

SPATIAL ANALYSIS OF COMPLEX BIOLOGICAL TISSUES
FROM SINGLE CELL GENE EXPRESSION DATA

Clustering and visualizing fonctionnal tissues in *P. dumerillii*

JEAN-BAPTISTE OLIVIER GEORGES PETTIT



UNIVERSITY OF
CAMBRIDGE

2014 – version 0.9

CONTENTS

I	SPATIAL ANALYSIS OF COMPLEX BIOLOGICAL TISSUES FROM SINGLE CELL GENE EXPRESSION DATA	13
1	CAPTURING GENE EXPRESSION IN <i>platynereis dumerilii</i> 'S BRAIN	15
1.1	Platynereis dumerilii, an ideal organism of brain development studies	15
1.1.1	General description	15
1.1.2	Larval development	16
1.2	Gene expression in Platynereis' developing brain	18
1.2.1	Platynereis' nervous development until 48hpf	18
1.2.2	Spatial organization of complex biological tissues like the brain	19
1.2.3	Generalities about gene expression and development	20
1.3	Capturing gene expression in the laboratory	22
1.3.1	In-situ hybridization assays	22
1.3.2	Building a image library of gene expression for Platynereis	23
1.3.3	RNA sequencing	24
1.4	Conclusions	25
2	FROM TISSUE TO SINGLE CELL TRANSCRIPTOMICS, A PARADIGM SHIFT	27
2.1	Spatially referenced single cell-like in-situ hybridization data	27
2.1.1	Dividing images into "cells"	27
2.1.2	A simple cell model, the "cube" data	28
2.2	Singe cell RNA sequencing, building a map of the full transcriptome	29
2.2.1	Sequencing single cell RNA contents	29
2.2.2	Mapping back gene expression to a spatial reference	30
2.3	About the quantitative trait of single cell expression data	30
2.3.1	Light contamination in in-situ hybridization data	30
2.3.2	Technical noise in single cell RNA-seq data	32
2.3.3	Conclusions	32

2.4	Binarizing gene expression datasets	32
2.4.1	Binarizing in-situ hybridization datasets	32
2.4.2	Binarizing whole transcriptomes	35
2.5	Preliminary results on single cell RNA-seq spatial back-mapping	36
2.5.1	Single cell RNA-seq in Platynereis' brain	36
2.5.2	Mapping back RNA-seq data back to PrimR in-situ hybridization assays	37
2.6	Conclusions	39
3	CLUSTERING AND VISUALIZING TISSUES FROM 3D SINGLE CELL EXPRESSION DATA	43
3.1	Elements of clustering for biological tissues	43
3.1.1	Motivations	43
3.1.2	General considerations about clustering	44
3.2	Visualizing clustering results in 3D with bioWeb3D	45
3.2.1	Background	45
3.2.2	Implementation	46
3.2.3	Results and Discussion	48
3.2.4	Conclusions	50
3.2.5	Availability and requirements	51
3.3	Non spatial clustering methods	51
3.3.1	Hierarchical clustering	51
3.3.2	Other clustering methods adapted to gene expression data	53
3.4	Discussion	53
3.4.1	Spatial clustering techniques (hierarchical, model based)	53
3.4.2	Method chosen	54
4	HIDDEN MARKOV RANDOM FIELDS FOR BIOLOGICAL DATA CLUSTERING	55
4.1	Markov random fields	55
4.1.1	Neighbourhood systems	55
4.1.2	Gibbsian prior distribution	56
4.1.3	Single and multiple beta models in a biological context	57
4.1.4	Summary of the prior distribution parameters	58
4.2	The emission model	58
4.2.1	Conditional independence in the observed data	58
4.2.2	Full likelihood of the Hidden Markov random field model	59
4.3	Parameter estimation using the EM algorithm	60

4.4	Mean field approximations	61
4.5	Maximization	62
4.6	Estimating K	63
4.7	Summary	63
5	METHOD VALIDATION AND PERFORMANCE ANALYSIS ON SIMULATED DATA	65
5.1	Simulating data with a spatial component	65
5.1.1	Simulating non spatial gene expression data	66
5.1.2	Introducing a known spatial context	67
5.1.3	Expected results	67
5.2	Comparing clustering results using the Jaccard similarity coefficient	67
5.2.1	Theoretical problem in comparing clustering re- sults	67
5.2.2	Alignment via similarity-specificity matrix	68
5.3	Validation of parameters estimation and model choos- ing	69
5.3.1	Estimation of Θ	69
5.3.2	Estimation of beta	69
5.3.3	Choosing K	72
5.4	Method performance and initialization	73
5.4.1	The EM principle and local maximum	73
5.4.2	Random initialization vs Hclust initialization	74
5.5	Method performance compared to Hclust and indepen- dent mixture models	74
5.5.1	Results of comparison	74
5.5.2	Discussion	75
II	APPENDIX	77
A	INPUT FILE FORMATS FOR BIOWEB3D	79
A.1	Dataset file specification	79
A.1.1	JSON format	79
A.1.2	XML format	79
A.1.3	CSV format	81
A.2	Information layer file specification	82
A.2.1	JSON format	82
A.2.2	XML format	83
	BIBLIOGRAPHY	87

LIST OF FIGURES

Figure 1	<i>Platynereis dumerillii</i> 's larva and adult forms. 15
Figure 2	<i>Platynereis dumerillii</i> 's larvea development at 48hpf or late trochophore. Striped in red is indicated the area which forms the developing brain of the larvae. 17
Figure 3	<i>Platynereis dumerillii</i> 's stereotypical and synchronous development. In green and red are two different <i>P. dumerillii</i> individuals' with the same gene expression being highlighted. They show extremely similar patterns of development. 18
Figure 4	Cell types anatomical heterogeneity, gene expression and protein translation and gene regulatory networks. The schematics shows that genes in the DNA are transcribed to RNA molecules that are further translated outside the nucleus into proteins. Those proteins can serve various purposes inside the cell or come back to the nucleus to regulate gene expression. 22
Figure 5	Fluorescent in-situ hybridization assays to create a 169 genes catalogue of gene expression in the brain of <i>P. dumerillii</i> . From the live tissue cut into thin fixed layers, every slice is stained with a reference gene and a gene of interest that will reveal the areas of expression under fluorescent microscopy. The process repeated 169 times for key genes in <i>P. dumerillii</i> development has been generated by [78] 24
Figure 6	Errors introduced by the "cube" cell model. Path A shows how regions with highly expressed genes can introduce errors through light contamination. Path B shows how some ubes may appear artificially void of expression because of the not even distribution of transcripts inside the cytoplasm especially for large cells. 29

- Figure 7 **Light contamination in in-situ hybridization luminescence data seen with the example of the gene *Ascl*.** Panel A shows the raw fluorescent microscopy capture of the gene's expression for one layer in the brain of *Platynereis*. Panel B shows the light intensity measured along the red line in panel A. Because of the small scale of study, cells surrounded by other cells expressing a particular gene will have a higher intensity values because of nearby light contamination. 31
- Figure 8 Dilution series of total *A. thaliana* RNA 33
- Figure 9 **Densities of log luminescence values for two genes (*rOpsin*, *PRDM8*) over the 32,302 cells.** For *rOpsin*, the density exhibits two clear peaks making the choice of a binarizing threshold easy. By contrast, for *PRDM8* there is no such clear threshold, making an automated binarization method hard to implement. 34
- Figure 10 Thresholding RNA sequencing data for *P. dumerillii* 36
- Figure 11 Regions defined by the expression overlap of the top 3 scoring genes in [78] binarized in-situ hybridization data. The red colour shows the co-expression of the three considered genes, the blue areas are those where one or more of the three considered genes are expressed but not all, in grey are the areas where none of the considered genes are expressed. The 4 figures are from a apical view with the dorsal side on top. 41
- Figure 12 bioWeb3D allows several datasets to be visualized at the same time in up to 4 different "worlds" 46

- Figure 13 The 3D location of cells within the brain of the marine annelid *Platynereis dumerillii* is shown. Two classes are displayed (in green and blue) along with the shadow of the remaining cells. The User interface is visible on the right of the screen and can be hidden. Data for this figure was taken from [78], see 1 for a presentation of *P. dumerillii* and chapter 2 for detailed presentation the data. 48
- Figure 14 The three control panels to control visualization in bioWeb3D. A: the datasets panel, where new datasets and new information layer files can be inputted. From the dataset panel, information layers can be selected and unselected, and datasets attached or detached from the “worlds”. B: the view panel where the user can choose the worlds to display. C: the settings panel where the user can customize visualization. 49
- Figure 15 Dendrogram outputted by the hClust clustering method. This is the full dendrogram, to get the resulting clustering partition, the tree must be cut. The red line shows the cut needed to get 7 clusters out of the dendrogram. Importantly, a dendrogram does not provide any rational about the best number of clusters. 52
- Figure 16 **First order neighbouring system.** In the chosen first order neighbouring system, each node in the graph (representing a “cube” \approx cell in the data) is linked to a maximum of 6 other cubes. The Markov property on the graph implies that the state of any node (the red one for example) can be fully determined by knowing the state of its neighbours (the orange ones). 56

- Figure 17 **Simulation scheme used to generate gene expression data with a spatial component and known parameters.** The values of Θ are used to generate a dataset of clusters with the same gene expression profile as the reference. Each simulated cell is then assigned to its corresponding spatial localization so that the simulated data keeps the spatial component of the biological data. 66
- Figure 18 **Validating the estimation of Θ for $K = 6$.** On the x axis are shown the 6 clusters obtained after clustering the simulated data. On the y axis are shown the 6 “true” reference clusters. Each cell of the heatmap corresponds to the mean of the pairwise (as regard to the 86 genes considered) difference between “true” and simulated Θ values. A small number means that the difference between the reference Θ values and the ones obtained after clustering the simulated data is very small. 70
- Figure 19 **Validating the estimation of β .** This figure shows the evolution for $K \in [4, 80]$ of the mean value of β across all the clusters. The red dots represent the biological data clustering (i.e the reference in our simulations scheme). The green dots represent the results obtained after clustering simulated data, which shows an underestimation of β . To confirm that this underestimation come from the simulation scheme and not the clustering method, the simulated data was used as the reference to generate a “second generation” of simulated data, suppressing the simulation scheme bias (see Figure 19). The results of this re-simulation are shown by the blue dots, which exhibit no underestimation of β . Finally the brown dots represent the mean value of β on the same simulated data but spatially randomized, as expected the β are now estimated to 0. 71

Figure 20

Decrease in spatial coherency due to the simulation scheme. For an example cluster h , gene m may only be expressed in half of the cells. This will yield $\theta_{h,m} = 0.5$. However, in the biological data, the cells expressing gene m may be spatially coherent (i.e., located close to one another), leading to a reduced area of expression discontinuity (the green line). By contrast, in the simulated data the expression of such a gene will lose its spatial coherency, leading to an increased area of expression discontinuity. The number of cells having a neighbour with some differences in the gene expression pattern is directly linked to the value of β_h through the energy function described in chapter 4. This explains the underestimation of β observed in Figure 18. 72

Figure 21

Jaccard coefficient between "true" and resulting clusters on the simulated data with different methods and initializations. Panel A compares the performance of the MRF method with a randomly initialization with an independent mixture model also with a random initialization, the MRF method initialized with the hClust classification and hClust alone on data simulated with a spatial component. Panel B shows the Jaccard coefficient for the MRF method and independent mixture model both with a random initialization; in this case both methods are applied to simulated data that lacks a spatial component. 73

Figure 22 **Jaccard coefficient between “true” and resulting clusters on the simulated data with different methods and initializations.** Panel A compares the performance of the MRF method with a randomly initialization with an independent mixture model also with a random initialization, the MRF method initialized with the hClust classification and hClust alone on data simulated with a spatial component. Panel B shows the Jaccard coefficient for the MRF method and independent mixture model both with a random initialization; in this case both methods are applied to simulated data that lacks a spatial component. 75

LIST OF TABLES

Table 1 Top 3 most specific genes for 4 sequenced cells and the potential tissue they belong to. The resulting localization of those four cells inferred from the in-situ hybridization data are shown on figure 11. 39

LISTINGS

Listing 1	Json dataset file	80
Listing 2	XML dataset file	81
Listing 3	CSV dataset file	82
Listing 4	JSON information layer file	83
Listing 5	XML information layer file	84

Listing 6 CSV information layer file 85

ACRONYMS

Part I

SPATIAL ANALYSIS OF COMPLEX BIOLOGICAL TISSUES FROM SINGLE CELL GENE EXPRESSION DATA

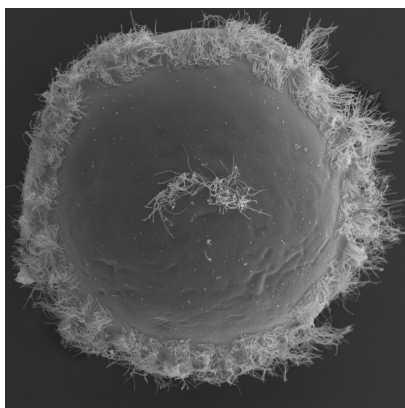
CAPTURING GENE EXPRESSION IN *PLATYNEREIS DUMERILLII*'S BRAIN

1.1 *PLATYNEREIS DUMERILLII*, AN IDEAL ORGANISM OF BRAIN DEVELOPMENT STUDIES

1.1.1 *General description*

Platynereis dumerillii is a marine annelid of the class Polychaeta, it has been established as one of the main marine animal models in the fields of evolutionary, developmental biology as well as ecology, toxicology and neurobiology [35, 76, 32, 22, 25, 26].

P. dumerillii populates shallow (no more than 3m deep) hard ocean floors around the world. It is commonly found in the Mediterranean sea, the north Atlantic coast of Europe as well as in the shallow seas surrounding Sri Lanka, Java and the Philippines. Eggs, embryos and larvae are roughly 160 μm while the adults can measure up to 6cm in length.



(a) Larval form of *P. dumerillii*. Image: MPI for Developmental Biology.



(b) Adult *P. dumerillii*. Image: Arendt group, EMBL

Figure 1.: *Platynereis dumerillii*'s larva and adult forms.

There are several reasons why *P. dumerillii* has been chosen as a model by numerous laboratories. Indeed, evolutionary wise, *P. dumerillii* shows several interesting characteristics. As a member of the bilaterians *P. dumerillii* has a defined bilateral symmetry. It belongs to

the lophotrochozoan taxon of the bilaterians as opposed to most of the well established model animals which either belong to the ecdysozoans (*Caenorhabditis elegans*, *Drosophila melanogaster*) or the deuterostomes (mouse, human). Lophotrochozoans being extremely under represented, *P. dumerillii* as a model organism is essential to be able to use comparative approaches on bilaterians [26].

P. dumerillii also exhibits an exceptionally slow evolutionary lineage. It has even been described as a "living fossil" for that reason [26]. Therefore, the numerous ancestral developmental characteristics exhibited by *P. dumerillii* translate into an image of the common past of all bilaterians. For example, an interesting example described in [20, 77] is the conserved molecular topography of the genes responsible for the development of the central nervous system between *P. dumerillii* and all vertebrates. This slow evolutionary makes *P. dumerillii* a better comparison with vertebrates than fast evolving species like *Drosophila* and nematodes where derived features can obscure evolutionary signal [26?].

In terms of practicality, *P. dumerillii* can easily be kept and bred in captivity producing offspring throughout the year [25]. The behavioural characteristics of *P. dumerillii*'s mating ritual have been well studied. The "nuptial dance" happens on the water surface. Male and female respectively release the sperm and eggs synchronously. This activity is synchronized by pheromones released into the water [85]. Over 2000 individuals can be produced within a single batch. Every new individual will undergo embryonic then larval development before reaching *P. dumerillii*'s adult form.

1.1.2 Larval development

Similarly to the other polychaetes, the larval development of *P. dumerillii* can be decomposed into three main anatomical stages: the trochophore, the metotrochophore and the nectochaete. The trochophore is spherical and moves thanks to equatorial belt of ciliated cells as well as an apical organ displaying a ciliary tuft [65, 56] as seen on figure 1a and schematically on figure 2. the metotrochophore stage is characterized by the development of a slightly elongated segmented trunk compared to that of the trochophore [31]. The next stage is

the nectochaete larvae that resembles the adult (figure 1b) in most of the traits especially with parapodial appendages used for swimming and crawling [31]. This traditional subdivision has been applied to *P. dumerillii* [34].

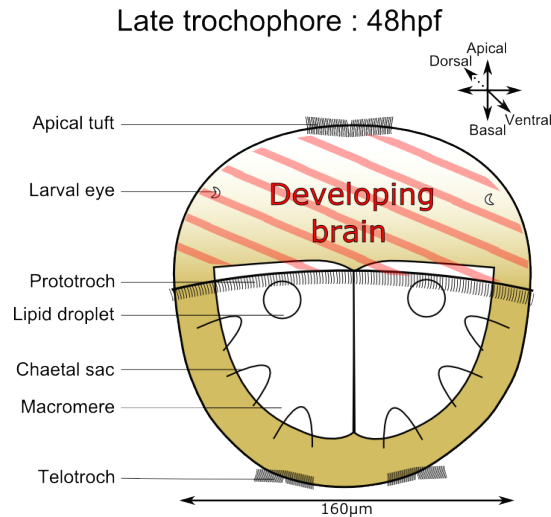


Figure 2.: *Platynereis dumerillii*'s larva development at 48hpf or late trochophore. Striped in red is indicated the area which forms the developing brain of the larvae.

Aside from this purely anatomical description, an additional staging system exists and has become the norm for current studies. The development is measured in *hours post fertilization* (hpf) at 18°C.

A key factor making *P. dumerillii* such an interesting model to work with is the fact that after fertilization, the ≈ 2000 larva will start developing at the exact same time, in a synchronous fashion. Furthermore, the larval development of *P. dumerillii* follows a very stereotypical pattern with very little variation from one individual to the other and even between batches provided the temperature is kept constant [25, 22]. An example showing the similarity between individuals during development can be seen on figure 3. This is a very important feature as it allows biologists to repeat experiments on several individuals at a very close developmental stage even if they are from different batches.

Describing the entire development of *P. dumerillii* does not fall within the scope of this thesis. Indeed, I will only be interested in the brain of *P. dumerillii*'s larvae at 48hpf. Therefore, it is important

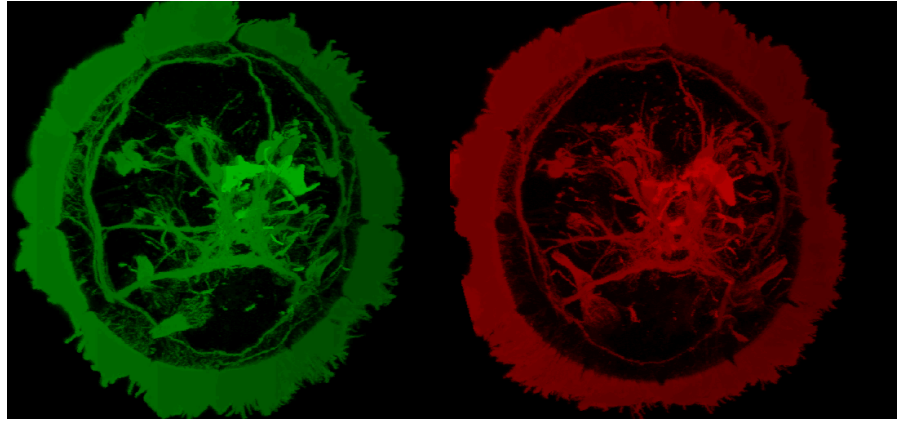


Figure 3.: *Platynereis dumerillii*'s stereotypical and synchronous development. In green and red are two different *P. dumerillii* individuals' with the same gene expression being highlighted. They show extremely similar patterns of development.

to have an anatomical idea of what the brain looks like at this time in development and what inherent characteristics will be the most interesting to investigate.

1.2 GENE EXPRESSION IN PLATYNEREIS' DEVELOPING BRAIN

1.2.1 *Platynereis*' nervous development until 48hpf

The main purpose of this thesis is not to fully understand the patterns of development in *P. dumerillii*'s larval brain. Therefore I will only give a brief summary of what the main component of the brain are at 48hpf, the time point I will be interested in, in the next chapters. *P. dumerillii*'s larval brain development is detailed in [26].

From the early trochophore (24-26hpf) neural system development starts taking place. The apical ganglion forms at the apical tuft. It contains one serotonergic cell and a few neurons linked to the nerve of the ciliary band of the larva called the prototroch (see figure 2). This allows the first movements of the larve thanks to the ciliated cells of the prototroch.

The mid-trochophore (26-40 hpf) sees the formation of the first cerebral commissure, it is a band of nerves interconnecting the ventral nerve cord and the brain. This trait is a typical feature of annelids' brains. During this phase the apical ganglion becomes bigger with three more serotonergic cells.

The late trochophore (40-48hpf) sees the formation of the second commissure in the ventral nerve cord. It is at the end of this stage that the tissues of the brain become more complex with a notable increase in the number of neurites [26].

The data used in the rest of this thesis will not encapsulate the whole larvae, just the developing brain (see figure 2) thus excluding the ventral part of the nervous system. The best studied areas of the developing brain are the larval eyes, the developing adult eyes and the apical organ on the dorsal side. On the ventral side are located the mushroom bodies, a pair of structures that are known to play a role in olfactory learning and memory in insects and annelids [78].

Even at this early stage in a relatively "simple" organism, the brain quickly becomes an extremely complex tissue. Cell types diverge and functional areas are formed. Before trying to understand more about *P. dumerillii*'s brain organization, it is interesting to ask the more general question of how complex tissues such as the brain are defined spatially.

1.2.2 *Spatial organization of complex biological tissues like the brain*

This section does not intend to demonstrate any specificity in the brain of *P. dumerillii*, it is meant to ask some of the fundamental questions that intrinsically motivate the work presented in the rest of this thesis. Complex tissues, the obvious example of which is the brain, could be viewed as an interconnected mosaic of cells having different functions, working together to achieve the global function of the organ.

When looking closely at this mosaic of cells, it is easy to observe that the spatial organisation is not random. Indeed, cells that serve the same function will often be close to each other, thus defining functional tissues. However, the spatial coherency of those tissues is not necessarily always the same. Some cell types may consist of cells that are scattered inside another more spatially coherent tissue. To illustrate that fact, an interesting example is the difference between the spatial coherency of cells forming the neuronal tissue in the brain and cells forming a well defined region in the brain like an exocrine gland. When asking the question: "is it likely that this particular cell is fully surrounded by cells belonging to the same cell type?", the

extensions created by the axons of neurones will decrease this probability. Indeed, axons will grow through other types of tissues to reach their destination [11, 17], making the overall spatial coherency of neural tissues smaller than very well spatially defined tissues.

When trying to analyse the full structure of the brain with an automated method, keeping in mind that fact could prove important to improve the results. This fact and its consequences on the work presented in this method are presented in chapters 4, 5 and add chapter 6.

So far, organs and cell types have been defined by their anatomical traits. But as mentioned in the introduction ?? the functional heterogeneity of complex tissues goes further than simple anatomical traits. As a result, I will be interested in traits that fundamentally define how cells are functioning.

1.2.3 Generalities about gene expression and development

Throughout this thesis the term *cell* will be referred to eukaryotic cells and more specifically those of multicellular organisms. Every cell in a complex organism possesses the same genome, that is, the sum of all the genetic information contained in the cell (nucleus and other compartments). This fundamental homogeneity is in plain contradiction with the heterogeneity observed anatomically. If every cell has the exact same DNA, where does the great variability between cell types come from (what makes a neurone become a neurone and not a pancreatic cell)? Answering this sort of questions defines the field of developmental biology.

The short and rather complete answer to any developmental biology question actually is: same genome but different pattern of gene expression. As gene expression is the central, most important, most studied cellular activity, it is indeed the common denominator of life, as large parts of the mechanisms making up gene expression are actually shared by every living creature known to man.

Of course, to understand what gene expression is, the notion of *gene* must first be defined. The precise definition of a gene is still controversial. The concept of a "*factor that conveys traits from parents*

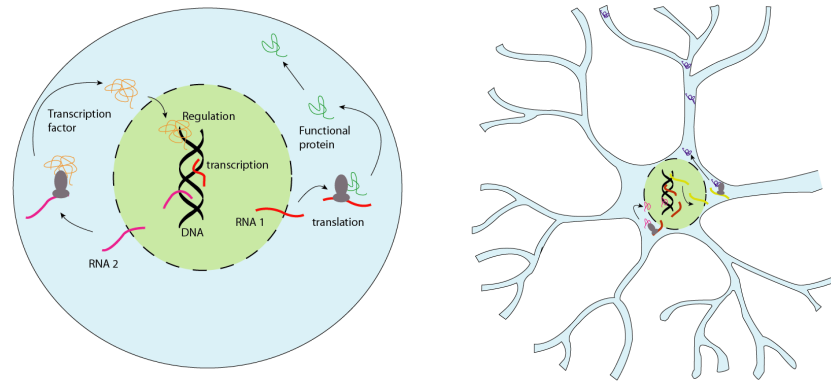
to offspring" was laid by Gregor Mendel in 1866 [51] when the accepted theory at the time was based on blending inheritance where the traits of the parents appeared mixed in the offspring following a continuous gradient. The most recently published definition of a gene followed the publication of the ENCODE project [24]. It states that a gene is "*a union of genomic sequences encoding a coherent set of potentially overlapping functional products.*"

Gene expression is the way cells express their genes. Expression of a gene is the process of transcribing the DNA of that particular gene. It is interesting to note that there are several ways to look at gene expression. Indeed in a cell or tissue, at a given time point it is possible to look whether a gene is expressed or not (binary expression) or how much a certain gene is expressed (quantitative expression). The product of gene expression is RNA molecules.

A portion of the RNA molecules are translated into proteins that can have very different purposes. Some will serve directly in the cellular life as functional/structural agents (elements of the ATP synthase for example [13]), others will be excreted by the cell and will serve a purpose at the scale of the organism [42] others called transcription factors will have a regulatory effect on gene expression [52]. In other terms the expression of gene G_a , coding for protein P_a might activate, accelerate, inactivate or decelerate the expression of gene G_b and potentially others. This outlines the complex interdependent regulatory system that is gene expression, see figure 4. For precise examples of gene regulation networks, see [29, 68, 28, 10].

During development, mechanisms exist that allow gene expression to become differential as the divisions occur. This is how the asymmetrical axis (dorso-ventral, and basal-apical) of the body are defined. The main mechanism involves chemical gradients. The first of these gradient has to come from the zygote which must contain some asymmetrically distributed chemical so that the first divisions lead to non identical cells. In the case of *Platynereis dumerillii*, the body axis are defined between 2hpf and 7hpf [26].

Gene expression is the key factor during tissue development. Therefore, the ability to study gene expression patterns has revolutionized the field of developmental biology. Technological innovation has been



(a) Undifferentiated cell with a circular shape. (b) Differentiated neuron, with typical axons development.

Figure 4.: Cell types anatomical heterogeneity, gene expression and protein translation and gene regulatory networks. The schematics shows that genes in the DNA are transcribed to RNA molecules that are further translated outside the nucleus into proteins. Those proteins can serve various purposes inside the cell or come back to the nucleus to regulate gene expression.

the main driving factor of this revolution. In the next section I will present two methods to capture gene expression.

1.3 CAPTURING GENE EXPRESSION IN THE LABORATORY

1.3.1 *In-situ hybridization assays*

In-situ hybridization (ISH) is an experimental technique where the practitioner is able to determine in which cells of the tissue under study a particular RNA is present. As opposed to Southern blotting [69], ISH assays not only allow to know whether a gene is expressed or not, but also where in the tissue it is expressed. First proposed in 1969 by Pardue [58] and John [40] independently, in-situ hybridization (ISH) used radioactive tritium labelled probes on a photographic emulsion to reveal parts of the studied tissues where particular RNA or DNA sequences were present. With the development of fluorescent labelling techniques [45, 62] allowing for faster, more sensitive and of course safer hybridization assays compared to radioactive probes [73], Fluorescent in-situ hybridization (FiSH) quickly became the standard technique to study gene expression in the spatial context of the biological tissue. Importantly, using multiple fluorescent probes of different colours allowed the simultaneous localization of several RNA frag-

ments in the tissues [55].

For small enough tissues to study, it is possible to stain hybridize the probes in the whole animal. This method called Wholemount in-Situ hybridization (WiSH) and a 3-Dimensional representation of the expression pattern of a gene can be deduced using confocal microscopy to study the patterns of gene expression slice by slice.

1.3.2 Building a image library of gene expression for *Platynereis*

During his PhD, Raju Tomer and other [cite thesis](#) members of the Detlev Arendt lab in EMBL, used wholemount in-situ hybridization to create an image library of gene expression in the brain of *P. dumerillii*. They were able to record gene expression in the full brain at 48hpf for 169 genes. In practice each individual larvae was dissected to isolate. Each brain was then stained with two different fluorescent probes corresponding to two messenger RNAs (mRNA). One of the gene is considered a reference, as it is always hybridized in all the assays (the main reference gene used was *Emx*) alongside an other gene of interest, see figure 5. Each brain was then visualized with laser confocal microscopy to reveal the gene expression patterns in the brain slice by slice, generating at the same time 3D coordinates for each slice.

As mentioned previously, the larval development of *P. dumerillii* is highly similar in every individual larvae. In the case of this study requiring a lot of different assays conducted each time a on different animal, the stereotypical development of *P. dumerillii* has proven essential. Indeed, having the same reference localized in all the assays has allowed Tomer to align all the other gene expression patterns onto this scaffold. The result is an image library of 169 gene expression patters in the full brain of *P. dumerillii* with a exploitable spatial reference that allows for a very precise mapping.

However useful and practical WiSH may be, such assays are limited in terms of the number of genes one is able to study. Indeed, each individual larvae only provides the expression of two genes, one being the reference. To answer this problem, crucial developments in sequencing technologies have brought a way to study the expression of the whole transcriptome landscape in a single assay, RNA sequencing.

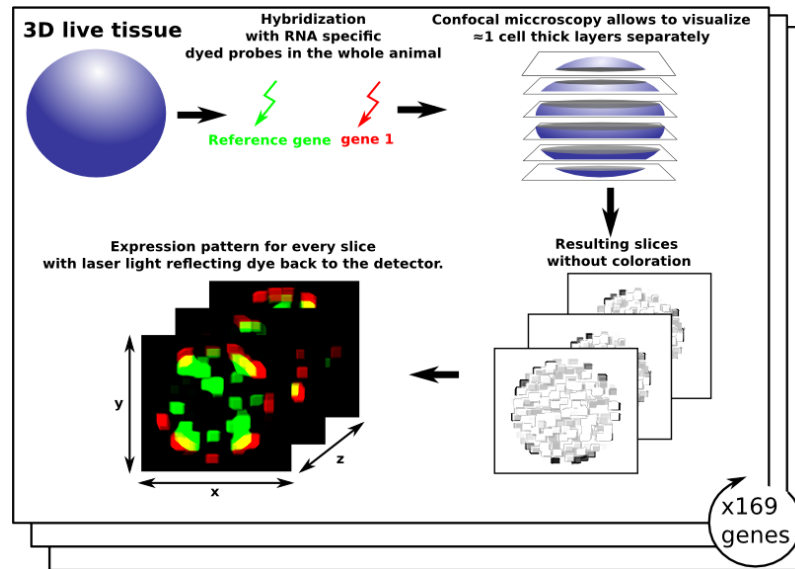


Figure 5.: Fluorescent in-situ hybridization assays to create a 169 genes catalogue of gene expression in the brain of *P. dumerillii*. From the live tissue cut into thin fixed layers, every slice is stained with a reference gene and a gene of interest that will reveal the areas of expression under fluorescent microscopy. The process repeated 169 times for key genes in *P. dumerillii* development has been generated by [78]

1.3.3 RNA sequencing

Whole Transcriptome Shotgun Sequencing (WTSS) also called RNA sequencing (RNA-seq) [53, 80] has developed alongside Next Generation Sequencing (NGS) techniques used to retrieve the genome (DNA). Instead, when preparing the starting material, only the RNAs are extracted. Protein coding mRNA can further be selected, they are separated from the rest by targeting the polyadenylated 3' tail, specific to protein coding transcripts. Most current technique use magnetic beads to achieve this separation [54, 53].

Once isolated from a population of cells, transcripts undergo fragmentation to obtain an average length of 200-300 nucleotides. The next step is the reverse transcription, which will create a complementary DNA (cDNA) library using viral reverse transcriptase enzymes. After amplification using quantitative Polymerase Chain Reaction (qPCR), the cDNA library is ready to be sequenced by NGS technology.

This will generate a large dataset of small reads, that needs to be mapped back onto the reference genome of the considered species, providing this genome is available. In that case, the resulting dataset will reflect a snapshot of the whole transcriptome in the studied cell population. However, in the case of *P. dumerillii*, this reference genome is not fully available yet, an alternative option is to map the reads back to a list of known gene sequences, for instance the 169 genes studied by [78] (PrimR genes). The resulting dataset will represent a quantitative image of the considered genes in the cell population at one point in time.

Because of technical limitations in this sequencing protocol, until very recently the starting quantity of RNA had to be relatively important (this issue is discussed further in 2.2). This is why most of the published RNA sequencing studies use a population of cell as a starting point. This however, means that the gene expression landscape obtained as an output will represent an averaged expression over all the cells used as an input.

Importantly, when comparing RNA-seq to the previously described in-situ hybridization technique, if the methodological burden to analyse the expression of a lot of genes at the same time is greatly reduced, the spatial localisation of the cells is lost during the protocol.

1.4 CONCLUSIONS

In this chapter I have presented *Platynereis dumerillii* and the advantageous traits it exhibits for developmental biologists especially in the field of neural development. I have discussed the fact that anatomical traits are not sufficient to fully comprehend the deep heterogeneous patterns of functionalities inside a complex organ such as the brain. In order push this understanding further an interest had to be taken in what defines the life of tissues and sub-tissues: gene expression. I have also described two methods that allow practitioners to capture gene expression from a biological tissue, and how an image library of gene expression for 169 gene was generated by [78] in the full brain of *P. dumerillii*.

So far, the scale of study has been the tissue, or the sub-tissue. However, as mentioned in the introduction ??, the heterogeneity of complex biological tissues does not stop at this scale of study. In fact, with a top-down approach looking at big tissues and then separating them in smaller sub-tissues until "true" functional tissues are defined is an extremely complicated problem. A solution to this problem would be to reverse the approach from a top-down to a bottom-up mindset. This means reducing the scale of study to the smallest biological unit available, the single cell, defining the heterogeneity of gene expression at the single cell level and going back up to the functional tissue level from there. Instead of a fragmentation problem, this becomes clustering problem, attaching single cells to a certain number of categories. In order to implement such an approach, single cell gene expression data is therefore needed.

FROM TISSUE TO SINGLE CELL TRANSCRIPTOMICS, A PARADIGM SHIFT

2.1 SPATIALLY REFERENCED SINGLE CELL-LIKE IN-SITU HYBRIDIZATION DATA

2.1.1 *Dividing images into "cells"*

Because in-situ hybridization keeps the studied tissue spatially untouched, achieving single cell gene expression resolution from one image obtained through confocal microscopy is a matter of microscope performance and cell size. For big enough cells, single cell resolution has been documented as far as 1989 [75] with some work specifically directed towards achieving this single cell resolution [63].

When considering the *P. dumerillii* brain dataset, with current microscope technology, achieving single cell level resolution on one particular image is feasible. However, the main limitation is the quantity of data involved, indeed, each brain is separated into 20 slices, for 169 genes. This technical bottleneck can be overcome with an automated way of analysing the fluorescence images. However this is not an easy task, as the computer program required needs to be able to *see* and divide the global picture into cells. Considering that all cells do not exhibit the same shape and size, constructing this *cell model* is a very complicated task.

It is for instance possible to highlight the limits of the cells and to automatically acquire those boundaries through computer vision methods. This process relies on targeting proteins in the membrane or in the extracellular matrix of the cells with specific fluorescent probes. Once the boundaries are acquired, defining every cell is a matter of finding enclosed spaces. To that end, numerous contour detection algorithms exist [47, 23, 9].

Unfortunately, a dataset with the cells limits highlighted does not yet exist for *P. dumerillii*'s brain, making a precise division of the

images into cells very difficult. Instead, Tomer used a basic approach to divided the images, the “cube” model [78].

2.1.2 A simple cell model, the “cube” data

Every slice of *P. dumerillii*’s brain being aligned onto the reference gene scaffold (see section 1.3) for all 169 genes, the “cube” model simply consists in dividing each image into squares approximately the size of an average cell. In the *P. dumerillii* dataset, the size chosen was $3\text{ }\mu\text{m}^2$. Importantly, this is actually smaller than the average cell size in *P. dumerillii*’s brain. Each slice of the brain being approximately $3\text{ }\mu\text{m}$ thick, the resulting dataset, spatially referenced in 3D, will contain $3\text{ }\mu\text{m}^3$ cubes, each of which is attached to the luminescence data for 169 genes.

Of course this cell model is far from perfect, it assumes that every cell in the brain is roughly the same size and cubical, which is clearly not the case. Consequently, the “cube” model will introduce errors in the dataset. The first type of error occurs within areas where the genes under study are highly expressed. In that case, the light emission might contaminate the cubes around that do not necessarily express the same gene see figure 6A. The second type of error is introduced by the choice of $3\text{ }\mu\text{m}^3$ cubes. As they are smaller than the average cell, some cubes will fall on areas that may be artificially empty. Indeed, transcription in the cells mainly happens in the nucleus. mRNA molecules then travel in the cytoplasm to be translated but they are not evenly distributed across the cell, in particular for some large cells, parts of the cytoplasm may record no expression in a cell that actually contains a lot of transcripts, see figure 6B.

Hence the data will tend to exhibit spatial discontinuity and inconsistency. With that fact in mind, any method that developed using this data, will have to take into account this spatial discontinuity and try as much as possible to smooth over those potential expression gaps.

However, even with this simple cell model the data generated by [78] is highly valuable. Indeed, not only does this dataset give a snapshot of gene expression for 169 genes in the full brain of *P. dumerillii*, it also attaches spatial information to each data point.

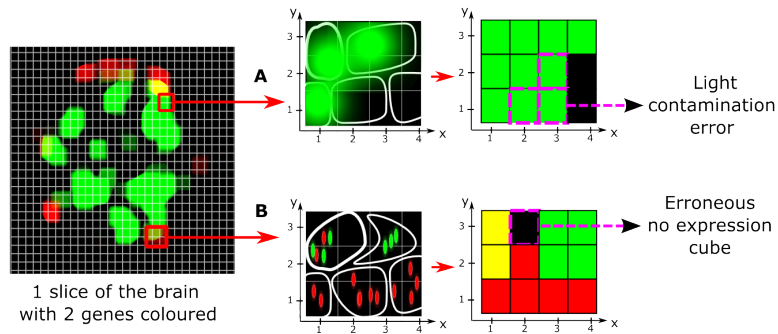


Figure 6.: Errors introduced by the "cube" cell model. Path A shows how regions with highly expressed genes can introduce errors through light contamination. Path B shows how some cubes may appear artificially void of expression because of the not even distribution of transcripts inside the cytoplasm especially for large cells.

2.2 SINGLE CELL RNA SEQUENCING, BUILDING A MAP OF THE FULL TRANSCRIPTOME

2.2.1 Sequencing single cell RNA contents

The scale shift from tissue to single cell is harder to achieve in the case of RNA-seq. As described in the previous chapter 1.3, an important factor for the success of RNA-seq assays is the starting input quantity of RNA to be sequenced. Taking mammalian cells as a reference, the quantity of RNA depends a lot on the cell type considered and can vary between 10 and 30 pg per cell, only 2% of which is mRNA [36, 38]. With such a small input quantity, distinguishing biological variation between different cells from the technical variation linked to cDNA amplification protocols has long been impossible to achieve.

However, with the creation of new protocols [64, 74], and the rise of microfluidics to facilitate the extraction and sequencing of single cells [57], the last couple of years have seen a dramatic increase in the number of single cell RNA-seq based studies [39, 49, 83, 70, 21]. However, challenges are yet to be overcome to be able to analyse further complex tissues from whole transcriptomes, one of which is the loss of spatial reference induced by the current protocols.

2.2.2 *Mapping back gene expression to a spatial reference*

Single cell RNA-seq achieves to capture a snapshot of the entire transcriptome of a given cell at a given point in time. However, to analyse cells from a complex tissue, current protocols require that the tissue is reduced to a suspension of single independent cells. This prevents from keeping track of any spatial information about the cells. Hence, when analysing single cell RNA-seq data from a complex tissue, back-mapping every cell to its original location becomes a crucial problem.

In order to achieve this back-mapping, a reference is needed. This reference should consist in an independent assay where gene expression in the considered tissue is defined for enough genes at a spatially small enough resolution to find for each sequenced cell, if not the exact original location of the cell, at least a restricted region of the tissue from which the sequenced cell originated with a high probability.

Fortunately, in-situ hybridization assays provide exactly this type of data and I will present in the last section of this chapter 2.5 a methodological proof-of-concept of this back-mapping in the brain of *P. dumerillii* with 72 sequenced single cells.

2.3 ABOUT THE QUANTITATIVE TRAIT OF SINGLE CELL EXPRESSION DATA

2.3.1 *Light contamination in in-situ hybridization data*

The light intensity value obtained from in-situ hybridization assays can be considered as quantitative [22]. Indeed, the light emitted by every cell in the considered tissue is correlated with the number of RNA fragments present in the cell as each fragment bound to a probe is an independent source of emission and the probes are hybridized in the cells in large excess. This means that if the targeted gene is highly expressed in a cell, there will be more sources of emission, thus making the overall light intensity captured on this area higher than in a cell expressing the gene at a low level.

As mentioned in a previous section 2.1, in-situ hybridization assays at the single cell level are prone to punctual errors due to the cell model. One of the culprit for those errors, as shown on figure

6B is the phenomenon of light contamination. When a large group of neighbouring cells express the same gene, because of the additivity of light intensity mentioned above, even though the cells express the gene at the same rate, cells surrounded by a lot of other cells expressing the same gene will have an abnormally high light intensity readings due to light contamination from the adjacent areas. As a result, when considering an hypothetical circular portion of tissue where a gene is monotonously expressed, the recorded light intensity will show a gradient with the maximum localized on the circle's centre.

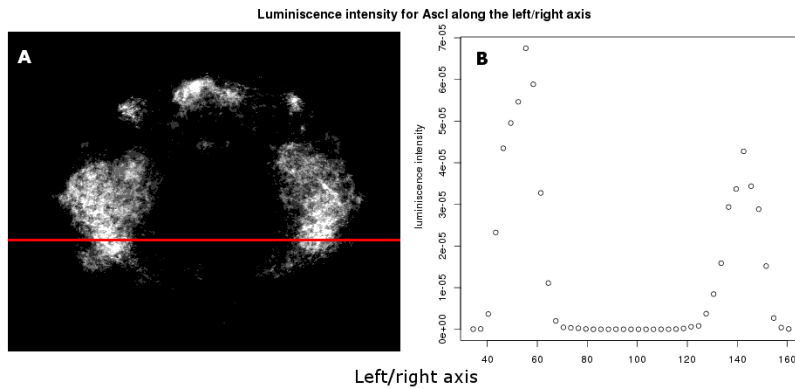


Figure 7.: **Light contamination in in-situ hybridization luminescence data seen with the example of the gene *Ascl*.** Panel A shows the raw fluorescent microscopy capture of the gene's expression for one layer in the brain of *Platynereis*. Panel B shows the light intensity measured along the red line in panel A. Because of the small scale of study, cells surrounded by other cells expressing a particular gene will have a higher intensity values because of nearby light contamination.

As shown on figure 7, the issue of light contamination seems to occur the used dataset when using the $3 \mu\text{m}^3$ "cube" model. In that context, and because of the single cell scale of this study, considering the in-situ hybridization data as quantitative may have introduced significant errors. In order to avoid this light contamination bias a solution is to transform the quantitative data set into a binary data set where for a given "cube", genes are simply expressed or not. The binarization method is described in the following section 2.4.

2.3.2 *Technical noise in single cell RNA-seq data*

Single cell RNA-seq is also prone to high levels of noise. This technical noise is caused by the minute amounts of starting RNA material. A study lead by Philip Brennecke, Simon Anders and Jong Kyoung Kim [14], proposes a statistical method to overcome this high noise level and distinguish between biological variation and technical variation in the gene expression levels.

To illustrate the dramatic increase in noise level, they used series of dilution assays, reducing step by step (5000 pg, 500 pg, 50 pg, 10 pg) the input quantity of RNA fragments extracted from *Arabidopsis thaliana* with two technical replicates each time using the Tang protocol [74]. The authors of the study let me analyse this data, and after normalizing by the size factor using the Bioconductor package DESeq [8] the scatter plots shown on figure 8 were generated.

It is clear from these dilution assays that the noise level is correlated with the input quantity. Even though highly expressed genes are consistently well quantified even with 10 pg input material, for most of the genes, with less than 50 pg input RNA it seems dangerous to assume the results of single cell RNA-seq as quantitative with the current technological capabilities.

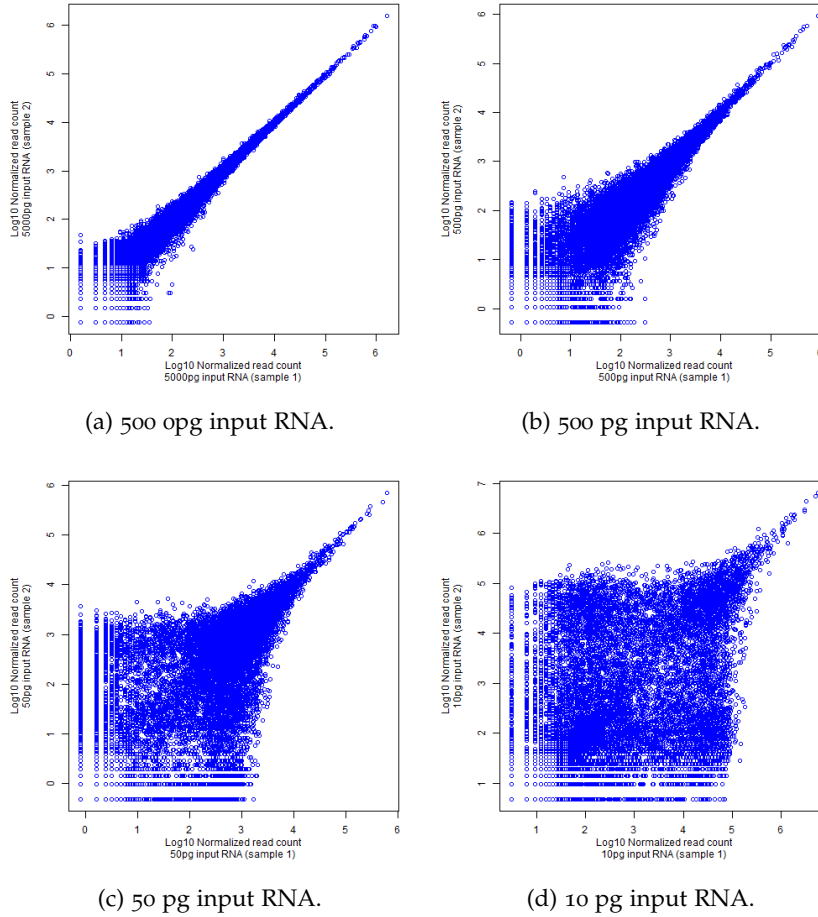
2.3.3 *Conclusions*

The paragraphs above have shown that neither in-situ hybridization nor RNA-seq data can be safely assumed as quantitative when the scale is lowered at the single cell level. To avoid nonsensical conclusions linked to the noise level in the rest of the study, a solution was to binarize single cell datasets. However, binarization is not a trivial problem as discussed in the following section.

2.4 **BINARIZING GENE EXPRESSION DATASETS**

2.4.1 *Binarizing in-situ hybridization datasets*

As shown in Figure 7 and discussed in the previous section 2.3, the various problems linked to light contamination can be avoided by transforming the “quantitative” fluorescence information into binary

Figure 8.: Dilution series of total *A. thaliana* RNA

data. In other words, if S is the set of all “cubes” in the brain, M the set of all the considered genes and $y_{i,m}$ the value retrieved from the in-situ hybridization data for “cube” $i \in S$ and gene $m \in M$, then $y_{i,m} = 1$ if gene m is expressed at site i , $y_{i,m} = 0$ otherwise. The binarization process itself is not trivial. Indeed, defining the light intensity threshold above which a gene is considered expressed is a complicated problem, especially for noisy data.

Looking at the density of intensities across all the “cubes” for each gene exhibited two very different scenarios: some densities were separated into two clear peaks, making the threshold easy to find while others exhibited a single peak making it hard to choose a clear cut value as shown in figure 9. After trying different thresholding methods based on those densities, I found in collaboration with Kaia Achim and Maria Tosches from the Detlev group in EMBL Heidelberg that the resulting binary expression was not satisfying for a large num-

ber of genes when compared to the in-situ hybridization raw images. Considering that this binarized dataset will be the cornerstone of the work presented in this thesis, it was very important to achieve a high confidence thresholding. Given the small number of genes studied (169), and the collaboration with a team of biologist working specifically on *Platynereis dumerillii*'s brain, a manual approach was decided in order to threshold each gene. Indeed, by going through the 169 genes one by one, it was possible to adjust the thresholds manually until the resulting binarized expression pattern corresponded perfectly to 1) the fluorescent stack images from in-situ hybridization data; 2) the biologically known expression patterns in the brain of *P. dumerillii* validated by the biologists.

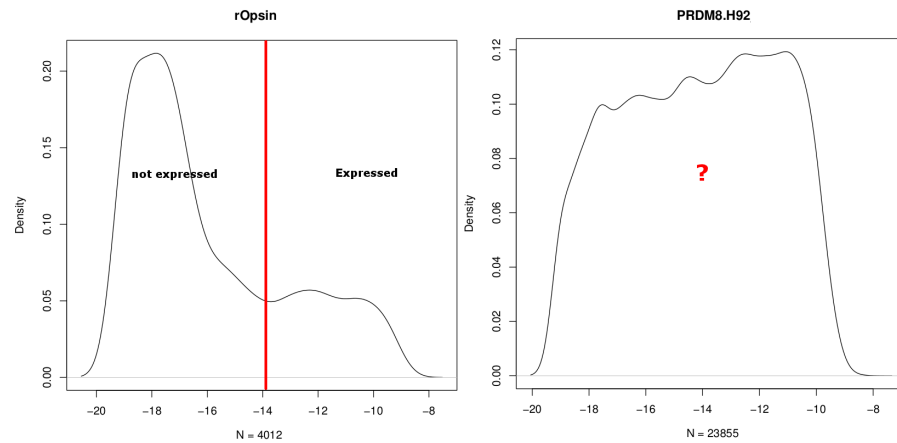


Figure 9.: **Densities of log luminescence values for two genes (rOpsin, PRDM8) over the 32,302 cells.** For *rOpsin*, the density exhibits two clear peaks making the choice of a binarizing threshold easy. By contrast, for *PRDM8* there is no such clear threshold, making an automated binarization method hard to implement.

This method resulted in a high confidence binarized dataset for 86 genes. Several reasons explain why 83 genes out of the starting 169 were removed from the dataset. For some of the genes no good threshold could be found, this was due to high noise level in the in-situ hybridization images. Other images suffered from experimental errors resulting in blurred and unexploitable expression patterns. Finally some images were polluted by a well known experimental artefact linked to confocal microscopy imaging.

Although the aforementioned method resulted in a high quality binary dataset, it has been possible only because the number of genes

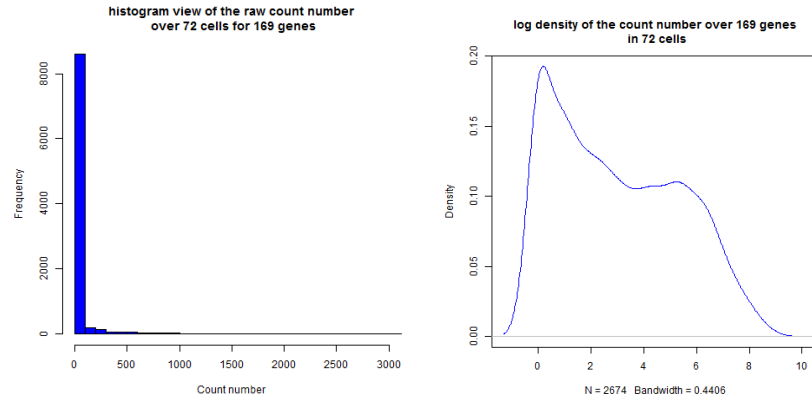
considered was small. This will not be the case when dealing with RNA-seq data.

2.4.2 *Binarizing whole transcriptomes*

When dealing with whole transcriptomes, manually finding thresholds to binarize gene expression data is no longer a valid option due to the high number of genes considered. An automated method is thus required. I will discuss possible ways to binarize single cell RNA-seq data, presenting some results from a small number (72) of sequenced cells in the brain of *P. dumerillii*, containing the count number for 169 genes (see next section 2.5 of a detailed presentation of this data).

A naive approach would be to simply consider that as long as one RNA fragment mapped to a particular gene has been found in a cell, the gene is considered as expressed. Although such a method would be justifiable in the case of a perfect dataset, with no noise or errors, as discussed above 2.3 in the case of single cell RNA-seq the noise level generated by the currently available sequencing technologies would prove too high to rely simply on this method. However, as a first approach on *P. dumerillii*'s dataset, we can see on figure 10a that as predicted the value 0 represents a very dominant peak. The problem remains that for very small count numbers it seems dangerous to set the gene as expressed.

Another option would be to find a global threshold over the complete dataset. The threshold $T > 0$ would represent the count number of RNA fragments for a particular gene and a particular cell needed to consider the gene as expressed. T could be inferred from the count density over all the genes and all the cells. The expected result would be a 2 peaks density curve, one peak would correspond to the non expressed count values, the second to expressed genes. The binary threshold would then be set between the first and second peak. Although more precise than the previous method, binarizing in such a manner may lead to numerous errors. Indeed, the underlying assumption behind this method is that all genes behave in a similar way. As figure 10b shows, if a 2 peak behaviour is indeed present, the cut is not extremely clear and an important portion of count numbers actually fall in between the two peaks. This is due to the fact that all



(a) Histogram showing the frequencies of count values over 72 sequenced cells, with the fragments mapped to 169 genes. (b) Density plot for the count values over 0 in the single cell sequencing dataset

Figure 10.: Thresholding RNA sequencing data for *P. dumerillii*

expressed genes are not expressed in the same way, some lowly some highly, which has a tendency to flatten the density curve making this thresholding method, if better, still not 100% reliable.

The more suitable approach to this thresholding problem, would be to compute one threshold per gene based on the density curve for every gene across all cells. However, with 72 cells into consideration, considering the sparse nature of the count data, no significant results can be extracted with this method on this particular dataset. However, I believe that one threshold per gene would prove a big improvement over the previously mentioned thresholding methods providing sufficient number of data points per gene.

2.5 PRELIMINARY RESULTS ON SINGLE CELL RNA-SEQ SPATIAL BACK-MAPPING

2.5.1 Single cell RNA-seq in *Platynereis*' brain

Collaborations with the Kaia Achim in the Detlev lab in EMBL have provided us with a unique RNA-seq dataset of 72 single cells from *P. dumerillii*'s 48hpf developing brain.

Experimentally, the work consisted in setting up *P. dumerillii* batches, picking up 50-100 individuals at 48hpf. These were washed in Ca, Mg - free sea water and incubated in a mixture of pronase (breaks extracel-

lular matrix) and thioglycolate (helps to break the chorion). After this treatment, the trunks and epispheres (brains) were separated. 40-60 epispheres were then picked out, transferred to Phosphate buffered saline (PBS) and then incubated for 1 minute in PBS containing collagenase to break more extracellular matrix. After two PBS washing, the cells were dissociated by pipetting up and down then washed again 1 ml of PBS and concentrated by centrifuging (1 min, 1000 rpm). Cells were re-suspended in 20 microliters of PBS, of which 5 microliters could be loaded on the capture chip.

Fluidigm's C1 Single-Cell Auto Prep System instrument with the Fluidigm Single-Cell Auto Prep IFC chip optimized for 10-17 micron cells were used. The reverse transcription was performed using Clontech SMARTer Ultra Low Input RNA Kit and for on-chip PCR the Clontech ADVANTAGE-2 PCR kit. Sequencing libraries were prepared using Nextera DNA Sample Preparation kit from Illumina.

With two chips and a capture rate of 65%, 72 libraries were sequenced including 11 cells from first chip, 35 live single cells, 17 dead single cells, 3 tubes from 2 cells, one with 4 cells, and 3 unsure ones from the second chip resulting in 72 raw reads files.

Of course those results do not include the spatial localization of the cells as the protocol requires the separation of the coherent tissue into a cell suspension. As a crucial point in any downstream analysis, being able to map back the single cells to their original location in the brain is required. To that end, I took advantage of the spatially localized in-situ hybridization described in the previous section.

2.5.2 *Mapping back RNA-seq data back to PrimR in-situ hybridization assays*

Firstly, the RNA-seq raw reads were mapped to the 86 reference genes composing the in-situ hybridization data using Bowtie.[cite nuno pipeline](#). The resulting dataset is the count number for each of the 86 genes in the 71 cells sequenced. In order to map back to the in-situ hybridization data, the chosen approach consisted in extracting the genes that were the most specifically expressed for each sequenced cell, then compare this specific fingerprint to the in-situ 3D data in

order to isolate the regions of the brain where those specific genes are co-expressed.

Given the set of 86 considered genes M , and the set of 72 cells C , with the read counts matrix D of size $M \times C$, the expression specificity ratio $r_{m,c}$ can be computed for each cell and each gene as :

$$r_{m,c} = \frac{D_{m,c}}{\frac{1}{\|C\|} \sum_{a \in C} D_{m,a}}$$

where $\|C\| = 72$ is the number of cells considered. By looking for each cell, at the genes with the highest specificity scores, a list specifically expressed genes in each cell can be determined in comparison to the other cells sequenced. On the one hand, this mapping method has the inconvenient of using the average expression level across all considered cells to compute the ratio r . This means that the mapping quality of each cell will depend on the overall sequencing quality. Furthermore, this method performance relies on the assumption that the data is in fact a collection of cells from different cell types, given the experimental protocol described above, this seems to be an acceptable hypothesis. On the other hand, this mapping method has the advantage of not being sensible to technical noise in the RNA-seq protocol, providing the technical noise between cells remains at a constant level. This explains the use of the read counts in a quantitative way and not a binarized dataset.

The goals of this study were to validate the protocol used in order to obtain single cell RNA-seq results in *P. dumerillii*'s brain and to establish a methodological proof-of-concept on spatially mapping RNA-seq results onto in-situ hybridization data. I will present here a few examples of sequenced cells, their most specifically expressed genes and their resulting potential original location in the brain as well as the probable cell type they belong to.

In table 1 are shown the most specific genes for four cells sequenced. With that list of genes that are the most representative ones for the four cells considered, visualizing the areas where those genes are co-expressed in the brain of *P. dumerillii* according to the in-situ hybridization data, seems to give a good indication about the probable the original cell location. A snapshot of this visualization is shown on figure 11. In every case, simply looking at the three most representative genes seems to allow a clear localization of the sequenced cells. Of course this mapping is not at the single cell level, but having an

X2C911L	X2C521L	X2C61L	X2C241S
Emx	Wnt8	VACht	Mitf
CALM.R29	HEN1-Y61	ChaT	Otx
Dach	Gsx	LYamide	Tolloid-Y68
Mushroom body	Developing lateral brain	Differentiated neural tissue	Adult eye

Table 1.: Top 3 most specific genes for 4 sequenced cells and the potential tissue they belong to. The resulting localization of those four cells inferred from the in-situ hybridization data are shown on figure 11.

idea of the tissue every cell originated from is already a nice proof-of-concept.

From the most specific genes to each cell and their potential localization can be used to hypothesize from previous biological studies the cell type of each sequenced cell. As shown in table 1, for cell "X2C911L" the most specifically expressed gene "Emx" has been used as a reference gene to localize the mushroom bodies, an hypothesis which is compatible with the co-expression of "CALM.R29" and "Dach" [78]. Cell "X2C521L" expresses Wnt8 very specifically, a gene shown to be linked to lateral brain development. Cell "X2C61L" can be easily classified as a developing neuron. Indeed both VACht (Vesicular acetylcholine transporter) and ChaT (Choline acetyltransferase) are genes coding for enzymes interacting with the neurotransmitter acetylcholine. Finally cell "X2C241S" displays the specific expression of the gene "Mitf", one of *P. dumerillii* most studied gene and expressed solely in the developing adult eyes [43, 30].

2.6 CONCLUSIONS

In this chapter I presented how the scale of gene expression studies has shifted from the tissue level to the single cell level. For two experimental protocols the generation of such data was described and I explained why at the time of writing, it is still not safe to assume the single cell datasets as quantitative. To avoid that problem, turning those datasets into binary gene expression is an attractive solution. However the binarization process is not trivial and I have presented

ways to obtain a high confidence dataset.

One big advantage of in-situ hybridization assays is the fact that the spatial information stays attached to each gene expression fingerprint. Using this information, it was possible to replace single cell RNA-seq data in a spatial context, using the most specifically expressed genes in every cell. To my best knowledge an a posteriori localization of single cell RNA-seq data had never been presented before.

As mentioned previously in chapter 1, if the ability to study the heterogeneity of cell populations at the single cell level offers incredible possibilities for the future of developmental biology and potentially of cancer research, the development of new statistical methods adapted to this single cell scale, allowing conclusions to be drawn at the tissue level is crucial.

The work presented thereafter was done to answer simple but important questions: can known functional tissues of a complex organ like the brain be defined and localized from single gene expression data? Can unknown regions in such a complex tissue be detected and finally is it possible to hypothesize the functional role of those unknown regions based on the single cell expression data?

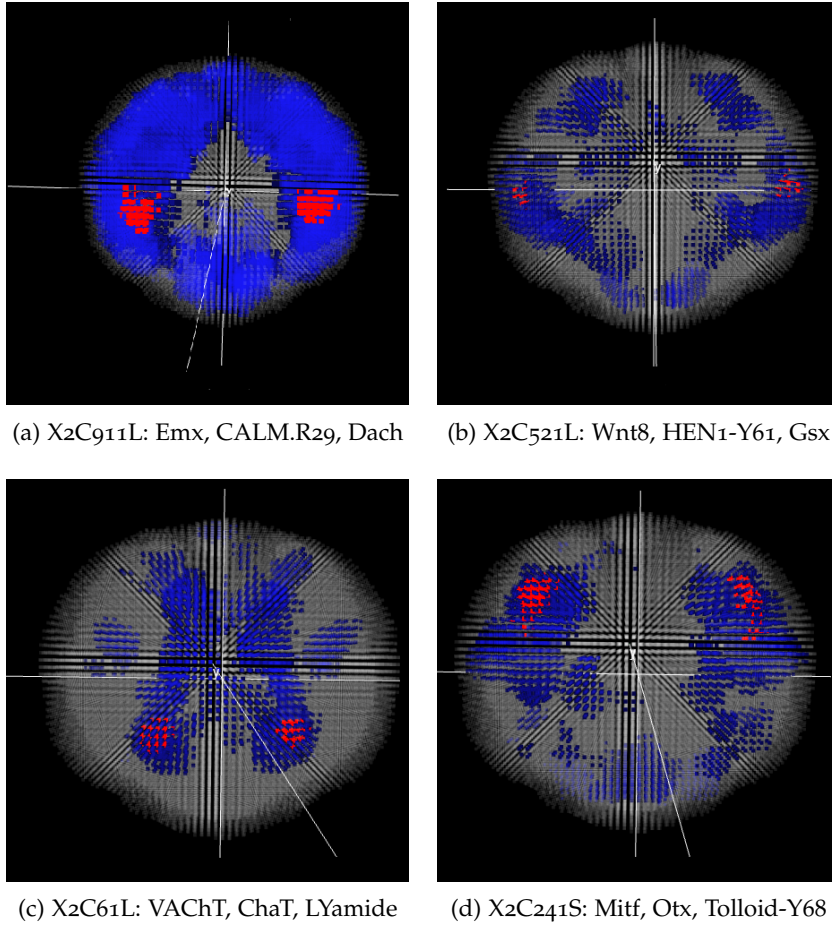


Figure 11.: Regions defined by the expression overlap of the top 3 scoring genes in [78] binarized in-situ hybridization data. The red colour shows the co-expression of the three considered genes, the blue areas are those where one or more of the three considered genes are expressed but not all, in grey are the areas where none of the considered genes are expressed. The 4 figures are from a apical view with the dorsal side on top.

CLUSTERING AND VISUALIZING TISSUES FROM 3D SINGLE CELL EXPRESSION DATA

3.1 ELEMENTS OF CLUSTERING FOR BIOLOGICAL TISSUES

3.1.1 Motivations

Following the conclusions of chapter 2 in which was discussed the shift from the tissue to the single cell scale of study in developmental biological assays as well as the main challenges to analyse and interpret such data, the first question that seems natural to answer is the following: given single cell gene expression data only, is it possible to classify cells that are the most alike together and define the organization of complex biological tissues like the brain of *Platynereis dumerillii*?

This is fundamentally a clustering or a classification problem. Indeed, small units need to be put in a determined number of classes or clusters because they are alike in one way or another. As anatomical and functional information about some tissues in the brain of *P. dumerillii* is already available (see chapter 1), the obvious validation for any clustering/classification method developed is be to check that the single cell level information leads to the definition of those known tissue. Once this has been validated, it seems important to determine whether the single cell expression data *adds* to the known biology by redefining (subdividing for example) known tissues or finding new ones. Finding the localization of putative functional already is a big challenge but the question can be pushed further. Indeed, as mentioned in chapter 1, because gene expression is the key process in a cell's life, a method based on gene expression to classify single cells will be informative about the genes that drive the resulting clusters specifically and may thus help us understand what the putative functions of the unstudied tissues are.

3.1.2 General considerations about clustering

I mentioned before that methods developed to answer those questions could either be clustering or classification methods. In the field of machine learning, those two notions are fundamentally divergent. Indeed, when clustering describes a method that assigns points to an unknown number (*a priori*) of sets, in an undirected way, classification takes advantage of an already known classification, in a directed manner, new elements in a determined number of clusters.

In my case, the number of tissues in the brain of *P. dumerillii* is unknown and there is no previous classification or a strong enough biological knowledge of each and every cell to opt for a directed classification. Therefore the methods presented hereafter will be clustering methods with respect to the machine learning definition of the word.

As a general consideration about clustering, it is interesting to note that unless working on simple enough datasets, there is no perfect method. This is especially true when dealing with biological data, where the inner complexity and the noise level (see 2.3) tend to be extremely high.

To illustrate that notion, for the single cell expression data in *P. dumerillii*'s brain, the question of the finding the "true" number of tissues is extremely complicated. Without any prior knowledge, the statistical methods to determine the number of clusters presented in this thesis will yield indications about what the optimal number of "tissues" is given the data and the model, which does not necessarily mean that this number is biologically *true*.

For a complex dataset such as brain tissue, a crucial matter in the upstream and the downstream analysis is to be able to visualize the data. This is especially true when the considered tissue is in 3 dimensions. For example, an important part of the upstream analysis to clustering was to visualize the gene expression patterns in the brain to find the right binarization thresholds. In the downstream analysis, after developing a clustering method, seeing the resulting clusters and their localizations in the brain is crucial. Therefore, I decided to develop a tool for 3D datasets visualization taking advantage of the latest developments in browser based technologies to create the software "bioWeb3D". This work has been published in [61].

3.2 VISUALIZING CLUSTERING RESULTS IN 3D WITH BIOWEB3D

3.2.1 Background

Visualisation is a key challenge in the analysis of large biological datasets, especially when analysing organized structures with distinct sub-clusters [66]. This is particularly important when analysing 3-Dimensional (3D) datasets. When a biological process or feature has been described spatially by a set of 3D referenced points, either via laboratory work (confocal microscopy for example) or generated within a simulation, with some data attached to each point in space, the first step in interpreting the data is to visualise it. Once the data are visualised and their quality assessed, downstream analysis can proceed. For example, a typical second step is to cluster the observations into different classes based upon the information associated with each point; those results will also need visualisation.

While various 3D visualisation tools have been developed, they have typically been made available via a locally installed piece of software such as BioLayout Express^{3D} [27], Arena3D [59], 3D Genome Tuner [79], Amira 3D [71], V3D [60], the Allen Brain Atlas [46] or Cytoscape [67]. These tools are very complete and usually complex to operate for non-expert users. Moreover, they require installation on every machine they are used on, which makes sharing inconvenient. To address this issue, some 3D visualisation tools have been built online and are accessible through the browser directly, such as AstexViewer [33], which is utilised by the Protein Databank Europe via a Java Applet. More recently, visualisation tools developed using HTML5/WebGL capabilities have been described, although they have focused on very specific applications, such as analysing radiology data [44].

Importantly, before bioWeb3D [61], no tool has allowed biologists to view their own 3D data directly online in an easy, fast, interactive and secure way. Using WebGL and the JavaScript 3D library Three.js, bioWeb3D aims to be a simple, generic, tool for tackling this problem.

3.2.2 Implementation

bioWeb3D allows the user to represent any 3D dataset on their browser by defining only two files. The two files can either be formatted as JSON or XML files, two widely used structured formats on the web [81] [7], or directly as Comma Separated Values files (CSV).

The first file used by the application, referred to as the *dataset file*, contains the spatial coordinates of every point in the dataset. The second type of file used, the *information layer* file, describes one or several information layers that are associated with every point defined in the first file. For example, if each point defines the location of a cell within a tissue, the second file could describe whether a particular gene is expressed in each cell. That way the tissue expression profile can be represented in the spatial context of the tissue.

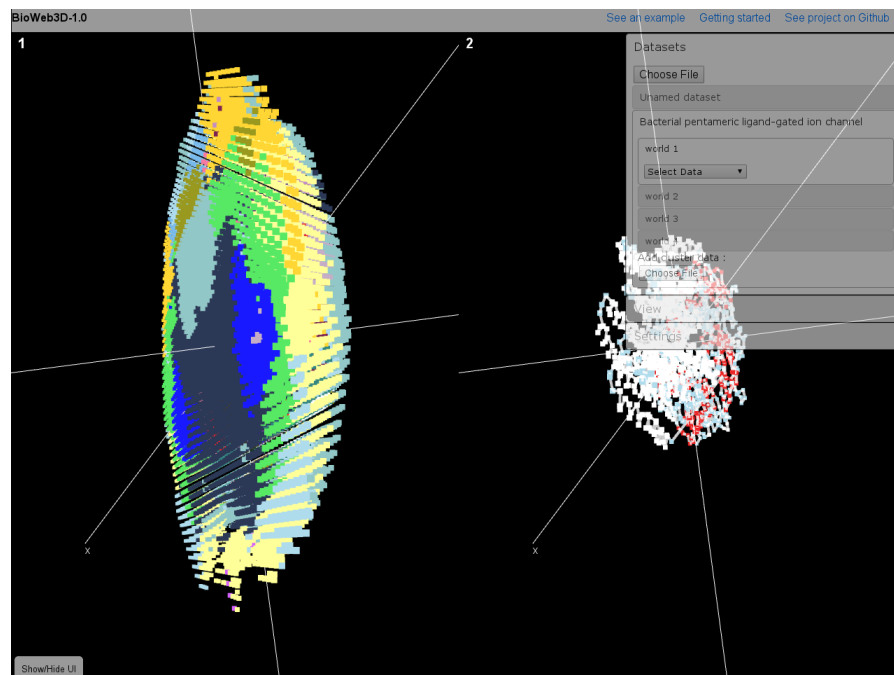


Figure 12.: bioWeb3D allows several datasets to be visualized at the same time in up to 4 different “worlds”

Datasets can be viewed and compared in up to four “worlds” (each world refers to a separate visualisation sub-window) at the same time (see figure 12). Although browser based, the application, fully written in Javascript, does not need to send any data to the host server.

Instead, the modern internet browser's local file system reading capabilities are used through the HTML 5 FileReader functionality. This allows the application to handle, in a very short period of time, large datasets while ensuring that the privacy of the data is maintained.

Although the focus is on making bioWeb3D simple and easy to use, some options are available to customise how datasets are represented. The application can be used to visualise sequential information, such as 3D protein structures, in which case a solid line can be drawn between the points (figure 12 (right)). In other situations, such as when a population of cells is considered, the points are viewed as individual particles. The information layers are visualised by colouring the 3D points according to the class that each point belongs to.

TECHNOLOGICAL OVERVIEW bioWeb3D is fully written in HTML/JavaScript. It relies heavily upon a relatively recent 3D javascript library called Three.js [5]. This library is used as the main interface between WebGL (cross-platform, royalty-free web standard for a low-level 3D graphics API) [6] and javascript. More specifically, bioWeb3D allows the generation and manipulation of simple Three.js objects. Indeed the primary challenge associated with the creation of bioWeb3D has been to design interactions between the 3D visualisation and the user interface in the most efficient way.

The 3D data are rendered using simple 2D quadrilaterals positioned in the 3D space according to their coordinates. This simple technique has been selected to keep bioWeb3D as light-weight as possible whilst ensuring good quality visualisation performance and fluidity.

JSON is the recommended format to input files into bioWeb3D because of its rigorous structure and its fast object generation, which is directly built into all of the primary internet browsers' interpreter. Compared to other data-interchange languages, such as XML, JSON is also easily human readable thanks to a light-weight syntax.

However, some applications might output data only in an XML format and not JSON, as the latter is generally more web oriented. For this reason bioWeb3D can also accept XML as an input format.

Furthermore, much data generated in the biological sciences is stored within CSV files. Converting CSV documents to the JSON or XML format is not always trivial. In order to facilitate this process, the application is also able to directly render simple CSV files that follow a certain format as an input. The file formats to input data into bioWeb3D are described with examples in Appendix A.

3.2.3 Results and Discussion

BASIC USAGE The goal of bioWeb3D is to allow scientists unfamiliar with visualisation software to explore 3D data very quickly without having to install any software. To illustrate its utility I used bioWeb3D to visualize some preliminary results within the single cell gene expression data of *P. dumerillii*'s (data described in chapter 2. In the context of bioWeb3D, the locations of the "cubes" are used to generate the "Dataset" file and information about the sets of cells that define clusters with similar gene expression profiles are used to generate the "Information Layer" file. In Figure 13 the results are illustrated — each point represents a pseudo-cell and its colour indicates the class (or cluster) to which it belongs to, here only two clusters are highlighted.

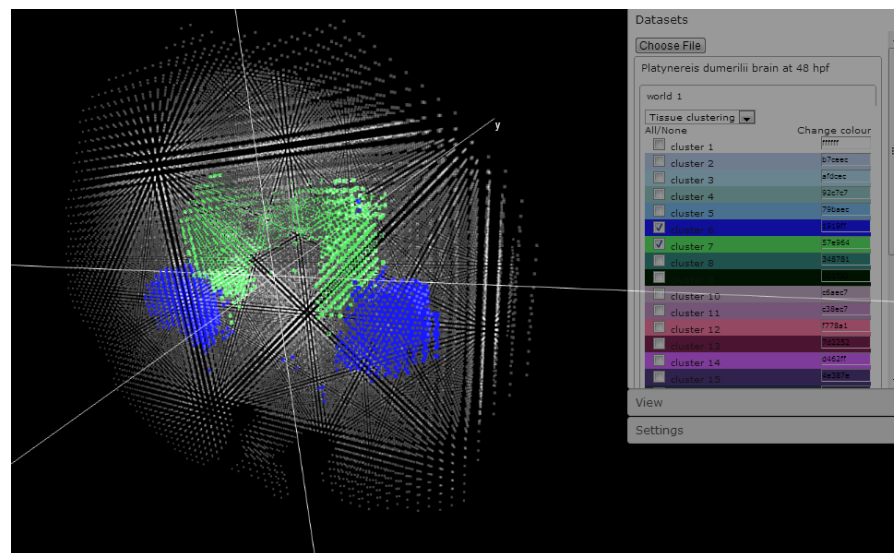


Figure 13.: The 3D location of cells within the brain of the marine annelid *Platynereis dumerillii* is shown. Two classes are displayed (in green and blue) along with the shadow of the remaining cells. The User interface is visible on the right of the screen and can be hidden. Data for this figure was taken from [78], see 1 for a presentation of *P. dumerillii* and chapter 2 for detailed presentation the data.

bioWeb3D can be used to visualise datasets derived from a wide variety of biological assays. Examples are shown on the Github wiki [3], where a 3D representation of a Principal Component Analysis (PCA) carried out with R and the 3D structure of a protein extracted from the PDBe database are displayed.

More generally, the user can interact with the visualisation via an interface on the right of the screen, which contains three panels as shown on figure 14. In the “dataset” panel, the user can choose the *datasets* and *information layer* files that should be represented in each world. This panel also allows the user to show/hide specific classes of the selected information layers. Each dataset file entered will create a new sub-panel where the user can input *information layer* files for that world. Selecting an *information layer* in the drop-down list will display the data in the current world and generate a list of classes that the user can modify regarding their visibility and colour. The “View” panel enables the user to choose which of the worlds are shown on the screen, ranging from 1 to 4. Finally, the “Settings” panel provides the user with a number of options that affect all worlds and all datasets, such as modifying the axes scales, modifying the transparency and size of raw data points and information layer coloured points. The user can also choose to enable centering of the data around 0 or leave the coordinates as inputted.



Figure 14.: The three control panels to control visualization in bioWeb3D.

A: the datasets panel, where new datasets and new information layer files can be inputted. From the dataset panel, information layers can be selected and unselected, and datasets attached or detached from the “worlds”. B: the view panel where the user can choose the worlds to display. C: the settings panel where the user can customize visualization.

BIOWEB3D AND LOCAL SOFTWARE Many 3D visualisation software tools, most of which require local installation, exist and provide similar functionalities with standard 3D format input such as Wavefront .OBJ. Some are extremely generic and powerful like Blender or

Amira 3D. However, these tools are not typically oriented towards a scientific audience. Moreover, those that are more focused on science are often targeted towards a very specific application, especially in the medical sciences [79]. In this context, I believe that bioWeb3D can be useful as it is completely generic and browser based. It should also be noted that recent browser improvements regarding GPU acceleration through the WebGL paradigm allow bioWeb3D to visualise several hundred thousand points. Additionally, local software is usually platform specific, which is not the case for browser based applications.

BIOWEB3D AND JAVA APPLETS As mentioned previously, browser based 3D visualisation tools currently exist mainly in the form of Java Applets. This technology has attracted much criticism in 2012 regarding security flaws, leading the "United States Computer Emergency Readiness Team" to advise that all Java Applets should be disabled due to current and future Java vulnerabilities [4]. The development of WebGL technology is viewed by many as a candidate for replacing Applets.

CURRENT LIMITATIONS The main current limitation of a WebGL based application is the machine and browser compatibility. Only computers with fairly recent graphic cards will be able to run a 3D environment. It should also be noted that Microsoft has notified the developer community that Internet Explorer is not scheduled to support WebGL in the near future. However, importantly, Chrome, Firefox, Safari and Opera all now support WebGL applications. Moreover, WebGL is also supported on mobile platforms such as iOS or Android. [2]

OPEN SOURCE AND COLLABORATIVE DEVELOPMENT As a fully open source software, the source code for bioWeb3D is available on Github [3], a web platform that allows interested parties to collaborate on the development of the project. In the wiki page "Contribute to bioWeb3D", directions to alter or add capabilities to bioWeb3D are provided for users who wish to get involved.

3.2.4 *Conclusions*

bioWeb3D is designed to be a simple and quick way to view 3D data with a specific focus on biological applications. Being browser-

based, the software can be easily used from any computer without the need to install a piece of software. Importantly, bioWeb3D has been designed to offer a very straightforward and easy-to-use working environment. Despite current limitations in terms of compatibility or rendering performance for large numbers of points, I believe that bioWeb3D will enable non-experts in 3D data representation to quickly visualise their data and the information attached to it in many biological contexts, thus facilitating downstream analyses.

3.2.5 *Availability and requirements*

The full source code is available on the Github page of the project [3]. A live version of the software is online [1]. You will require a graphical card and a browser with WebGL capabilities to run bioWeb3D.

3.3 NON SPATIAL CLUSTERING METHODS

Being able to visualize clustering results will be key in analysing any method's output from a biological perspective. Of course the next step is to actually develop a clustering method that would be able to cluster binarized single cell gene expression data. I will briefly describe in the next paragraphs existing methods that are able to cluster single cells together based solely on gene expression patterns without considering the spatial structure of the data.

3.3.1 *Hierarchical clustering*

The first method I took an interest in was hierarchical clustering (hClust) [41]. Indeed, in the field of molecular biology and biology in general, this clustering method is extremely popular mainly because it is relatively straight forward to use, and because the obtained dendrogram helps the downstream analysis of the data.

hClust relies on the computation of a distance matrix, considering the matrix D of the in-situ hybridization data, with 86 columns corresponding to the 86 genes considered and 32,203 row corresponding to every cell in the dataset. The computation of the distance matrix was performed using the *dist* function in R with the *euclidean* metrics (or *Manhattan* which is equivalent for a binary dataset) in order to

compute the $32,203 \times 32,203$ matrix of distances between rows.

It is interesting to note that this step, in addition to being computationally expensive, creates a very large object in memory making it a limiting factor for very large datasets. Based on this distance matrix, the hierarchical clustering can take place using the “hclust” function in R. With the “complete” option turned on, the resulting object will be a dendrogram representing the hierarchical classification of all 32,203 cells as shown on figure 15.

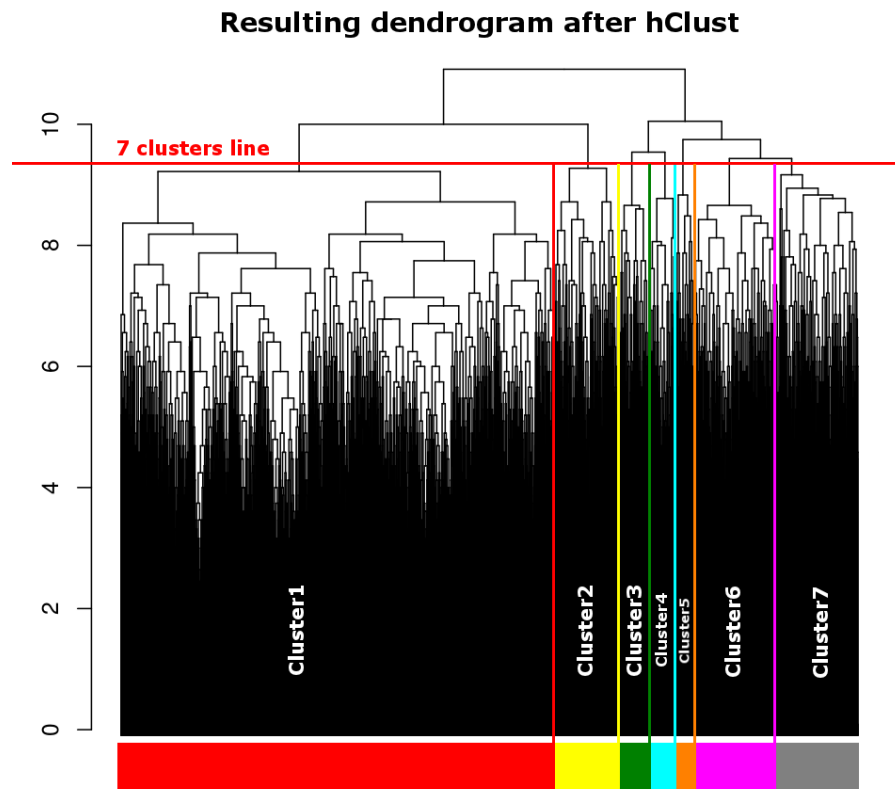


Figure 15.: Dendrogram outputted by the hClust clustering method. This is the full dendrogram, to get the resulting clustering partition, the tree must be cut. The red line shows the cut needed to get 7 clusters out of the dendrogram. Importantly, a dendrogram does not provide any rational about the best number of clusters.

Importantly, once this dendrogram is outputted, the hClust method gives no indication on where to “cut” the dendrogram to output the actual clustering results. This highlights a very important issue in all clustering methods that is identifying the optimal number of clusters needed for a certain dataset when there is no prior information. For

the rest of this thesis the number this crucial parameter will be referred to as K .

After choosing K , the dendrogram is cut so as to allocate each cell to one of the K clusters. Interestingly, the resulting partition obtained with this method does not give any direct information regarding the genes defining each requiring more steps to be able to generate biological hypothesis about the cells in each cluster.

3.3.2 *Other clustering methods adapted to gene expression data*

There are several other clustering methods that could be applied in order to create a partition of gene expression data. They include but are not limited to K-means clustering [48] and independent mixture models [19]. Those methods are well suited to cluster gene expression data. However, they all fail to take into account the spatial information linked to the gene expression dataset.

3.4 DISCUSSION

3.4.1 *Spatial clustering techniques (hierarchical, model based)*

Using a clustering method that would in addition to the gene expression pattern of each cell, take into account its spatial localization, that is the context of each cell regarding the other cells around it, could theoretically improve the clustering performances for several reasons.

As mentioned in chapter 1, the way development in complex organisms operates, from asymmetrical cell divisions leads to a highly structured spatial organization of complex biological tissue. In those circumstances and without any other prior knowledge, not taking into account the spatial localization of each cell into the clustering scheme would seem to be an unexploited potentially important information.

Furthermore, as discussed in chapter 2, single cell gene expression datasets whether they are generated from in-situ hybridization (see details on figure 6) or single cell RNA-seq (figure 8), are prone to errors and incoherency. A clustering method that would be able to compensate “erroneous” data points by taking into account the spa-

tial context of each cell, could potentially dramatically decrease the effect of noise level into the clustering results. Additionally, from a downstream analysis perspective, as far as general hypothesis about clusters are concerned, “smoother”/less scattered clusters are easier to interpret from a direct visualization of the results.

3.4.2 *Method chosen*

In order to utilise both the spatial and the gene expression information, it was decided to extend a graph theoretical approach developed for image segmentation to reconstruct noisy or blurred images [18], a method that finds its roots in the field of statistical mechanics as the Ising model [37] and its generalization, the Potts model [82]. The core concept of this method is to use an Expectation-Maximization (EM) procedure to estimate the parameters of a Markov Random Field based model using mean-field approximations to estimate intractable values as described in [15].

This approach exhibits several important advantages as will be described in details in the next sections. Indeed, in addition to providing a way to take into account the spatial information in the clustering results, it also offers some nice features in terms of downstream analysis through the analysis of the optimal parameters upon convergence.

In the next three sections, I will first describe the theoretical framework and the equations used in this model followed by a assessment of the method’s performances compared to other non spatial clustering method on simulated data, finally I will show and analyse the clustering results obtained through the developed method on the single cell in-situ gene expression data in *P. dumerillii*’s brain described in 2.

HIDDEN MARKOV RANDOM FIELDS FOR BIOLOGICAL DATA CLUSTERING

This chapter gives a theoretical overview of a Hidden Markov Random Field based approach aimed to be applied to cluster single cell in-situ hybridization gene expression data “cubes” as described in chapter 2 into K clusters ($K \in [2, \infty]$). Subsequently, we will describe our approach for estimating K .

4.1 MARKOV RANDOM FIELDS

4.1.1 Neighbourhood systems

Let S be a finite set of sites, each of which represents one “cube” of data. Given the 3D coordinates of each site, the first challenge to be able to use the spatial characteristics of the data in the clustering scheme will be to express the data and their spatial relationship in mathematically formal manner. To that end, starting from the spatial coordinates in 3D of each “cube”, instead of a list of isolated measurements, it is possible to build a connecting graph representing the same data and the spatial dependence between the “cubes”. In the case of this study, each node of the graph will represent a “cube” in the single cell expression data. Nodes that are linked together by an edge will be spatially dependent upon each other.

With prior biological data, one can manually create the spatial dependency graph by linking nodes together that are known to be functionality similar. In the case of this study however, no such prior knowledge being available, it is necessary to define the spatial dependences in a different way.

The central hypothesis while developing this method is to assume that “cubes” that are close to one another are more likely to belong to the same cell type (i.e cluster), the spatial dependencies connecting graph will then consist in a *neighbouring graph* where “cubes” close to each other will be joined by edges.

As shown on figure 16, it was decided to use a first order neighbouring graph called G , i.e the 6 closest sites, to represent S . using a first order neighbourhood system, i.e the 6 closest sites.

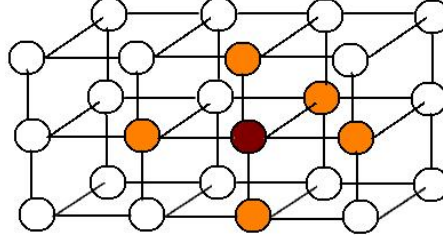


Figure 16.: **First order neighbouring system.** In the chosen first order neighbouring system, each node in the graph (representing a “cube” \approx cell in the data) is linked to a maximum of 6 other cubes. The Markov property on the graph implies that the state of any node (the red one for example) can be fully determined by knowing the state of its neighbours (the orange ones).

In G , a *clique* C_1 is a subset of nodes that are all interconnected, i.e it is possible to go from any the nodes in C_1 to any other node in C_1 by simply following one single edge. Let C be the set of cliques of G . Because we chose a first order neighbouring system, C is therefore the set of all the couple of sites that are neighbours of one another.

4.1.2 Gibbsian prior distribution

Let a Random Field Z be defined as a set of random variables $Z = \{Z_i, \forall i \in S\}$ where $Z_i \in [1, K]$. For every site $i \in S$, let $N(i)$ represent the set of its neighbours (see 4.1.1) and $z_{S-\{i\}}$ a realization of the field restricted to $S - \{i\} = \{j \in S, j \neq i\}$. Z is a *Markov Random Field* if and only if it follows the Markov property at every site :

$$\forall i \in S, P_G(z_i | z_{S-\{i\}}) = P_G(z_i | z_j, j \in N(i)) \quad (1)$$

Equation (1) states that the realization of the field, z_i at any site $i \in S$ can be fully determined using only the state of its neighbours $N(i)$. In other words the probability that a “cube” is in a given state depends only upon the state of its neighbours.

The Hammersley-Clifford theorem state that if Z is a Markov Random Field, the joint distribution of the field P_G follows a Gibbs distribution so that :

$$\begin{aligned}
P_G(\mathbf{z}; \boldsymbol{\beta}) &= W(\boldsymbol{\beta})^{-1} \exp(-H(\mathbf{z}; \boldsymbol{\beta})) \\
&= \frac{e^{-H(\mathbf{z}; \boldsymbol{\beta})}}{\sum_{\mathbf{z}'} e^{-H(\mathbf{z}'; \boldsymbol{\beta})}} \quad (2)
\end{aligned}$$

with $H(\mathbf{z})$ the Energy function summed over the cliques C of the graph G . Since we are working with a first order neighbouring system, C is the set of all pairs of sites (i, j) that are neighbours. We chose to consider H as a function of vector $\boldsymbol{\beta} = (\beta_1, \dots, \beta_K)$ containing K parameters, one per cluster and $v_{i,j}$ a potential function set to 1 in our method.

$$H(\mathbf{z}) = - \sum_{i \in S} \beta_{z_i} \sum_{\substack{i,j \\ \text{neighbours}}} v_{i,j} \times \mathbf{1}_{[z_i = z_j]} \quad (3)$$

The denominator in (2) where \mathbf{z}' represents all the possible realizations of the field is a normalizing constant referred to as $W(\boldsymbol{\beta})$.

4.1.3 Single and multiple beta models in a biological context

This model is closely related to a K -colour Potts model [82]. However, the unusual nature of the data used in this thesis lead to the idea of extending the model used mainly in the field of image segmentation. Indeed, the K -colour Potts model defines only one general spatial coherency parameter β [72, 86]. Importantly, the method presented here was extended by assigning one β per cluster so that:

$$\boldsymbol{\beta} = \text{diag}(\beta_1, \dots, \beta_K)$$

Interestingly, equation (3) is a decreasing function of every component of $\boldsymbol{\beta}$ and of the number of neighbouring “cubes” in the field having the same class. This Energy function thus favours spatially regular partitions and a higher value of β_h , with $1 \leq h \leq K$ will amplify the smoothing effect, or coherence over cluster h .

The reason why the model was extended to a multiple $\boldsymbol{\beta}$ parameter model, is inherent to the data used in this thesis. The first motivation is purely cytological. Indeed, in a biological context, it is expected that some tissues will be more spatially coherent than others. As mentioned in 1 and visualized in figure 4, tissues composed of different

cell types may be interacting differently with their neighbours. For example, differentiated neural cells with long axons are likely to be in contact with numerous other cell types they go through leading to a low value for β . The second motivation for the extended model finds its root in the cell model described in 2.1. Indeed, the single cell nature the in-situ dataset is flawed by the inaccuracy of the cell model and as visualized in figure 6 some "cubes" may have inconsistent gene expression patterns. This will be especially true for cell types where cells are big.

4.1.4 *Summary of the prior distribution parameters*

From this prior distribution are defined K unknown parameters $\beta = \text{diag}(\beta_1, \dots, \beta_K)$ to be estimated by the model. It is important to note at this point that $W(\beta)$ is summed over all possible realizations of the field Z , this is an exponentially complex sum as the cardinality of S rises. Therefore the computation of the normalizing factor becomes intractable very quickly. To address this problem, we are going to need to make some approximations in order to compute this quantity (see Mean Field Approximations).

4.2 THE EMISSION MODEL

We have described the prior distribution of a Markov Random Field representing our partition, we now need to describe the relationship between Z and the data.

4.2.1 *Conditional independence in the observed data*

As Z is unknown a priori and represents the partition, let Y be a set of random variables representing the observations (the in-situ hybridization data). The model requires a conditional independence assumption with regard to the observations Y given the partition Z so that, with f_{z_i} the density function relative to cluster $z_i, i \in S$ (the realization of the field at node i):

$$\begin{aligned}
p(\mathbf{y} \mid \mathbf{z}; \Theta) &= \prod_{i \in S} p(y_i \mid z_i; \Theta) \\
&= \prod_{i \in S} f_{z_i}(y_i \mid z_i; \Theta)
\end{aligned} \tag{4}$$

Equation 4 defines one unknown parameter per cluster: $\Theta = (\theta_1, \dots, \theta_K)$. It is interesting to note that this part of the model is equivalent to an independent mixture model [50]. Indeed, hidden Markov models can be viewed as independent mixture models where Z is a set of independent, identically distributed random variables, which happens when $\beta = 0$.

Given a particular cluster $h \in [1, K]$ and M the set of considered genes, gene expression for each gene $m \in M$ in each cluster is modelled a Bernoulli distribution with parameter $\theta_{h,m}$. This leads to one unknown Bernoulli parameter per gene per cluster so that :

$$\begin{aligned}
\Theta &= (\theta_1, \dots, \theta_K) \\
&= \begin{pmatrix} \theta_{1,1} & \dots & \theta_{1,K} \\ \vdots & \ddots & \vdots \\ \theta_{M,1} & \dots & \theta_{M,K} \end{pmatrix}
\end{aligned}$$

4.2.2 Full likelihood of the Hidden Markov random field model

The conditional density function $f_i, i \in S$ can be expressed as :

$$\begin{aligned}
f_i(y_i \mid z_i; \Theta) &= f_i(y_i \mid z_i; \theta_{z_i}) \\
&= \prod_{m \in M} \theta_{z_i, m}^{y_{i,m}} \times (1 - \theta_{z_i, m}^{1-y_{i,m}})
\end{aligned} \tag{5}$$

Looking at both fields Z and $Y \mid Z$ together, the complete likelihood of the model is expressed as :

$$\begin{aligned}
P_G(\mathbf{y}, \mathbf{z} \mid \Theta, \beta) &= f(\mathbf{y} \mid \mathbf{z}, \Theta) P_G(\mathbf{z} \mid \beta) \\
&= \frac{\exp\{-H(\mathbf{z} \mid \beta) + \sum_{i \in S} \log f_i(y_i \mid z_i, \theta_{z_i})\}}{\sum_{\mathbf{z}'} e^{-H(\mathbf{z})}}
\end{aligned} \tag{6}$$

Because equation (6) is a Gibbs distribution, using the Hammersley-Clifford theorem we can conclude that the conditional field Y given $Z = z$ is another a Markov Random Field with the Energy function

$$H(z | \mathbf{y}, \beta, \Theta) = H(z | \beta) - \sum_{i \in S} \log f_i(y_i | z_i, \Theta)$$

In our case, the goal is to recover the unknown realization of $Z : z$. To this end we need to maximize the values of all the parameters of the model $\psi = (\Theta, \beta)$. Additionally, choosing the unknown value K will also be crucial (ref to chosing K).

4.3 PARAMETER ESTIMATION USING THE EM ALGORITHM

As mentioned before, the aim is to assign each cell i to one of the K possible clusters. To do so, it is interesting to consider the Maximum Posterior Marginal (MPM) that maximizes $P(Z_i = h | \mathbf{y}, \psi)$, where the ψ are unknown and need to be estimated.

The Expectation Maximisation [19] (EM) principle can be applied to this end. After initializing the clusters z , choosing ψ^{l+1} at iteration $(l + 1)$ allows the maximization of the model's expectation $Q(\psi | \psi^l)$ defined as:

$$\begin{aligned} Q(\psi | \psi^l) &= \sum_z p(z | \mathbf{y}; \psi^l) \log p(\mathbf{y}, z; \psi) \\ &= \underbrace{\sum_z p(z | \mathbf{y}; \psi^l) \log p(\mathbf{y} | z; \Theta)}_{R_y(\Theta | \psi^l)} \\ &\quad + \underbrace{\sum_z p(z | \mathbf{y}; \psi^l) \log p(z | \beta)}_{R_z(\beta | \psi^l)} \end{aligned} \quad (7)$$

The decomposition in (7) allows to consider separately the maximization of $R_y(\Theta | \psi^l)$ and $R_z(\beta | \psi^l)$:

$$\begin{aligned} \Theta^{l+1} &= \arg \max_{\Theta} R_y(\Theta | \psi^l) \\ \beta^{l+1} &= \arg \max_{\beta} R_z(\beta | \psi^l) \end{aligned}$$

R_y can be estimated in the E step by using equation (4) so that:

$$\begin{aligned}
R_y(\Theta \mid \psi^l) &= \sum_z p(z \mid \mathbf{y}; \psi^l) \sum_{i \in S} \log f_{z_i}(y_i; \Theta) \\
&= \sum_{i \in S} \sum_{h=1}^K [\log f_h(y_i; \Theta)] p(Z_i = h \mid \mathbf{y}; \psi^l)
\end{aligned}$$

Therefore, at each iteration computing the following probably is required in the E step:

$$t_{ih}^{m+1} = p(Z_i = h \mid \mathbf{y}; \psi^l)$$

Computing this conditional probability is problematic because of the dependence between neighbouring “cubes”, and an exact value cannot be obtained without considerable computing resources. Indeed, each point being dependent upon its neighbours, and the neighbours being themselves dependent upon their neighbours, unsurprisingly computing these conditional probabilities becomes exponentially complex as the number of connected nodes in the graph grow. Additionally, as mentioned previously, the problem is the to compute the value of the normalizing constant $W(\beta)$.

To compute those quantities, approximations are needed. Methods to do so include Besag’s pseudo-likelihood [12] to compute $W(\beta)$, and simulating the posterior distribution of Z given \mathbf{y} with the parameters at iteration (l) , with a Gibbs sampler to estimate t_{ih}^{m+1} [16].

However, another method exists, the mean field approximation originally proposed in the field of statistical mechanics. Since then, it has been used in a variety of fields including computer vision [84] and more recently to approximate the distribution of both $W(\beta)$ (with a single β) and t_{ih}^{m+1} [87]. I present here the extension of this method to a model with one β parameter per cluster.

4.4 MEAN FIELD APPROXIMATIONS

The idea behind this approximation is to compute intractable quantities at any point $i \in S$ by setting all the other sites in the field to their mean values. Keeping in mind the Markov property expressed in equation (1), when considering a single site $i \in S$, setting all the other sites in the graph to a defined value is equivalentm, in the case

of a MRF, to setting the values of $N(i)$ only.

When computing t_{ih}^{m+1} , the mean fields approximation yields the following fixed point equation for $i \in S$ and $1 \leq h \leq K$ [18]:

$$t_{ih}^{m+1} \approx \frac{f_h(y_i; \theta_h^m) \exp\{\beta_h^m \sum_{j \in N(i)} t_{jh}^{m+1}\}}{\sum_{u=1}^K f_u(y_i; \theta_u^m) \exp\{\beta_u^m \sum_{j \in N(i)} t_{ju}^{m+1}\}} \quad (8)$$

write the fixed point algorithm

For the normalizing constant $W(\beta)$, by applying the mean-field approximation, using equation (3), $W(\beta)$ can be written as:

$$W(\beta) = \sum_{z'} \exp(-H(z')) \approx \sum_{i \in S} \sum_{z_i} \exp(-H(z_i)) = \sum_{i \in S} \sum_{z_i} \exp(\beta_{z_i} \sum_{N(i)} [z_i = z_j])$$

With this new set of equations, it becomes possible to estimate all quantities needed in the E step in order to compute the model's expectation.

4.5 MAXIMIZATION

After the E step, maximizing ψ is relatively straightforward. For Θ , once the $t_{ih}^m = p(Z_i = h \mid \mathbf{y}; \psi^l)$ have been computed during the E-step, those probabilities may be used to assign each cell to its most probable cluster at step l . Once the new partition is created, the values of Θ that maximize the model's expectation can be computed iteratively for cluster $h \in [1, K]$ and gene $m \in M$ with function $\text{Expr}_{h,m}$ the number of cells expressing gene m in cluster h and function Num_h the total number of cells in cluster h .

$$\theta_{m,h}^{l+1} = \arg \max_{\Theta} R_y(\Theta \mid \psi^l) = \frac{\text{Expr}_{h,m}}{\text{Num}_h}$$

In order to maximize β^{l+1} , an iterative approach such as the gradient ascent algorithm, the positive version of the gradient descent algorithm [?] can be used for each $\beta_h^{l+1}, h \in [1, K]$ over the function $R_z(\beta \mid \psi^l)$.

write the gradient ascent algorithm

The described EM algorithm, summarized in write EM algo leads, after convergence, to a partition over K clusters that finds a local maximum of the model's expectation. Importantly, maximum reached is

only a local one as indeed the EM algorithm does not guarantee to reach the global maximum of a function. It is interesting to note that this fact makes the initialization of the algorithm a crucial step as I will develop in [link to init schemes](#).

The partition computed is still over K clusters, consequently, a method to choose the value K is still needed.

4.6 ESTIMATING K

Without any prior knowledge, choosing the right number of clusters K is challenging. I decided to use an *a posteriori* method relying on the final log Likelihood of the model derived from equation (6):

$$\log L(\boldsymbol{\psi}) = \log P_G(\mathbf{y}, \mathbf{z} \mid \Theta, \beta)$$

Because $\log L(\boldsymbol{\psi})$ monotonically increases with the number of parameters of the model, the BIC approach penalizes the addition of new parameters to the model. Let P be the total number of parameters in the model and N the cardinality of S , the BIC is expressed as:

$$-2 \log L(\boldsymbol{\psi}) + P \log N$$

By computing the final likelihood for a large range of possible K values, the minimal resulting BIC will be chosen as the optimal number of classes, \hat{K} . When applied to the biological data however, this approach is not ideal as I will describe later in this thesis (see [ref biological BIC](#)) but yields good results when applied to simulated data (see simulations [ref next chapter](#)).

4.7 SUMMARY

The goal was to allocate the $S = 32,203$ “cubes” described above in chapter 2 into K clusters, where K is unknown, using the binarised matrix of $M = 86$ gene expression measurements, Y . To incorporate spatial information into the clustering scheme, I assumed that Z , the (latent) vector of length S that describes the allocation of cells to clusters, satisfies a first-order Markov Random Field (MRF), where

the probability that a cell is allocated to a given state depends only upon the states of its immediate neighbours (Figure ??). Additionally, within cluster h ($h \in [1, K]$), I assumed that the expression of gene m follows a Bernoulli distribution with parameter $\theta_{m,h}$. The $M \times K$ matrix Θ denoting the full set of Bernoulli parameters. In a typical MRF, the degree of spatial cohesion is determined by a single parameter β , which is assumed to be constant for all clusters [72, 86]. However, in the context of tissue organisation, it is reasonable to expect that the degree of spatial cohesion will differ between clusters; consequently, a separate value of β is estimated for each of the K clusters.

To estimate the parameters of the model an Expectation-Maximisation (EM) based approach has been used in conjunction with mean-field approximations to infer intractable values [15]. Finally, to choose the optimal number of clusters, K , the use of the Bayesian Information Criterion (BIC) has been proposed.

The next step is to validate the method's behaviour and to assess the quality of the results compared to the other non-spatial clustering schemes described in chapter 3. Consequently, I decided to perform a simulation study that I will describe in the next chapter.

METHOD VALIDATION AND PERFORMANCE ANALYSIS ON SIMULATED DATA

Complete all the stuff that is missing. Separate the initialization and method comparison analysis. Rewrite because this is copy paste from paper, check the references, try to fit figures in the text

5.1 SIMULATING DATA WITH A SPATIAL COMPONENT

Simulating data with a spatial component is a non-trivial problem. Existing methods rely on MCMC approaches as described in [16]. However, in this case with a relatively large number of nodes in the graph ($\sim 34,000$), this is computationally expensive. To overcome this problem, I exploited the fact that the *Platynereis* dataset already possesses a spatial component. As outlined in Figure 17, the simulation starts by clustering the gene expression data using different values of K with the HRMF method described in chapter 4 and by storing the resulting parameter estimates. Subsequently, I use the values of the estimated parameter Θ to simulate binarised gene expression data from K clusters where, for cluster h , the expression of gene m is simulated from a Bernoulli distribution with parameter $\theta_{m,h}$ as described in 5.1.1. This non spatial simulated data is then reintroduced in the spatial context of the biological data 5.1.2 leading to a simulated dataset with all parameters being fully determined. In the next paragraphs, I will describe each step of this simulation scheme.

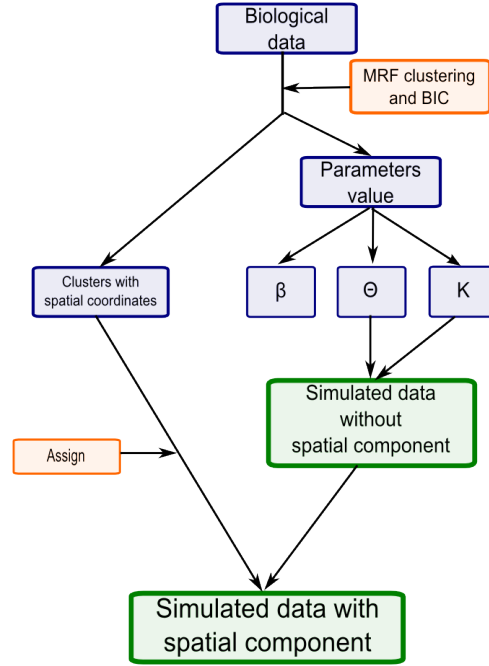


Figure 17.: **Simulation scheme used to generate gene expression data with a spatial component and known parameters.** The values of Θ are used to generate a dataset of clusters with the same gene expression profile as the reference. Each simulated cell is then assigned to its corresponding spatial localization so that the simulated data keeps the spatial component of the biological data.

5.1.1 Simulating non spatial gene expression data

The first step of the simulation scheme is to simulate binary gene expression data for $S = 32,203$ sites and $M = 86$ genes belonging to K clusters. Each cluster will be assigned N_h sites with $h \in [1, K]$. Given the emission model described in chapter 4, for each gene and each cluster, a $K \times M$ matrix Θ is needed where each $\theta_{h,m}$ represents a Bernoulli parameter or the probability for each site in cluster h to express gene m .

In order to generate a biologically coherent Θ matrix, the clustering method is applied to the biological data for K clusters and the resulting final Θ matrix is kept to simulate new data. The clustering on the biological data also gives the number of cells per cluster $N_h, \forall h \in K$.

Once the parameters values are available it is relatively straight forward to simulate N_h sites per cluster following the Bernoulli distributions thus defined resulting in $S = 32,203$ sites with known Θ parameters [Appendix with R script](#).

5.1.2 *Introducing a known spatial context*

Each simulated is then assigned to the same spatial location as the corresponding “cube” in the biological dataset, meaning that both the simulated and the biological datasets have the same neighbouring graph. The rationale behind this simulation scheme is to allow the model to validate itself. By replacing simulated gene expression data equivalent to the biological one in the same spatial context, the hypothesis is that the set of parameters β will stay relatively stable when the simulated data is clustered.

5.1.3 *Expected results*

Given this simulation scheme, the expected results after clustering the simulated data, is to find a strong conservation for the all the parameter $\psi = \{\Theta, \beta\}$ between the “true” values $\hat{\psi}$ obtained from clustering the biological data and the estimated values $\tilde{\psi}$ obtained after clustering the simulated data.

Importantly, when clustering the simulated data with the HMRF method, a different random initialization than the one used to cluster the biological data in the first place was used in order to avoid any bias.

5.2 COMPARING CLUSTERING RESULTS USING THE JACCARD SIMILARITY COEFFICIENT

5.2.1 *Theoretical problem in comparing clustering results*

To compare clustering results, a lot of metrics exist to estimate the similarity between two lists of clusters. One of the widely used ones is the Jaccard coefficient [?]. For two clustering results, A and B, the Jaccard coefficient is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Although theoretically very simple, in practice computing this metric is not trivial. Indeed, depending on the clustering method used and on the initialization used for clustering methods that require an initialization step like the independent mixture models or the HRMF method presented in chapter 4, even if the clustering results are 100% identical, they are going to be misaligned.

This means that for method A *cluster 1* could for example be *cluster 5* in method B. In order to compute the Jaccard coefficient and compare clustering results, it is necessary to be able to align partitions. To this end I used a similarity specificity matrix approach as described in the next paragraph.

5.2.2 Alignment via similarity-specificity matrix

The “count” matrix D and the “similarity/specificity” matrix H for two sets of clusters z and z' with K clusters each so that $z = \bigcup_{h \in [1, K]} c_h$ and $z' = \bigcup_{h \in [1, K]} c'_h$ are defined as:

$$D = \begin{pmatrix} |c_1 = c'_1| & \dots & |c_1 = c'_K| \\ \vdots & \ddots & \vdots \\ |c_K = c'_1| & \dots & |c_K = c'_K| \end{pmatrix}$$

and

$$H_{ij} = \frac{D_{ij}}{\sum_a D_{aj} \sum_b D_{ib}}$$

With z the set of reference clusters, the ones other clustering results will be aligned to (in the case of the simulation study, z is the set of “true” clusters obtained after clustering the biological data), for each row of the matrix H , the column with the highest value is then select as the corresponding cluster.

In the case of two sets of clusters being extremely similar this approach will align the cluster sets. However some errors may arise if the two cluster sets are quite different. For example if one cluster h_{z_1} in z is split into two clusters $h_{z'_4}, h_{z'_5}$ in z' , they will both be assigned to h_{z_1} after alignment meaning that, because K is the same for z and z' one cluster in z will have no corresponding clusters in z' .

This sort of error is central to clustering results analysis and unfortunately unavoidable. Therefore, the resulting Jaccard coefficient will

not necessarily be linearly correlated with the similarity between the reference clusters and the clusters under study, instead it will have a tendency to worsen faster than the similarity due to the biases created in this alignment step. It remains however a good indication of the divergence between clustering sets. An example of aligning the clusters is shown through the values of Θ by comparing figures ?? and ??.

The first thing I need to do now is to validate the correct estimation of the model's parameters. I will describe this essential step in the next section.

5.3 VALIDATION OF PARAMETERS ESTIMATION AND MODEL CHOOSING

5.3.1 Estimation of Θ

| 5.3.1 Estimation of bold

To validate the consistency in estimating the values of Θ , it is important to compare the "true" values used to simulate the data to the ones obtained after clustering the simulated data. A simple example with $K = 6$ is presented in figure

[bth] ["Proximity" between values of Θ before alignment]
 [width=.45]gfx/chapter5/heatmap6.png ["Proximity" between
 values of Θ after alignment]
 [width=.45]gfx/chapter5/heatmap6_aligned.png Validating the estimation of Θ for $K=6$. On the axis are
 #I

#I

It is also interesting to note that comparing the similarity between the inferred and true clusters with the Jaccard coefficient also implicitly assesses the accuracy of the estimation of Θ : if the inferred and true clusters are identical, the estimates of Θ must be equal to the true values. In practice, a Jaccard coefficient of 1 implies perfect agreement. Figure 20 shows the value of the Jaccard coefficient for several clustering methods including the HRMF (red points). The very high value of the Jaccard coefficient confirms that the values of Θ are consistently estimated for $K \in [4, 80]$ (see ?? for the full description of figure 20)

5.3.2 *Estimation of beta*

To determine how accurately the β parameters are estimated, as shown in Figure 18 it is possible to compare the true and inferred mean values of β for different values of K . The values of β increase with K , which is to be expected since more clusters implies the existence of more transition areas, thus making an increase of β necessary to maintain the optimal spatial coherency of the model.

Interestingly, Figure 18 also shows a slight but consistent underestimation of β . This can be explained by noting that the simulation scheme used may reduce the spatial coherency within clusters. Specifically, as illustrated in Figure 19, clusters may not display homogeneous expression of a given gene: instead, depending upon the value of θ , a gene will be expressed only in a fraction of cells. In reality, the cells in which such genes are expressed may have a coherent spatial structure within the biological cluster that is lost in the simulation, thus explaining the consistently smaller value for β that are estimated.

To confirm this explanation, I performed a second simulation using the parameter values estimated from the first simulation as a reference. In this context no further loss of spatial coherency was expected, which was indeed confirmed as shown by the blue curve in Figure 18.

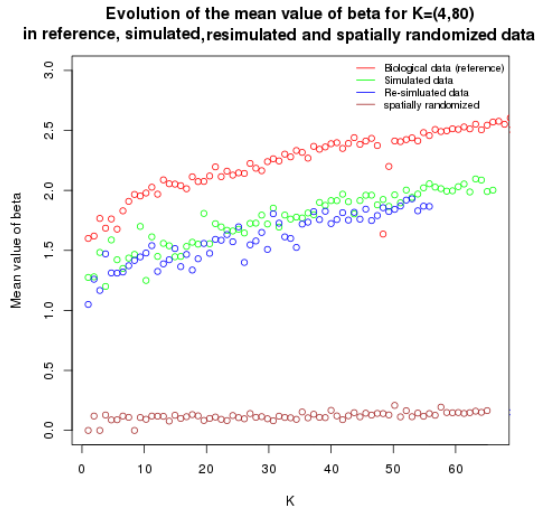


Figure 18.: **Validating the estimation of beta.** This figure shows the evolution for $K \in [4, 80]$ of the mean value of β across all the clusters. The red dots represent the biological data clustering (i.e the reference in our simulations scheme). The green dots represent the results obtained after clustering simulated data, which shows an underestimation of β . To confirm that this underestimation come from the simulation scheme and not the clustering method, the simulated data was used as the reference to generate a "second generation" of simulated data, suppressing the simulation scheme bias (see Figure 19). The results of this re-simulation are shown by the blue dots, which exhibit no underestimation of β . Finally the brown dots represent the mean value of β on the same simulated data but spatially randomized, as expected the β are now estimated to 0.

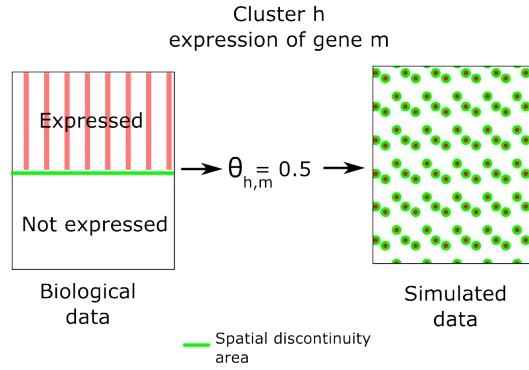


Figure 19.: **Decrease in spatial coherency due to the simulation scheme.**

For an example cluster h , gene m may only be expressed in half of the cells. This will yield $\theta_{h,m} = 0.5$. However, in the biological data, the cells expressing gene m may be spatially coherent (i.e., located close to one another), leading to a reduced area of expression discontinuity (the green line). By contrast, in the simulated data the expression of such a gene will lose its spatial coherency, leading to an increased area of expression discontinuity. The number of cells having a neighbour with some differences in the gene expression pattern is directly linked to the value of β_h through the energy function described in chapter 4. This explains the underestimation of β observed in Figure 18.

To validate further our estimation of β , I randomized the coordinates of the simulated “cubes” to lose any spatial component before re-clustering the data. As expected, we observed that the estimates of β were very close to 0 for all clusters (Figure 18), as well as there being very similar Jaccard coefficient values (relative to the true values) for the independent mixture and the MRF model as shown in figure 20. Both of these observations provide confidence in our assertion that the spatial component plays an important role in the fit.

5.3.3 Choosing K

Finally, assessing the ability of the model to choose the correct number of clusters, K is crucial. To do this, the “true” number of clusters underlying the simulated data \hat{K} was compared to the chosen value, \tilde{K} obtained after applying the BIC method (see 4). The results for two representative choices of K are shown in Figure 21 and demonstrate that our clustering approach, in conjunction with the BIC, is

able to accurately determine the optimal number of clusters.

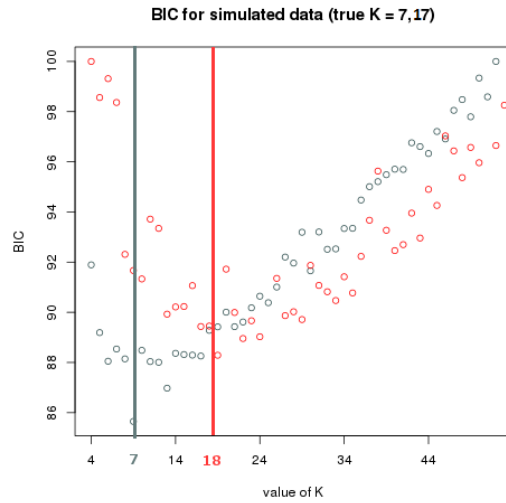


Figure 20.: **Jaccard coefficient between “true” and resulting clusters on the simulated data with different methods and initializations.** Panel A compares the performance of the MRF method with a randomly initialization with an independent mixture model also with a random initialization, the MRF method initialized with the hClust classification and hClust alone on data simulated with a spatial component. Panel B shows the Jaccard coefficient for the MRF method and independent mixture model both with a random initialization; in this case both methods are applied to simulated data that lacks a spatial component.

5.4 METHOD PERFORMANCE AND INITIALIZATION

As pointed out in the above paragraph for this method initialization is a key step of the HMRF clustering, working on the simulated data allows comparing the results of the clustering with a variety of initialization schemes.

5.4.1 The EM principle and local maximum

As explained in chapter 4, the HMRF clustering relies on Maximum Posterior Marginal (MPM) approach, and the EM algorithm is used to estimate the unknown parameter values. The likelihood function that needs to be maximised possesses many stationary points of different natures. Thus, convergence to the global maximum with the

EM algorithm, depends strongly on the parameters initialisation. To overcome this problem, different initialisation strategies have been proposed and investigated (see for instance [? ? ?]).

Indeed, if the procedure is initialized with a set of clusters that are close to a local maximum in the likelihood function, the EM algorithm will converge to this local maximum and will never reach the global maximum of the model.

5.4.2 *Random initialization vs Hclust initialization*

To shed some light on the initialization scheme issue I decided to compare two theoretically opposed initialization schemes :

- A random approach: 10.000 random initialization were generated for $K \in [4, 70]$ and for each one the initial likelihood of the model was computed, the random initialization with the highest initial likelihood was selected to apply the EM algorithm.
- A directed approach: the data was clustered using the non-spatial hClust method described in chapter 3 and used the resulting set of cluster to initialize the EM algorithm

The results are shown in figure 20 (black and green dots on panel A). Looking at the effect of the initialization scheme on the quality of the resulting clusters through the Jaccard coefficient for $K \in [4, 70]$ it is clear that, unsurprisingly considering the problem of the EM clustering not guaranteeing to reach the global maximum of the likelihood function, the random initialization scheme performs a lot better than the directed initialization scheme with hClust. Indeed, for the HMRF randomly initialized, the average Jaccard coefficient is around 0.8 when it averages 0.7 only when initialized with hClust.

Following this observation, for the rest of this thesis and especially in chapter [ref chapter biological analysis](#), the HMRF method will be randomly initialized.

5.5 METHOD PERFORMANCE COMPARED TO HCLUST AND INDEPENDENT MIXTURE MODELS

Because the simulated data provides a clear set of "true" clusters and parameter values, it allows the HRMF clustering to be compared to

other clustering methods in terms of clustering quality through the Jaccard coefficient. Additionally, computing time being a key factor for large dataset, I provide in this section some key figures about the execution time of the different methods.

5.5.1 *Quality of the clustering results*

The resulting Jaccard coefficients obtained by comparing the “true” clusters and the clusters generated by different clustering methods are shown in Figure 20 for $\tilde{K} \in [4, 70]$. The HRMF method, when used with a random initialization scheme, has an average Jaccard coefficient of 0.8, and clearly demonstrates better performance than the other methods. The second best performing method is the independent mixture model with a random initialization, which has an average Jaccard coefficient of 0.7. Since the independent mixture approach is equivalent to the MRF with all the β parameters set equal to 0 (i.e., without a spatial component) this suggests that accounting for the spatial aspect yields improved results. Given this, it is perhaps unsurprising that hClust also performs relatively poorly.

Although as mentioned previously, the Jaccard coefficient may not be linearly correlated with the quality of the clustering results because of the alignment step which can create biases, it is still a good indicator of the methods’ performances and it would seem that the HMRF consistently outperforms the other methods tested.

5.5.2 *Discussion*

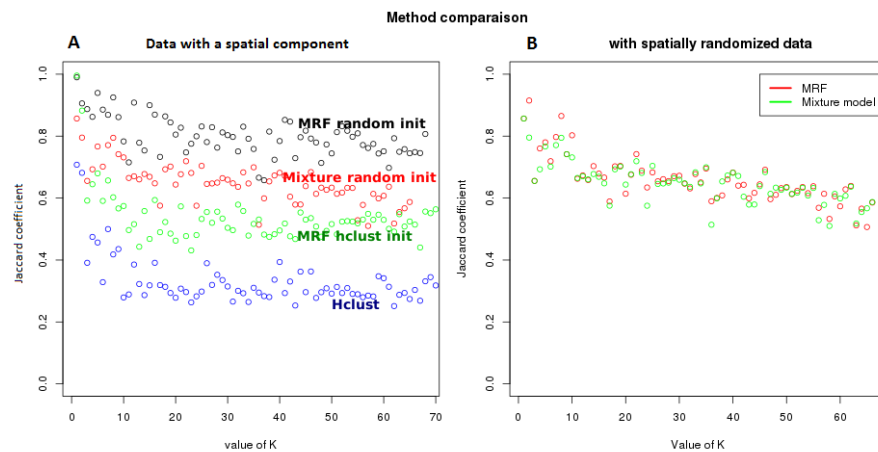


Figure 21.: Jaccard coefficient between “true” and resulting clusters on the simulated data with different methods and initializations. Panel A compares the performance of the MRF method with a randomly initialization with an independent mixture model also with a random initialization, the MRF method initialized with the hClust classification and hClust alone on data simulated with a spatial component. Panel B shows the Jaccard coefficient for the MRF method and independent mixture model both with a random initialization; in this case both methods are applied to simulated data that lacks a spatial component.

Part II

APPENDIX

INPUT FILE FORMATS FOR BIOWEB_{3D}

A.1 DATASET FILE SPECIFICATION

When the user adds a new *Dataset* file, a new Dataset section is created in the “Data” panel of the application. Each dataset file contains one dataset.

A.1.1 JSON format

The *dataset* file should have a root object called “dataset” which contains:

- The “name” property of the dataset (*e.g.*, “my dataset”);
- The “chain” parameter, which should be set to *true* if the points are connected (the default value is *false*) - the data will be considered sequentially, with each point connected by a solid line to the previous and next point according to its order in the dataset file;
- The “points” property, which is a two dimensional array representing a list of (x,y,z) vectors that define the co-ordinates of the points.

Listing [A.1](#) is an example of a minimal 3 points dataset file.

A.1.2 XML format

The *dataset* XML format used is very similar to the previously defined JSON format. The file must have a root object called “<dataset>” which contains:

- The “<name>” property of the dataset (*e.g.*, “my dataset”);
- The “<chain>” parameter, which should be set to *true* if the points are linked (the default value is *false*) - the data will be considered sequentially, with each point connected by a solid

Listing A.1: Json dataset file

```
{ "dataset" : {  
    "name" : "my superb dataset",  
    "chain" : true,  
    "points" :  
    [  
        [  
            0.5,  
            100,  
            -50.5  
        ],  
        [  
            200,  
            10,  
            0.0  
        ],  
        [  
            3,  
            250.15,  
            15  
        ]  
    ]  
}
```


Listing A.2: XML dataset file

```

<?xml version="1.0" ?>
<dataset>
  <name>my superb dataset</name>
  <chain>true</chain>
  <points>
    <point>
      <x>0.5</x>
      <y>100</y>
      <z>-50.5</z>
    </point>

    <point>
      <x>200</x>
      <y>10</y>
      <z>0.0</z>
    </point>

    <point>
      <x>3</x>
      <y>250.15</y>
      <z>15</z>
    </point>
  </points>
</dataset>

```

line to the previous and next point according to its order in the dataset file;

- The “<points>” property, which contains all the single “<point>” elements that define the dataset. Each “<point>” has three properties to define its spatial location, namely “<x>”, “<y>” and “<z>”.

Listing A.2 contains the same minimal dataset as Listing A.1 but formatted in XML.

A.1.3 CSV format

Each line represents a point and the three coordinates on each line must be separated by “comma” characters.

As an example, listing A.3 carries the same information as the JSON

Listing A.3: CSV dataset file

```
0.5,100,-50.5
200,10,0.0
3,250.15,15
```

file in Listing [A.1](#). We note that although the spatial information remains the same it is not possible to set a name or to connect the points within a CSV file input.

A.2 INFORMATION LAYER FILE SPECIFICATION

The *Information layer* file contains information about the points described in the Dataset file. The information in this file has to be given in the same order as the points defined in the Dataset file.

A.2.1 JSON format

The *information layer* files must have a root element named “information”. Since one information file can define multiple information sets, the structure below “information” is a list. Each element of the list is structured as follows:

- The “name” property (optional);
- The “numClass” property, which indicates the number of different classes the data will be assigned to;
- The “labels” property, which defines a list of names for the “numClass” classes previously defined (optional);
- The “values” property, which defines the class of each point in the dataset. As points do not have single IDs, this property must be in the same order and have the same length as the points defined in the *dataset* file.

For example coming back to the 3 points defined in Listing [A.1](#), two information layers could correspond to:

- one clustering algorithm that puts the first two points together in class one and the third point alone in a second class
- a second clustering algorithm that puts each point in a separate class

Listing A.4: JSON information layer file

```

{ "information" :
  [
    {
      "name": "clustering algo 1",
      "numClass": "2",
      "labels" : [
        "Category 1",
        "Category 2"
      ],
      "values": [
        1,
        1,
        2
      ]
    },
    {
      "name": "clustering algo 2",
      "numClass": "3",
      "values": [
        1,
        2,
        3
      ]
    }
  ]
}

```

In this case the Information layer file would look like Listing [A.4](#).

A.2.2 XML format

The *information layer* XML format used is very similar to the previously defined JSON format. The *information layer* files must have a root element named "<information>". Since one information file can define multiple information sets, the structure below "<information>" is a list of "<set>" elements. Each "<set>" element is structured as follows:

- The "<name>" property (optional);
- The "<numClass>" property, which indicates the number of different classes the data will be assigned to;

Listing A.5: XML information layer file

```

<?xml version="1.0" ?>
<information>
  <set>
    <name>clustering algo 1</name>
    <numClass>2</numClass>
    <labels>
      <label>Category 1</label>
      <label>Category 2</label>
    </labels>
    <values>
      <value>1</value>
      <value>1</value>
      <value>2</value>
    </values>
  </set>
  <set>
    <name>clustering algo 2</name>
    <numClass>3</numClass>
    <values>
      <value>1</value>
      <value>2</value>
      <value>3</value>
    </values>
  </set>
</information>

```

- The “<labels>” property, which contains as many individual “<label>” properties as the number of different classes. Each “<label>” defines the names for one class (optional);
- The “<values>” property, which contains all the single “<value>” properties, each one defining the class of each point in the dataset. As points do not have single IDs, the “<value>” properties must be in the same order and have the same length as the points defined in the *dataset* file.

Listing A.5, carries the exact same information as Listing A.4.

CSV FORMAT Each column represents the class to which a point belongs. The separation character between columns must be a “comma”. Listing A.6, carries the same information as Listing A.4. Note that it

Listing A.6: CSV information layer file

```
1,1  
1,2  
2,3
```

is not possible to use the "labels" or "name" properties available in Listing [A.4](#) within a CSV information layer file.

BIBLIOGRAPHY

- [1] bioweb3d online. URL <http://www.ebi.ac.uk/~jbpettit/bioWeb3D>.
- [2] Compatibility table for webgl. URL <http://caniuse.com/webgl>.
- [3] bioweb3d on github. URL <http://github.com/jbogg/bioWeb3D>.
- [4] United states computer emergency readiness team. URL <http://www.kb.cert.org/vuls/id/636312>.
- [5] Three.js - javascript 3d library. URL <http://mrdoob.github.com/three.js/>.
- [6] Webgl 1.0 specification. URL <https://www.khronos.org/registry/webgl/specs/1.0/>.
- [7] Xml applications and initiatives. URL <http://xml.coverpages.org/xmlApplications.html>.
- [8] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome Biology*, 11:R106, 2010. doi: 10.1186/gb-2010-11-10-r106. URL <http://genomebiology.com/2010/11/10/R106/>.
- [9] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):898–916, 2011.
- [10] James E Balmer and Rune Blomhoff. Gene expression regulation by retinoic acid. *Journal of lipid research*, 43(11):1773–1808, 2002.
- [11] WILLIAM P Bartlett and GARY A Banker. An electron microscopic study of the development of axons and dendrites by hippocampal neurons in culture. i. cells which develop without intercellular contacts. *The Journal of neuroscience*, 4(8):1944–1953, 1984.
- [12] Julian Besag. Statistical analysis of non-lattice data. *The statistician*, pages 179–195, 1975.

- [13] Paul D Boyer. The atp synthase-a splendid molecular machine. *Annual review of biochemistry*, 66(1):717–749, 1997.
- [14] Philip Brennecke, Simon Anders, Jong Kyoung Kim, Aleksandra A Kołodziejczyk, Xiuwei Zhang, Valentina Proserpio, Bianka Baying, Vladimir Benes, Sarah A Teichmann, John C Marioni, et al. Accounting for technical noise in single-cell rna-seq experiments. *Nature methods*, 2013.
- [15] Gilles Celeux, Florence Forbes, and Nathalie Peyrard. Em procedures using mean field-like approximations for markov model-based image segmentation, 2001.
- [16] B CHALMOND. An iterative gibbsian technique for reconstruction of m-ary images. *Pattern recognition*, 22(6):747–761, 1989.
- [17] RJ Colello and RW Guillery. The early development of retinal ganglion cells with uncrossed axons in the mouse: retinal position and axonal course. *Development*, 108(3):515–523, 1990.
- [18] M. Dang and G. Govaert. Spatial fuzzy clustering using EM and markov random fields. *International Journal of System Research and Information Science*, 8(4):183–202, 1998.
- [19] Arthur P Dempster, Nan M Laird, Donald B Rubin, et al. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal statistical Society*, 39(1):1–38, 1977.
- [20] Alexandru S Denes, Gáspár Jékely, Patrick RH Steinmetz, Florian Raible, Heidi Snyman, Benjamin Prud’homme, David EK Ferrier, Guillaume Balavoine, and Detlev Arendt. Molecular architecture of annelid nerve cord supports common origin of nervous system centralization in bilateria. *Cell*, 129(2):277–288, 2007.
- [21] Qiaolin Deng, Daniel Ramsköld, Björn Reinius, and Rickard Sandberg. Single-cell rna-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science*, 343(6167):193–196, 2014.
- [22] Adriaan WC Dorresteijs. Quantitative analysis of cellular differentiation during early embryogenesis of platynereis dumerilii. *Roux’s archives of developmental biology*, 199(1):14–30, 1990.
- [23] Jianping Fan, David KY Yau, Ahmed K Elmagarmid, and Walid G Aref. Automatic image segmentation by integrating

- color-edge extraction and seeded region growing. *Image Processing, IEEE Transactions on*, 10(10):1454–1466, 2001.
- [24] EA Feingold, PJ Good, MS Guyer, S Kamholz, L Liefer, K Wetterstrand, FS Collins, TR Gingeras, D Kampa, EA Sekinger, et al. The encode (encyclopedia of dna elements) project. *Science*, 306(5696):636–640, 2004.
- [25] Albrecht Fischer and Adriaan Dorrestijn. The polychaete platynereis dumerilii (annelida): a laboratory animal with spiralian cleavage, lifelong segment proliferation and a mixed benthic/pelagic life cycle. *Bioessays*, 26(3):314–325, 2004.
- [26] Antje Fischer, Thorsten Henrich, and Detlev Arendt. The normal development of platynereis dumerilii (nereididae, annelida). *Frontiers in zoology*, 7(1):31, 2010.
- [27] Tom C Freeman, Leon Goldovsky, Markus Brosch, Stijn Van Dongen, Pierre Mazière, Russell J Grocock, Shiri Freilich, Janet Thornton, and Anton J Enright. Construction, visualisation, and clustering of transcription networks from microarray expression data. *PLoS computational biology*, 3(10):e206, 2007.
- [28] Clay Fuqua, Matthew R Parsek, and E Peter Greenberg. Regulation of gene expression by cell-to-cell communication: acyl-homoserine lactone quorum sensing. *Annual review of genetics*, 35(1):439–468, 2001.
- [29] Manfred Gossen and Hermann Bujard. Tight control of gene expression in mammalian cells by tetracycline-responsive promoters. *Proceedings of the National Academy of Sciences*, 89(12):5547–5551, 1992.
- [30] Keren Guy. Development and molecular characterization of adult and larval eyes in platynereis dumerilii (polychaeta, annelida, lophotrochozoa). 2008.
- [31] Valentin Häcker. *Die pelagischen Polychaeten-und Achaetenlarven der Plankton-expedition...* Lipsius & Tischer, 1898.
- [32] Jörg D Hardege. Nereidid polychaetes as model organisms for marine chemical ecology. *Hydrobiologia*, 402:145–161, 1999.
- [33] M. J. Hartshorn. AstexViewer: a visualisation aid for structure-based drug design. *J. Comput. Aided Mol. Des.*, 16(12):871–881, Dec 2002.

- [34] Carl Hauenschild, G Czihak, A Fischer, and R Siewing. *Platynereis dumerilii: mikroskopische Anatomie, Fortpflanzung, Entwicklung*. Fischer, 1969.
- [35] Thomas H Hutchinson, Awadhesh N Jha, and David R Dixon. The polychaete platynereis dumerilii (audouin and milne-edwards): a new species for assessing the hazardous potential of chemicals in the marine environment. *Ecotoxicology and environmental safety*, 31(3):271–281, 1995.
- [36] Norman N Iscove, Mary Barbara, Marie Gu, Meredith Gibson, Carolyn Modi, and Neil Winegarden. Representation is faithfully preserved in global cDNA amplified exponentially from sub-picogram quantities of mRNA. *Nature biotechnology*, 20(9):940–943, 2002.
- [37] Ernst Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik A Hadrons and Nuclei*, 31(1):253–258, February 1925. ISSN 0044-3328. doi: 10.1007/bf02980577. URL <http://dx.doi.org/10.1007/bf02980577>.
- [38] Saiful Islam, Una Kjällquist, Annalena Moliner, Pawel Zajac, Jian-Bing Fan, Peter Lönnerberg, and Sten Linnarsson. Characterization of the single-cell transcriptional landscape by highly multiplex RNA-seq. *Genome research*, 21(7):1160–1167, 2011.
- [39] Saiful Islam, Amit Zeisel, Simon Joost, Gioele La Manno, Pawel Zajac, Maria Kasper, Peter Lönnerberg, and Sten Linnarsson. Quantitative single-cell RNA-seq with unique molecular identifiers. *Nature methods*, 2013.
- [40] HA John, ML Birnstiel, and KW Jones. RNA-DNA hybrids at the cytological level. *Nature*, 223(5206):582, 1969.
- [41] Stephen C Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.
- [42] ET Kaiser and FJ Kezdy. Amphiphilic secondary structure: design of peptide hormones. *Science*, 223(4633):249–255, 1984.
- [43] Zbynek Kozmik, Jana Ruzickova, Kristyna Jonasova, Yoshifumi Matsumoto, Pavel Vopalensky, Iryna Kozmikova, Hynek Strnad, Shoji Kawamura, Joram Piatigorsky, Vaclav Paces, et al. Assembly of the cnidarian camera-type eye from vertebrate-like components. *Proceedings of the National Academy of Sciences*, 105(26):8989–8993, 2008.

- [44] Dinesh B. Kulkarni, Mahesh M. Doijade, Chetan S. Devrukhkar, Ganesh R. Zilpe, and Rajesh R. Surana. Article: Netraris - a web based dicom viewer. *International Journal of Computer Applications*, 48(24):40–44, June 2012. Published by Foundation of Computer Science, New York, USA.
- [45] JE Landegent, De Wal, N Jansen In, RA Baan, JHJ Hoeijmakers, and M Van Der Ploeg. 2-acetylaminofluorene-modified probes for the indirect hybridocytochemical detection of specific nucleic acid sequences. *Experimental cell research*, 153(1):61–72, 1984.
- [46] E. S. Lein, M. J. Hawrylycz, N. Ao, M. Ayres, A. Bensinger, A. Bernard, A. F. Boe, M. S. Boguski, K. S. Brockway, E. J. Byrnes, L. Chen, L. Chen, T. M. Chen, M. C. Chin, J. Chong, B. E. Crook, A. Czaplinska, C. N. Dang, S. Datta, N. R. Dee, A. L. Desaki, T. Desta, E. Diep, T. A. Dolbeare, M. J. Donelan, H. W. Dong, J. G. Dougherty, B. J. Duncan, A. J. Ebbert, G. Eichele, L. K. Estin, C. Faber, B. A. Facer, R. Fields, S. R. Fischer, T. P. Fliss, C. Frenslley, S. N. Gates, K. J. Glattfelder, K. R. Halverson, M. R. Hart, J. G. Hohmann, M. P. Howell, D. P. Jeung, R. A. Johnson, P. T. Karr, R. Kawal, J. M. Kidney, R. H. Knapik, C. L. Kuan, J. H. Lake, A. R. Laramee, K. D. Larsen, C. Lau, T. A. Lemon, A. J. Liang, Y. Liu, L. T. Luong, J. Michaels, J. J. Morgan, R. J. Morgan, M. T. Mortrud, N. F. Mosqueda, L. L. Ng, R. Ng, G. J. Orta, C. C. Overly, T. H. Pak, S. E. Parry, S. D. Pathak, O. C. Pearson, R. B. Puchalski, Z. L. Riley, H. R. Rockett, S. A. Rowland, J. J. Royall, M. J. Ruiz, N. R. Sarno, K. Schaffnit, N. V. Shapovalova, T. Sivisay, C. R. Slaughterbeck, S. C. Smith, K. A. Smith, B. I. Smith, A. J. Sodt, N. N. Stewart, K. R. Stumpf, S. M. Sunkin, M. Sutram, A. Tam, C. D. Teemer, C. Thaller, C. L. Thompson, L. R. Varnam, A. Visel, R. M. Whitlock, P. E. Wohnoutka, C. K. Wolkey, V. Y. Wong, M. Wood, M. B. Yaylaoglu, R. C. Young, B. L. Youngstrom, X. F. Yuan, B. Zhang, T. A. Zwingman, and A. R. Jones. Genome-wide atlas of gene expression in the adult mouse brain. *Nature*, 445(7124):168–176, Jan 2007.
- [47] Hui Li, BS Manjunath, and Sanjit K Mitra. A contour-based approach to multisensor image registration. *Image Processing, IEEE Transactions on*, 4(3):320–334, 1995.
- [48] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berke-*

- ley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.
- [49] Georgi K Marinov, Brian A Williams, Kenneth McCue, Gary P Schroth, Jason Gertz, Richard M Myers, and Barbara J Wold. From single-cell to cell-pool transcriptomes: stochasticity in gene expression and rna splicing. *Genome research*, pages gr-161034, 2013.
 - [50] Geoffrey McLachlan and David Peel. *Finite mixture models*. Wiley.com, 2004.
 - [51] G Mendel. Versuche ber pflanzen-hybriden. verh. *Naturforsch. Ver. Brnn*, 4:347, 1866.
 - [52] Pamela J Mitchell and Robert Tjian. Transcriptional regulation in mammalian cells by sequence-specific dna binding proteins. *Science*, 245(4916):371–378, 1989.
 - [53] Ryan D Morin, Matthew Bainbridge, Anthony Fejes, Martin Hirst, Martin Krzywinski, Trevor J Pugh, Helen McDonald, Richard Varhol, Steven JM Jones, and Marco A Marra. Profiling the hela s3 transcriptome using randomly primed cda and massively parallel short-read sequencing. *Biotechniques*, 45(1):81, 2008.
 - [54] Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by rna-seq. *Nature methods*, 5(7):621–628, 2008.
 - [55] PM Nederlof, D Robinson, R Abuknesha, J Wiegant, AHN Hopman, HJ Tanke, and AK Raap. Three-color fluorescence in situ hybridization for the simultaneous detection of multiple nucleic acid sequences. *Cytometry*, 10(1):20–27, 1989.
 - [56] Claus Nielsen. Trochophora larvae: Cell-lineages, ciliary bands, and body regions. 1. annelida and mollusca. *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution*, 302(1): 35–68, 2004.
 - [57] Fatih Ozsolak and Patrice M Milos. Rna sequencing: advances, challenges and opportunities. *Nature Reviews Genetics*, 12(2):87–98, 2010.

- [58] Mary Lou Pardue and Joseph G Gall. Molecular hybridization of radioactive dna to the dna of cytological preparations. *Proceedings of the National Academy of Sciences*, 64(2):600–604, 1969.
- [59] G. A. Pavlopoulos, S. I. O’Donoghue, V. P. Satagopam, T. G. Soldatos, E. Pafilis, and R. Schneider. Arena3D: visualization of biological networks in 3D. *BMC Syst Biol*, 2:104, 2008.
- [60] Hanchuan Peng, Zongcai Ruan, Fuhui Long, Julie H Simpson, and Eugene W Myers. V3d enables real-time 3d visualization and quantitative analysis of large-scale biological image data sets. *Nature Biotechnology*, 28(4):348–353, 2010. URL <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2857929&tool=pmcentrez&rendertype=abstract>.
- [61] Jean-Baptiste Pettit and John C Marioni. bioweb3d: an online webgl 3d data visualisation tool. *BMC bioinformatics*, 14(1):185, 2013.
- [62] D Pinkel, J Landegent, C Collins, J Fuscoe, R Segraves, J Lucas, and J Gray. Fluorescence in situ hybridization with human chromosome-specific libraries: detection of trisomy 21 and translocations of chromosome 4. *Proceedings of the National Academy of Sciences*, 85(23):9138–9142, 1988.
- [63] Lars K Poulsen, Gwyn Ballard, and David A Stahl. Use of rna fluorescence in situ hybridization for measuring the activity of single cells in young and established biofilms. *Applied and Environmental Microbiology*, 59(5):1354–1360, 1993.
- [64] Daniel Ramsköld, Shujun Luo, Yu-Chieh Wang, Robin Li, Qiaolin Deng, Omid R Faridani, Gregory A Daniels, Irina Khrebukova, Jeanne F Loring, Louise C Laurent, et al. Full-length mrna-seq from single-cell levels of rna and individual circulating tumor cells. *Nature biotechnology*, 30(8):777–782, 2012.
- [65] Greg W Rouse. Trochophore concepts: ciliary bands and the evolution of larvae in spiralian metazoa. *Biological Journal of the Linnean Society*, 66(4):411–464, 1999.
- [66] O. Rubel, G. H. Weber, M. Y. Huang, E. W. Bethel, M. D. Biggin, C. C. Fowlkes, C. L. Luengo Hendriks, S. V. Keranen, M. B. Eisen, D. W. Knowles, J. Malik, H. Hagen, and B. Hamann. Integrating

- data clustering and visualization for the analysis of 3D gene expression data. *IEEE/ACM Trans Comput Biol Bioinform*, 7(1):64–79, 2010.
- [67] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13:2498–504, 2003.
- [68] Kazuo Shinozaki, Kazuko Yamaguchi-Shinozaki, and Motoaki Seki. Regulatory network of gene expression in the drought and cold stress responses. *Current opinion in plant biology*, 6(5):410–417, 2003.
- [69] Edwin Mellor Southern. Detection of specific sequences among dna fragments separated by gel electrophoresis. *Journal of molecular biology*, 98(3):503–517, 1975.
- [70] Anders Ståhlberg, Vendula Rusnakova, and Mikael Kubista. The added value of single-cell gene expression profiling. *Briefings in functional genomics*, 12(2):81–89, 2013.
- [71] Detlev Stalling, Malte Westerhoff, Hans-Christian Hege, et al. Amira: A highly interactive system for visual data analysis. *The visualization handbook*, 38:749–67, 2005.
- [72] Badri Narayan Subudhi, Francesca Bovolo, Ashish Ghosh, and Lorenzo Bruzzone. Spatio-contextual fuzzy clustering with markov random field model for change detection in remotely sensed images. *Optics & Laser Technology*, 57:284–292, 2014.
- [73] RR SWIGER and JD TUCKER. Fluorescence in situ hybridization: A brief review. *Environmental and molecular mutagenesis*, 27(4):245–254, 1996.
- [74] Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382, 2009.
- [75] Diethard Tautz and Christine Pfeifle. A non-radioactive in situ hybridization method for the localization of specific rnas in drosophila embryos reveals translational control of the segmentation gene hunchback. *Chromosoma*, 98(2):81–85, 1989.

- [76] Kristin Tessmar-Raible and Detlev Arendt. Emerging systems: between vertebrates and arthropods, the lophotrochozoa. *Current opinion in genetics & development*, 13(4):331–340, 2003.
- [77] Kristin Tessmar-Raible, Florian Raible, Foteini Christodoulou, Keren Guy, Martina Rembold, Harald Hausen, and Detlev Arendt. Conserved sensory-neurosecretory cell types in annelid and fish forebrain: insights into hypothalamus evolution. *Cell*, 129(7):1389–1400, 2007.
- [78] Raju Tomer, Alexandru S Denes, Kristin Tessmar-Raible, and Detlev Arendt. Profiling by image registration reveals common origin of annelid mushroom bodies and vertebrate pallium. *Cell*, 142(5):800–809, 2010.
- [79] Qi Wang, Qun Liang, and Xiuqing Zhang. 3d genome tuner: Compare multiple circular genomes in a 3d context. *Genomics Proteomics Bioinformatics*, 7(3):143–146, 2009.
- [80] Zhong Wang, Mark Gerstein, and Michael Snyder. Rna-seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, 2009.
- [81] Erik Wilde. Putting things to rest. *Transport*, 15(November):567–583, 2007.
- [82] Fa-Yueh Wu. The potts model. *Reviews of modern physics*, 54(1):235, 1982.
- [83] Liying Yan, Mingyu Yang, Hongshan Guo, Lu Yang, Jun Wu, Rong Li, Ping Liu, Ying Lian, Xiaoying Zheng, Jie Yan, et al. Single-cell rna-seq profiling of human preimplantation embryos and embryonic stem cells. *Nature structural & molecular biology*, 2013.
- [84] Alan L Yuille. Generalized deformable models, statistical physics, and matching problems. *Neural Computation*, 2(1):1–24, 1990.
- [85] Erich Zeeck, Tilman Harder, and Manfred Beckmann. Uric acid: the sperm-release pheromone of the marine polychaete platynereis dumerilii. *Journal of Chemical Ecology*, 24(1):13–22, 1998.
- [86] Hua Zhang, Wenzhong Shi, Yunjia Wang, Ming Hao, and Zelang Miao. Spatial-attraction-based markov random field approach

for classification of high spatial resolution multispectral imagery. *Geoscience and Remote Sensing Letters, IEEE*, 11(2):489–493, 2014.

- [87] Jun Zhang. The mean field theory in em procedures for markov random fields. *Signal Processing, IEEE Transactions on*, 40(10):2570–2583, 1992.

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography "*The Elements of Typographic Style*". `classicthesis` is available for both L^AT_EX and L^yX:

<http://code.google.com/p/classicthesis/>

Happy users of `classicthesis` usually send a real postcard to the author, a collection of postcards received so far is featured here:

<http://postcards.miede.de/>

DECLARATION

This thesis:

- is my own work and contains nothing which is the outcome of work done in collaboration with others, except where specified in the text;
- is not substantially the same as any that I have submitted for a degree or diploma or other qualification at any other university; and
- does not exceed the prescribed limit of 60,000 words.

Cambridge, 2014

Jean-Baptiste Olivier
Georges Pettit