P. K. Rivlin. Annotating Synapses in Large EM Datasets. *arXiv:1409.1801 [cs, q-bio]*, Sept. 2014. arXiv: 1409.1801.
- [132] C. Poth, C. Fung, O. Güntürkün, S. Ridgway, and H. Oelschläger. Neuron numbers in sensory cortices of five delphinids compared to a physeterid, the pygmy sperm whale. *Brain Research Bulletin*, 66(4–6):357–360, Sept. 2005.
- [133] T. P. S. Powell. Certain aspects of the intrinsic organisation of the cerebral cortex. *Brain mechanisms and perceptual awareness*, pages 1–19, 1981.
- [134] J. R. Quinlan. Learning With Continuous Classes. pages 343–348. World Scientific, 1992.
- [135] N. Randel, A. Asadulina, L. A. Bezares-Calderón, C. Verasztó, E. A. Williams, M. Conzelmann, R. Shahidi, and G. Jékely. Neuronal connectome of a sensory-motor circuit for visual navigation. *Elife*, 3:e02730, 2014.
- [136] Q. Rao, H. Han, W. Li, L. Shen, X. Chen, and Q. Xie. Automatically segmenting and reconstructing neurons in SEM images. In *2016 IEEE International Conference on Mechatronics and Automation*, pages 1908–1915, Aug. 2016.
- [137] A. Renart, J. d. l. Rocha, P. Bartho, L. Hollender, N. Parga, A. Reyes, and K. D. Harris. The Asynchronous State in Cortical Circuits. *Science*, 327(5965):587–590, Jan. 2010.
- [138] M. Revow. delve twonorm dataset, Apr. 1996.
- [139] A. A. C. M. Ribeiro, C. Davis, and G. Gabella. Estimate of size and total number of neurons in superior cervical ganglion of rat, capybara and horse. *Anat Embryol*, 208(5):367–380, Aug. 2004.
- [140] S. Rickard, R. Balan, and J. Rosca. Real-time time-frequency based blind source separation. *aje*, 2:1, 2001.
- [141] C. Riday, S. Bhargava, R. H. Hahnloser, and S.-C. Liu. Monaural Source Separation Using a Random Forest Classifier. In *INTERSPEECH*, pages 3344–3348, 2016.

- [142] M. Robnik-Šikonja. Improving Random Forests. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, J.-F. Boulicaut, F. Esposito, F. Giannotti, and D. Pedreschi, editors, *Machine Learning: ECML 2004*, volume 3201, pages 359–370. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.
- [143] A. J. Rockel, R. W. Hiorns, and T. P. S. Powell. The Basic Uniformity in Structure of the Neocortex. *Brain*, 103(2):221–244, June 1980.
- [144] S. Rosen. Processing of complex sounds by the auditory system - Temporal information in speech: acoustic, auditory and linguistic aspects. *Phil. Trans. R. Soc. Lond. B*, 336(1278):367–373, June 1992.
- [145] F. Rosenblatt. The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain. *Psychological Review*, pages 65–386, 1958.
- [146] D. K. Sarko, K. C. Catania, D. B. Leitch, J. H. Kaas, and S. Herculano-Houzel. Cellular scaling rules of insectivore brains. *Front. Neuroanat.*, 3:8, 2009.
- [147] R. J. Sayer, M. J. Friedlander, and S. J. Redman. The time course and amplitude of EPSPs evoked at synapses between pairs of CA3/CA1 neurons in the hippocampal slice. *Journal of Neuroscience*, 10(3):826–836, 1990.
- [148] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D. J. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona. Fiji: an open-source platform for biological-image analysis. *Nat Meth*, 9(7):676–682, July 2012.
- [149] M. N. Schmidt, J. Larsen, and F. T. Hsiao. Wind Noise Reduction using Non-Negative Sparse Coding. In *2007 IEEE Workshop on Machine Learning for Signal Processing*, pages 431–436, Aug. 2007.
- [150] C. M. Schneider-Mizell, S. Gerhard, M. Longair, T. Kazimiers, F. Li, M. F. Zwart, A. Champion, F. M. Midgley, R. D. Fetter, S. Saalfeld, and others. Quantitative neuroanatomy for connectomics in *Drosophila*. *Elife*, 5:e12059, 2016.
- [151] A. Schüz and G. Palm. Density of neurons and synapses in the cerebral cortex of the mouse. *The Journal of Comparative Neurology*, 286(4):442–455, Aug. 1989.
- [152] J. Seifter, D. Sloane, and A. Ratner. *Concepts in Medical Physiology*. Lippincott Williams & Wilkins, 2005. Google-Books-ID: A8H\_9S4E0I4C.
- [153] C. S. Sherrington. *The Central Nervous System, vol. 3 of A Textbook of Physiology, M. Foster, Ed.* MacMillan, London, ed, 1897.

- [154] X. Shu, V. Lev-Ram, T. J. Deerinck, Y. Qi, E. B. Ramko, M. W. Davidson, Y. Jin, M. H. Ellisman, and R. Y. Tsien. A Genetically Encoded Tag for Correlated Light and Electron Microscopy of Intact Cells, Tissues, and Organisms. *PLoS Biol*, 9(4):e1001041, Apr. 2011.
- [155] Y. M. Sigal, C. M. Speer, H. P. Babcock, and X. Zhuang. Mapping Synaptic Input Fields of Neurons with Super-Resolution Imaging. *Cell*, 163(2):493–505, Oct. 2015.
- [156] V. G. Sigillito, S. P. Wing, L. V. Hutton, and K. B. Baker. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10(3):262–266, 1989.
- [157] T. Skoglund, R. Pascher, and C.-H. Berthold. Heterogeneity in the columnar number of neurons in different neocortical areas in the rat. *Neuroscience Letters*, 208(2):97–100, Apr. 1996.
- [158] K. Smith, P. Fua, R. Achanta, V. Lepetit, and A. Lucchi. A Fully Automated Approach to Segmentation of Irregularly Shaped Cellular Structures in EM Images. *Proc. Medical Image Computing and Computer Assisted Intervention (MICCAI 2010), Part II*, LNCS 6362, 2010.
- [159] P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. Technical report, COLORADO UNIV AT BOULDER DEPT OF COMPUTER SCIENCE, 1986.
- [160] S. Srinivasan and C. F. Stevens. A quantitative description of the mouse piriform cortex. *bioRxiv*, page 099002, Jan. 2017.
- [161] A. Stepanyants, P. R. Hof, and D. B. Chklovskii. Geometry and Structural Plasticity of Synaptic Connectivity. *Neuron*, 34(2):275–288, Apr. 2002.
- [162] A. Stepanyants, L. M. Martinez, A. S. Ferecskó, and Z. F. Kisvárdy. The fractions of short- and long-range connections in the visual cortex. *PNAS*, 106(9):3555–3560, Mar. 2009.
- [163] D. C. Sterio. The unbiased estimation of number and sizes of arbitrary particles using the disector. *Journal of Microscopy*, 134(2):127–136, May 1984.
- [164] G. F. Striedter, T. G. Belgard, C.-C. Chen, F. P. Davis, B. L. Finlay, O. Güntürkün, M. E. Hale, J. A. Harris, E. E. Hecht, P. R. Hof, H. A. Hofmann, L. Z. Holland, A. N. Iwaniuk, E. D. Jarvis, H. J. Karten, P. S. Katz, W. B. Kristan, E. R. Macagno, P. P. Mitra, L. L. Moroz, T. M. Preuss, C. W. Ragsdale, C. C. Sherwood, C. F. Stevens, M. C. Stüttgen, T. Tsumoto, and W. Wilczynski. NSF Workshop Report: Discovering General Principles of Nervous System Organization by Comparing Brain Maps across Species. *BBE*, 83(1):1–8, 2014.

- [165] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen. An Algorithm for Intelligibility Prediction of Time-Frequency Weighted Noisy Speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, Sept. 2011.
- [166] T. Templier, K. Bektas, and R. H. Hahnloser. Eye-Trace: Segmentation of Volumetric Microscopy Images with Eyegaze. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 5812–5823, New York, NY, USA, 2016. ACM.
- [167] D. B. Tower. Structural and functional organization of mammalian cerebral cortex: The correlation of neurone density with brain size. Cortical neurone density in the fin whale (*Balaenoptera Physalus L.*) with a note on the cortical neurone density in the Indian elephant. *The Journal of Comparative Neurology*, 101(1):19–51, 1954.
- [168] R. Tremblay, S. Lee, and B. Rudy. GABAergic Interneurons in the Neocortex: From Cellular Properties to Circuits. *Neuron*, 91(2):260–292, July 2016.
- [169] A. Turiel, G. Mato, N. Parga, and J.-P. Nadal. Self-Similarity Properties of Natural Images Resemble Those of Turbulent Flows. *Phys. Rev. Lett.*, 80(5):1098–1101, Feb. 1998.
- [170] A. M. Turner and W. T. Greenough. Differential rearing effects on rat visual cortex synapses. I. Synaptic and neuronal density and synapses per neuron. *Brain Research*, 329(1–2):195–203, Mar. 1985.
- [171] A. Ungersböck, R. Kretz, and G. Rager. Synaptogenesis in the primary visual cortex of the tree shrew (*Tupaia belangeri*). *The Journal of Comparative Neurology*, 308(3):491–504, Oct. 2004.
- [172] P. Vanroose. *Blind Source Separation of Speech and Background Music for Improved Speech Recognition*.
- [173] R. Vigarío, J. Sarela, V. Jousmiki, M. Hamalainen, and E. Oja. Independent component approach to the analysis of EEG and MEG recordings. *IEEE Transactions on Biomedical Engineering*, 47(5):589–593, May 2000.
- [174] C. v. Vreeswijk and H. Sompolinsky. Chaos in Neuronal Networks with Balanced Excitatory and Inhibitory Activity. *Science*, 274(5293):1724–1726, Dec. 1996.
- [175] G. Vrensen, D. De Groot, and J. Nunes-Cardozo. Postnatal development of neurons and synapses in the visual and motor cortex of rabbits: A quantitative light and electron microscopic study. *Brain Research Bulletin*, 2(6):405–416, Dec. 1977.
- [176] Y. Wang, C. Song, and S.-T. Xia. Unifying Decision Trees Split Criteria Using Tsallis Entropy. *arXiv:1511.08136 [cs, stat]*, Nov. 2015. arXiv: 1511.08136.

- [177] A. A. Wanner, C. Genoud, and R. W. Friedrich. 3-dimensional electron microscopic imaging of the zebrafish olfactory bulb and dense reconstruction of neurons. *Scientific Data*, 3:sdata2016100, Nov. 2016.
- [178] A. A. Wanner, C. Genoud, T. Masudi, L. Siksou, and R. W. Friedrich. Dense EM-based reconstruction of the interglomerular projectome in the zebrafish olfactory bulb. *Nat Neurosci*, 19(6):816–825, June 2016.
- [179] A. A. Wanner, M. A. Kirschmann, and C. Genoud. Challenges of microtome-based serial block-face scanning electron microscopy in neuroscience. *Journal of Microscopy*, 259(2):137–142, Aug. 2015.
- [180] M. A. Warren and K. S. Bedi. Synapse-to-neuron ratios in the visual cortex of adult rats undernourished from about birth until 100 days of age. *The Journal of Comparative Neurology*, 210(1):59–64, Sept. 1982.
- [181] S. G. Waugh. *Extending and benchmarking Cascade-Correlation : extensions to the Cascade-Correlation architecture and benchmarking of feed-forward supervised artificial neural networks*. phd, University of Tasmania, 1995.
- [182] T. A. Weissman and Y. A. Pan. Brainbow: New Resources and Emerging Biological Applications for Multicolor Genetic Labeling and Analysis. *Genetics*, 199(2):293–306, Feb. 2015.
- [183] P. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Harvard University, 1974. Google-Books-ID: 2MsgnQEACAAJ.
- [184] J. G. White, E. Southgate, J. N. Thomson, and S. Brenner. The Structure of the Nervous System of the Nematode *Caenorhabditis elegans*. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 314(1165):1–340, Nov. 1986.
- [185] S. R. y Cajal. *Estructura de los centros nerviosos de las aves*. 1888.
- [186] S. R. y Cajal. *Recollections of my life*, volume 8. MIT Press, 1989.
- [187] I. C. Yeh. Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete Research*, 28(12):1797–1808, Dec. 1998.
- [188] N. Zecevic, J.-P. Bourgeois, and P. Rakic. Changes in synaptic density in motor cortex of rhesus monkey during fetal and postnatal life. *Developmental Brain Research*, 50(1):11–32, Nov. 1989.
- [189] T. Zeng, B. Wu, and S. Ji. DeepEM3d: approaching human-level performance on 3d anisotropic EM image segmentation. *Bioinformatics*.

# Curriculum Vitae

Cosimo Riday  
Born: 24 September 1985  
Nationality: USA

## Education

---

|           |  |
|-----------|--|
| 2017-     | Teaching Diploma in Computer Science in Computer Science ETH         |
| 2017-     | Teaching Diploma in in Mathematics ETH                               |
| 2011-2018 | Doctoral studies at the Institute of Neuroinformatics UZH/ETH Zurich |
| 2009-2010 | Master of Science ETH in Computational Science and Engineering       |
| 2006-2009 | Bachelor of Science ETH in Computational Science and Engineering     |
| 2003-2006 | Electrical Engineering ETH (cancelled study)                         |
| 1999-2003 | Gymnasium MNG Rämibühl   |

## Study projects

---

|                                       |   |
|---------------------------------------|---|
| PhD Thesis                            | A Generalist Approach to Learning and to Biological Neuron Networks |
| Master Thesis                         | The Influence of Valley Geometry on the Diurnal Valley Winds        |
| Bachelor Thesis &<br>Semester Project | Simulation of Aggregation and Fragmentation under Shear             |