# MASTERARBEIT / MASTER’S THESIS

Titel der Masterarbeit / Title of the Master‘s Thesis

„Dimensional reduction through metric pseudotime – interfering developmental processes from single cell RNA sequence data samples “

verfasst von / submitted by

Christiane Puchhammer, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2020/ Vienna, 2020

| | |
|---|---|
| Studienkennzahl lt. Studienblatt /<br>degree programme code as it appears on<br>the student record sheet: | A 066 876 |
| Studienrichtung lt. Studienblatt /<br>degree programme as it appears on<br>the student record sheet: | Masterstudium Physik |
| Betreut von / Supervisor: | Univ.- Prof. Mag. DDr. Stefan Thurner |

# Contents

# Abstract

In this work we analyse a set of single cell RNA sequencing data that describes the development of the adrenal medulla in the genesis of mouse embryos. We used methods in the overlap of physics and complex science to gain new insights into the temporal structure of gene regulation during such developmental processes if possible, but mainly to test whether RNA sequencing data is sufficiently informative to fit an ODE model to the data. We assume that a developmental process can be described by a linearized differential equation

$$\dot{x} = J + Ax$$

with a non-linear constraint forcing the species abundances $x$ to remain positive. To fit the process in this differential equation we need to approximate $\dot{x}$ in order to gain $J$ and $A$ that contain the information of the regulatory dynamics in developmental time. We employ two different methods to approximate $\dot{x}$ and compare their results for the model. In the first method (i) we use data projected to a future state of the process to make the approximation. In the second method (ii) we construct a so-called pseudotime description for the developmental process with the use of dimensional reduction to gain $\dot{x}$. A steepest descent method is then used to estimate the parameters $J$ and $A$. Genes with non-random regulatory characteristics are then identified from those parameters. In this thesis, we provide a basis of knowledge of the reviewed process as well as an investigation of the methods to do so, that should be further researched and improved in future works.

In dieser Arbeit analysieren wir ein Set aus Daten, das mithilfe von Einzelzell-RNA-Seqenzierung erhoben wurde und die Entwicklung des Nebennierenmarks bei der Entstehung von Mausembryonen beschreibt. Wir verwenden Methoden die sowohl in der Physik als auch in den Komplexitätswissenschaften verwendet werden, um, wenn möglich, neue Erkenntnisse über die zeitliche Struktur der Genregulation in diesem Entwicklungsprozess zu erhalten. Hauptsächlich aber testen wir dabei, ob RNA-Sequenzierungsdaten aussagekräftig genug sind, um ein Differentialgleichungssystem mit den Daten zu fitten. Wir nehmen an, dass ein Entwicklungsprozess durch eine linearisierte Differentialgleichung

$$\dot{x} = J + Ax$$

mit der nichtlinearen Bedingung, dass die Menge einer Spezies $x$ positiv sein muss, beschrieben werden kann. Um den Prozess mit dieser Differentialgleichung zu fitten, müssen wir erst $\dot{x}$ schätzen, um die Parameter $J$ und $A$ berechenen zu können, welche Information über die regulativen Dynamiken in der zeitlichen Entwicklung enthalten. Wir verwenden zwei verschiedene Methoden, um $\dot{x}$ zu nähern und vergleichen deren Ergebnisse für unser Modell. In der ersten Methode (i) verwenden wir Daten, die mathematisch in den zukünftigen Zustand der Zelle projeziert werden, um die Approximation zu machen. In der zweiten Mehtode (ii) konstruieren wir eine sogenannte Pseudozeit zur Beschreibung des Entwicklungsprozesses, indem wir eine Dimensionsreduktion vornehmen. Mit deren Hilfe können wir $\dot{x}$ approximieren. Ein konjugiertes Gradientenverfahren wird verwendet, um die Parameter $J$ und $A$ zu schätzen. Gene mit nichtzufälligen regulatorischen Eigenschaften werden dann anhand dieser Parameter identifiziert.

# Acknowledgement

First of all I would like to thank my supervisors Univ.-Prof. Mag. DDr. Stefan Thurner and Assoc. Prof. Priv.-Doz. Mag. Dr. Rudolf Hanel without whom this thesis would not have been possible. I would also like to thank Univ. Prof. Dr. Igor Adameyko, Artem Artemov, BSc and the rest of their research team for providing the data and support in biological questions. Furthermore, I want to thank my colleagues in the Medical University of Vienna and the Complexity Science Hub for helping with day to day problems as well as sharing their valuable input with me.

Finally, I want to thank my parents, my grandparents, my sister and my friends for their emotional and moral support and encouragement throughout my years of study and in the process of writing this thesis. They turned this stressful time into a wonderful one and this thesis would not have been possible without them.

# Chapter 1

# Introduction

A fundamental concern in *developmental biology* is to understand how cells evolve in a multi cellular organism and differentiate. To take our understanding of developmental processes and the mathematical framework suited for analyzing state of the art data of such processes a step further, we will analyze one specific step in the development of the mouse embryo on the basis of a mathematical model from the field of *dynamical systems* that we utilize for analyzing single cell RNA sequencing data. We are specifically looking at the development of the adrenal medulla that mainly originates from Schwann cell precursors. The course of this process can be seen in figure 1.1. [1]

Before we start our analysis it has to be said that cells of a multi-cellular organism can be identified as different types morphologically (information we will not use) and by the kind and amount of genes they express. What we therefore do in this master thesis is to look at gene expression profiles of samples of a few hundred cells profiled with *single cell RNA sequencing* to gain new insights into the process of one specific cell differentiation process in the development of mouse embryos. At the moment other methods to reveal lineage relationships like linage tracking have limited power. [2]

We are dealing with a dynamic cell process. There have been fundamental limits in the analysis of dynamic processes so far, as the expression profiling methods, that can capture a extremely detailed snapshot of the cell in its current state destroy the cell in the process of analysis. Therefore it is not possible to measure how the gene expressions of a cell change over time. [3] It is therefore important to know that during the dynamic processes of differentiation cells move through a high-dimensional *RNA expression space* determined collectively by all genes. In mathematical terms, we can therefore say that the physiological expression rates of genes belonging to cells live on a high dimensional manifold. The distribution of cells in the *expression space* allows for a partial reconstruction of this manifold. In the context of embryo-genesis cells move through expression space by changing their expression profiles and thereby sample the space along a low dimensional sub-manifold that is characterising the developmental process.

The differentiation events that happen during the development of a mouse embryo take place on timescales of hours to days. Those timescales are compareable to the duration of mRNA life cycles. A relatively novel and promising method to examine the dynamics of cellular states in biology is the method of single cell RNA sequencing. Single cell sequencing only provides data of the cellular states at one single

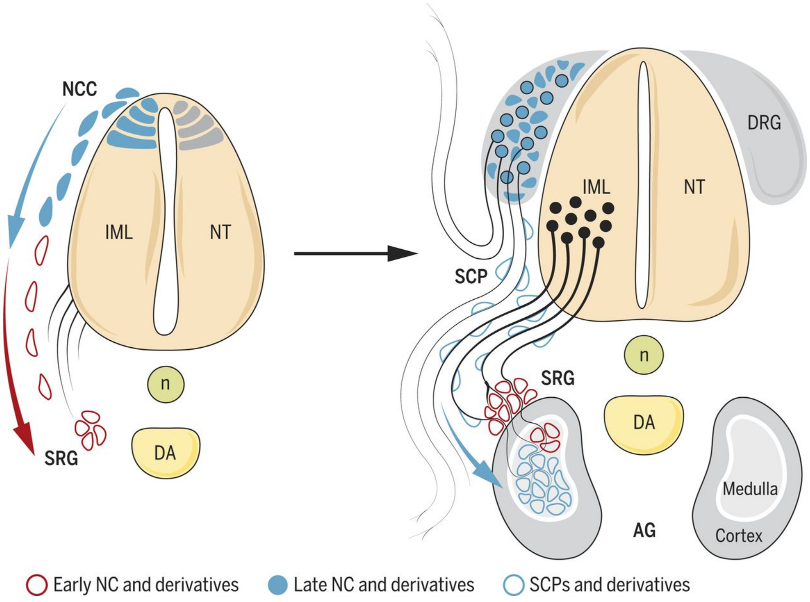


Figure 1.1: Detailled schematic representation of the development of the adrenal medulla from Schwann cell precursors (SCP).[1]

point in developmental time , i.e. in the developmental process the sample of cells is undergoing. [4] However, the cells in one sample can be expected to be *smeared out* in terms of *developmental time*, just like a group of pupils doing homework together will be found in various stages of having completed their homework when observed at the same point in time, each of them going through their homework in their own time.

Hence we need an approach suitable for reconstructing the process for such data in developmental time, sometimes also called pseudotime. While pseudotime is a relatively new concept it has already been utilized by a set of trajectory inference methods. [5][6][7] As all of these methods are still struggling with the amount of uncertainty one faces in constructing low dimensional developmental manifolds, we attempt another take on the pseudotime approach using a single cell RNA sequencing data set capturing the development of the adrenal medulla from Schwann cell precursors. The data-set has been provided by Adameyko Lab [8][9]. It therefore is useful to look at mathematical possibilities to capture dynamic processes in cells. Mouse cells are (just as human cells) eucaryotic cells meaning they have a nucleus and organelles. More importantly, it are eucaryotic cells of a multicellular organism, which can exist in various states of differentiation. This is important since the production of new proteins follow certain patterns, which depend on the cell type in characteristic ways. When a gene gets expressed the DNA is transcribed into pre-mRNA (precursor messenger RNA). The newly synthesized pre-mRNA still contains both introns (parts of RNA that will be cut out in the next step) and exons (parts of the RNA that are kept). In the next step the pre-mRNA is spliced (the introns are cut out and the exons are glued together) and thereby turned into mature mRNA. The mature mRNA is then used to translate the genetic information into protein which typically gets degraded by proteasomes after usage. Degradation rates however may vary strongly for different species of RNA. [4][10] The whole process can be seen in figure 1.2 (It may also be noted that RNA concentration can only give us an approximate idea on the actual protein production rates, which depend on the

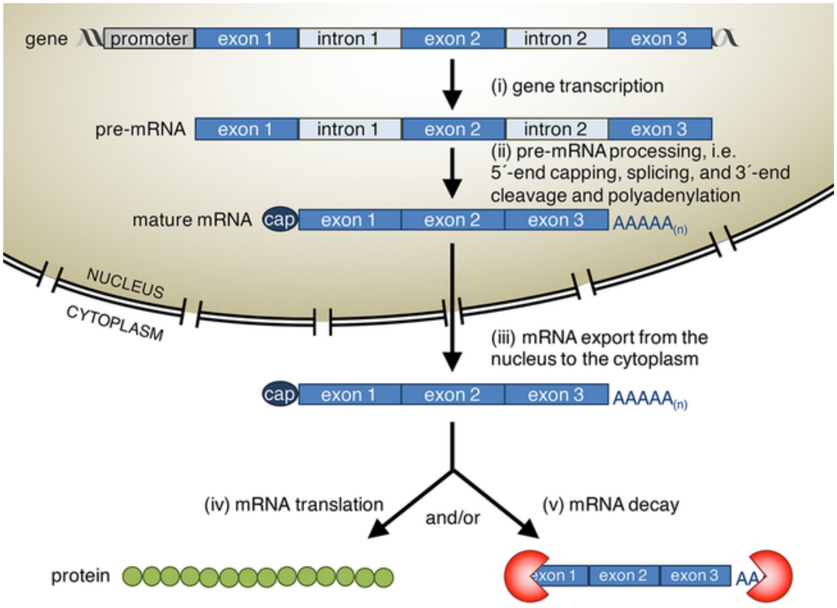


Figure 1.2: Detailed schematic overview of the transcription of DNA in an eucaryotic cell. [11]

number on ribosomes that simultanously translate the RNA molecules.)

This process takes place to some extent all throughout a cell cycle, meaning that each individual cell contains certain amounts of pre-mRNA, mature mRNA and partially or completely degraded mRNA at the same time. Previous research has showed that one can not only gain regulation information from the measurement of mature RNA, but from capturing pre-mRNA as well. One can use the change of the abundances of the mRNA stages in order to infer RNA turn over rates from the ratios of mRNA molecules at different stages in the RNA life-cycle which are distinguishable by sequencing. Molecules of all three stages can be used to measure the *transcriptional velocity*, which can be defined as the rate of change of the abundances of mRNA molecules in the cell. [4] Knowing this, it is possible to calculate the so-called *RNA-velocity* that then can be used to calculate the state a cell should be in in the future (details are discussed below).

Here our main interest is to pinpoint, list and discuss the challenges that need to be addressed and eventually solved for this kind of data to reveal its secrets.

Note that construction of a *developmental time* is a dimensional reduction process. One of the major findings of this work is that in high dimensional RNA space *noise* contributes substancially to metric distances in $\Re^{N_{genes}}$. We discuss strategies to deal with those contributions in low dimensional embeddings of the data.

# Chapter 2

# Theory

For dealing with single cell RNA sequencing data we require some theoretical basis, which we will briefly discuss in the following.

## 2.1 Metrics

In mathematics a metric is a function that defines the distance between each pair of elements of a set. (A set with a metric is a metric space.) A metric is defined as a function on a set $X$ with

$$d : X \times X \to [0, \infty)$$

where $[0, \infty)$ is a set of non-negative real numbers and for all $x, y, z \in X$ the following conditions are satisfied:

1. $d(x, y) = 0 \Leftrightarrow x = y$
2. $d(x, y) = d(y, x)$ (Symmetry)
3. $d(x, y) \leq d(x, z) + d(z, y)$ (Triangle inequation)

[12]. From these conditions also automatically follows that $d(x, y) \geq 0$. We chose to look at two metrics that are most suited for our problem, namely the L1 metric and the Jaccard metric.

### 2.1.1 L1 Metric

The *L1 metric*, sometimes also called *taxicab distance* is defined as

$$d_1(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_1 = \sum_i |x_i - y_i| \tag{2.1}$$

[13]

### 2.1.2 Jaccard metric

We define the Jaccard metric or Jaccard distance using the so called Jaccard index. The *Jaccard index* is a measure that is used to define the similarity of sets. It is defined as

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{2.2}$$

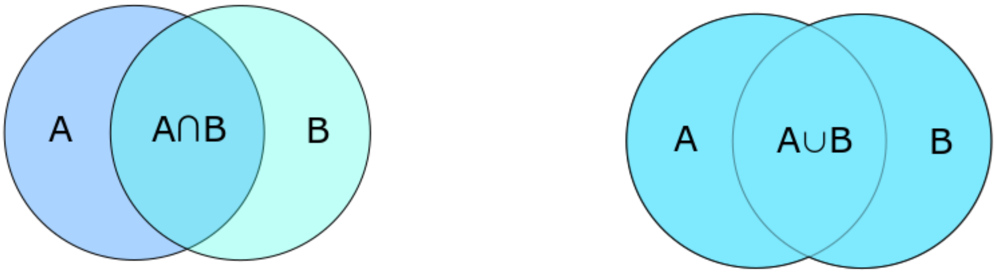
[14] with

$$0 < J(A, B) < 1 \tag{2.3}$$

For two finite sample sets A and B. The Jaccard-Distance then is

$$d_J(A, B) = 1 - J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|} \tag{2.4}$$

and as equation 2.4 satisfies all conditions for a metric the Jaccard distance is in fact a metric we call the *Jaccard metric* [15].



(a) (b)

Figure 2.1: The (a) intersection and (b) union of two overlapping sets A and B

## 2.2 Differential equation model

To capture the dynamic process, we decided to use a differential equation model, that follows a linearized equation

$$\dot{x}_i = J_i + \sum_j A_{ij} x_j \qquad\qquad x_i \geq 0 \tag{2.5}$$

with parameter models being the zero and first order coefficients $J_i$ and $A_{ij}$. Note that $A_{ij}$ encodes "catalytic" interactions between entities $i$ and $j$. Also the constraint $x_i \geq 0$ is necessary, to mathematically include the fact that concentrations cannot be smaller than 0. Data points are given by pairs $(\dot{x}, x)$ where $x_i$ corresponds to the RNA counts of type $i$ and is directly provided by RNA sequencing data. How to estimate $\dot{x}_i$ however, is not an entirely trivial task. Possibilities to do so will be discussed below. After a successful estimation of $\dot{x}_i$ we can minimize the error with respect to the model parameters $J_i$ and $A_{ij}$ by applying the method of steepest descent.

## 2.3 Steepest Descent

The method of steepest descent is an iterative method suited to solve a large sparse non-linear system of equations. This system of equations can generally be describes as

$$y_i = \sum_j A_{ij} x_j \tag{2.6}$$

where $x$ and $y$ are vectors and $A$ is the (sparse) coefficient matrix. In order for this method to work $A$ also needs to be symmetric and positive definite.
Given that the requirements are met, the goal is to minimize a functional $f(z)$. This is done by calculating the negative gradient $\nabla f(z)$ (also called *residuum* $r$), which in the general case is given by

$$r = -\nabla f(z) = -\left(\sum_j A_{ij} z_j - y_i\right) \tag{2.7}$$

where $z$ is an arbitrary vector defined as

$$z = x + p \tag{2.8}$$

when $x$ is the actual minimum of the functional. We of course move into the negative direction of the gradient, because this is the direction where the functional decreases the fastest. We do this until we find a minimum. Given that the starting vector $z$ is arbitrary, this procedure has to be repeated multiple times. Each time the local minimum of the last approximation of $z^{(n)}$ serves as next starting vector $z^{(n+1)}$. This iteration is repeated until $f$ hits its global minimum. [16]

The equation we want to approximate in this work is equation 2.5, which is an ordinary differential equation (ODE). We define a function $g$ to summarize the model parameters.This function is not to be confused with a functional and is defined as

$$g_i(J, A, x) \equiv J_i + A_{ij} x_j \tag{2.9}$$

Looking at equation 2.5 this also provides us with the relation

$$\dot{x}_i = g_i(J, A, x) \tag{2.10}$$

While $x$ is given by the data and $A$ and $J$ will be fitted in the course of the steepest descent, $\dot{x}$ has to be approximated from the data.

$$\dot{x}_i \sim \frac{\Delta x_i}{\Delta t} = \frac{x_i \overbrace{(t_k + \Delta t)}^{t_{k+1}} - x_i(t_k)}{\Delta t} \tag{2.11}$$

From this knowledge, we can now build up the error functional for the steepest descent that shall be called $\sigma^2$ here:

$$\sigma^2(J, A) = \sum_t \sum_i \left(\frac{\Delta x_i}{\Delta t} - g_i(J, A, x)\right)^2 \tag{2.12}$$

$\sigma^2$ is then minimized with respect to $J$ and $A$. In the following we apply this method to (a) fitting a differential equation model to RNA sequencing data and (b) to geometric dimensional embedding.
It has to be noted that one can make the steepest descent noisy. This means adding noise in the beginning and gradually reducing it in the process until it is turned off. This can be helpful to get out of possible minima at the start.

## 2.4 RNA velocity

As stated in the introduction the *RNA velocity* can be used to predict the gene expression state of a cell in the future. We will use this information to estimate $\dot{x}$ in the differential equation model, therefore it is important to look at how the future data set is obtained.

The time dependent relationship between precursor and mature mRNA abundance has been quantified with a simple model for transcriptional dynamics. RNA velocity, which has been defined as the first derivative of the mature mRNA abundance. The mature mRNA abundance is determined by the balance between the production of spliced from unspliced messenger RNA as well as the rate of mRNA degradation. Assuming steady state the cell is maintaining a constant RNA abundance and therefore the RNA velocity is zero. Hence the relationship between spliced mRNA (s) and unspliced mRNA (u) can be described with a fixed-slope

$$u = \gamma s \tag{2.13}$$

where $\gamma$ is the degradation rate. The balance of spliced and unspliced mRNA can then be used as an indicator for the future state of mature mRNA abundance. We can write down the rate equations for a single gene describing how the expected number of unspliced mRNA molecules $u$ and spliced molecules $s$ evolve over time:

$$\frac{du}{dt} = \alpha(t) - \beta(t)u(t) \tag{2.14}$$

$$\frac{ds}{dt} = \beta(t)u(t) - \gamma(t)s(t) \tag{2.15}$$

with $\alpha(t)$ being the time-dependent rate of transcription, $\beta(t)$ the rate of splicing and $\gamma(t)$ the rate of degradation. By assuming a constant rate of splicing leading to constant rates $\alpha(t) = \alpha$, $\gamma(t) = \gamma$ and setting $\beta(t) = 1$, the equations 2.15 simplify to

$$\frac{du}{dt} = \alpha - u(t) \tag{2.16}$$

$$\frac{ds}{dt} = u(t) - \gamma s(t) \tag{2.17}$$

The solution of the rate equations is then given by

$$u(t) = \alpha(1 - e^{-t}) + u_0 e^{-t} \tag{2.18}$$

$$s(t) = \frac{e^{-t}[e^{t(1+\gamma)}\alpha(\gamma - 1) + e^{t\gamma}(u_0 - \alpha)\gamma + e^{t}(\alpha - \gamma(s_0 + u_0 + s_0\gamma))]}{\gamma(\gamma - 1)} \tag{2.19}$$

with initial conditions $u(0) = u_0$ and $s(0) = s_0$. This solution can be used to extrapolate the mRNA abundance $s$ to a future time-point.

The normalized degradation rate $\gamma$ varies among RNA molecule types (and therefore needs to be estimated for each gene). In a steady state population $\frac{ds}{dt} = 0$, $\gamma$ can be determined as the ratio of unspliced mRNA molecules to spliced ones:

$$\gamma = \frac{u}{s} \tag{2.20}$$

For non-steady state populations there are two possible models to determine *gamma*:

**Model 1: Constant velocity assumption**
In this model it is assumed that the spliced molecules change with a constant rate $\frac{ds}{dt} = v$. Then the extrapolation is trivial since

$$s(t) = s_0 + vt \tag{2.21}$$

In case of a down-regulating gene $v < 0$ clipping the values at zero is required.

**Model 2: Constant unspliced molecules assumption**
Here it is assumed that the number of unspliced molecules stay constant $u(t) = u_0$. The problem then reduces to

$$\frac{ds}{dt} = u_0 - \gamma s(t) \tag{2.22}$$

and the solution becomes

$$s(t) = s_0 e^{-\gamma t} + \frac{u_0}{\gamma}(1 - e^{-\gamma t}) \tag{2.23}$$

Both of these models were used to estimate a projection of a data set into the future, capturing the expression rates of the genes of the same cells at a future point in time. [4]

The idea of utilizing information on the ratios of precursor RNA and spliced RNA abundance is an extremely interesting approach. However, it is not clear at this point to which extent cooperative effects, between RNA species and their degradation mechanisms, as we attempt to do in our analysis below, would modify such estimates.

## 2.5 Dimensional reduction

Another way to obtain a time order that makes it possible to approximate $\dot{x}$ is to look at the measured data and sort it by the small differences in the developmental time of the cells. A well suited way to do so is by the application of a dimensional reduction method. *Dimensional reduction* is the process of reducing the number of random variables under consideration by obtaining a set of principal variables [17]. (*Principal variables* in this context are reduced variables, that in the context of principal components are called principal variables.) There are two approaches to this, *feature selection*, where a subset of the features are chosen that are relevant for the model and *feature extraction* where from the original data set variables are derived that capture the relevant information of the input data [18]. We will be looking at the latter one.

There is a family of widely used techniques to perform the dimensional reduction, that more or less lead to the same results the main one being the *Principal component analysis* (PCA). The PCA performs a linear mapping of the data to a lower-dimensional space in such a way that the variance of the data in the low-dimensional

representation is maximized. [19]
Here however, we will employ "geometric" embedding methods based on the minimization of the metric mismatch made in an embedding map $\pi : \Re^N \rightarrow \Re^d$ $d = 1, 2, 3, ...$

# Chapter 3

# Methods

## 3.1 Regulatory Dynamics

In this master thesis two sets of data on the cells of mouse embryos in development have been analyzed. They each contain 384 cells and the expressions of 1518 of their genes in the process. One data set, which we will call *original* or *base* data set, was measured by single cell RNA sequencing. The other data set is a projection of the first original data set into the future. It was calculated using the RNA velocity of the first data set as explained above. We will attempt to use a differential equation model to capture regulatory dynamics in two different ways. We therefore first assume that a regulatory process can be described by a linearized system of ordinary differential equations

$$\dot{x}_i = J_i + \sum_j A_{ij} x_j + noise \tag{3.1}$$

with a non-linear constraint forcing the species abundances $x$ to remain positive. Then we can use the data to infer a time ordering approximating the parameters $x$ and $\dot{x}$. This can be done in two ways that will be explained below. In the next step we will try to find the parameters $J$ and $A$ in equation 3.1 by minimizing the error using the method of steepest decent for both time approximations and compare the results.

## 3.2 Projected RNA velocity

For the first attempt to fit the differential equation 3.1 we use both of the data sets, the original and the projected one, to estimate the parameters $x$ and $\dot{x}$. As $x$ is the state of the gene expression values at the current moment, we use the original data set as it is, to approximate it. The approximation of the developmental time derivative $\dot{x}$ is more complicated. We approximate the derivation of $x$ written as $\dot{x} = \frac{\Delta x}{\Delta \tau}$ using both of the data sets with

$$\dot{x} = x_{projected} - x_{original} \tag{3.2}$$

where $x_{projected}$ is the matrix with the values of $x$ from the data set that extrapolates the values of the originally measured data into the future, while $x_{original}$ is the matrix with the originally measured values.

## 3.3 Developmental time order and pseudo-time

A second approach is especially useful to gauge the quality of the results of both methods, because as both methods have never been used in this context before we have a priori no certainty on how well either of the approaches will actually capture regulatory dynamics of the cells. We look at another possibility to obtain an approximation of the temporal order of the cells in the sample that allows us to estimate $\dot{x}$. ($x$ will again be approximated by the original data set again.)
The cells in the sample naturally contain small differences in their developmental time. We now can use this developmental time ordering already included in the original data set to perform this extrapolation of $\dot{x}$.
The cells in the data set are in some order unknown to us that at no point of our analysis is thought to contain relevant information about the process, so we first have to find out the time order of the cells. It is possible to order the cells in time, by looking at how similar their genes expressions are as the gene expressions rates change over time. As this is a gradual process it should be safe to assume that the more similar a gene expression profile of one cell is to another, the closer they are likely to be in time too. One can quantify this relation as the metric distance between two cells. For this reason we specifically calculated the L1 metric and the Jaccard metric for the cells of our data sample, as they serve our purposes the best. The results can be seen in figure 3.1 that shows the distances between the cells and figure 3.2 that plots the normalized distances between cells ordered according to their order in the data sample. Furthermore the histogram in figure 3.3 shows the distribution of the metric distances of the cells to each other for both metrics in comparison. It can be seen that the distances between cells are higher for the Jaccard metric than for the L1 metric. Both metric measures may also capture different aspects of "distance". The Jaccard metric for instance focuses on the similarity or dissimilarity of the support of gene expression, i.e. the number of genes commonly expressed by two cells, disregarding the actual amplitude of the gene expression. This is the reason we use more than one metric. All metrics will have (at least slightly) different outcomes, but when they result in the same or at least a similar time order this means that the resulting time order can be believed in with some certainty.

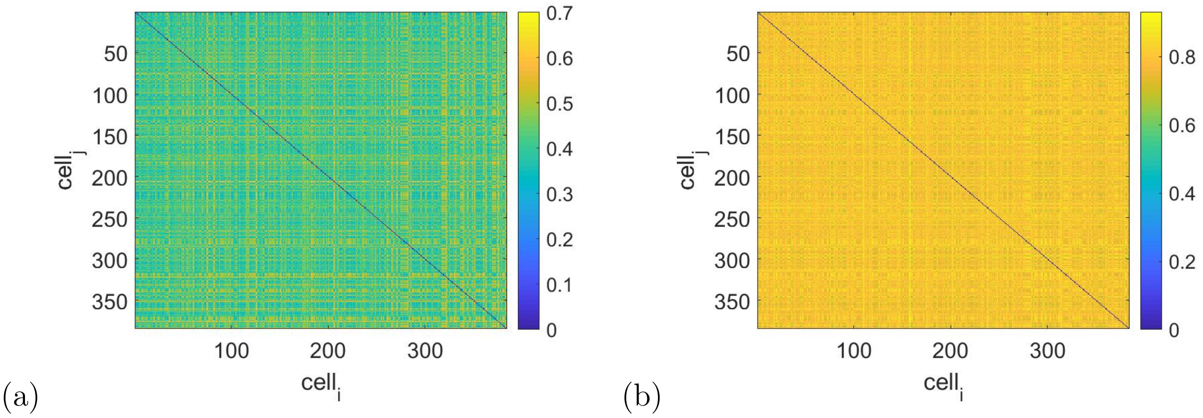


Figure 3.1: Color scaled plot of the L1 metric (a) and Jaccard metric (b), where each entry is colored according to its value.

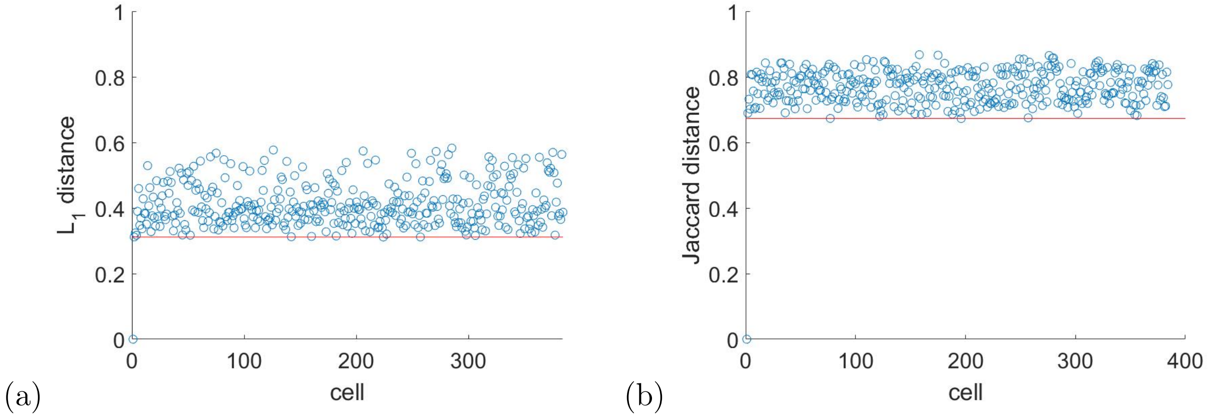


Figure 3.2: This plot shows the normalized distances of the first cell to all other cells in the sample for the L1 metric (a) and the Jaccard metric (b) ordered according to the original data sample. The lower bound is marked with a red line for both.

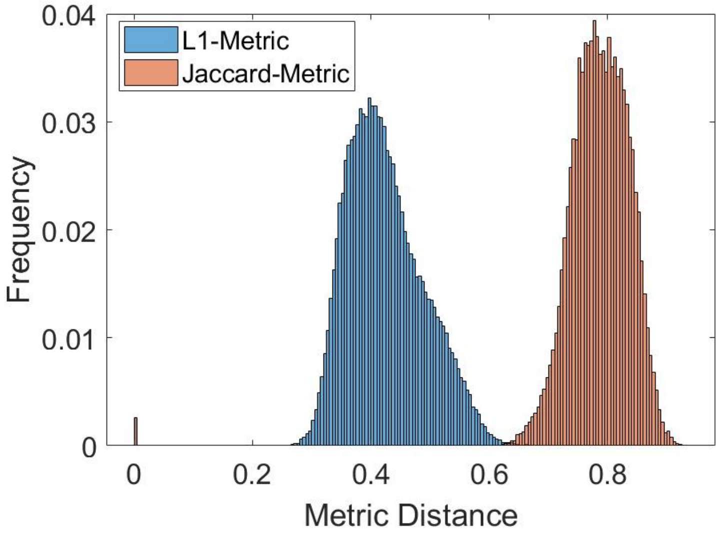


Figure 3.3: Histogram showing the distribution of the metric distances of the L1 metric (blue) as well as the ones of the Jaccard metric (orange)

The Jaccard metric is calculated in two different ways. One calculates the overlap between the amount of cells that are switched on, as well as the ones that are switched off. The results are similar, but have to be calculated seperately and one cannot be gained from the other. (Otherwise there would also be no reason to calculate both of them.) For both versions of the Jaccard metric the metric distances between cells were calculated with three different thresholds (defining zero), namely $10^{-3}$, $10^{-4}$ and $10^{-5}$. This results in a total of six different Jaccard metrics. As they are all very similar, only one of them was plotted in this part for figures 3.1 and 3.2 for the purpose of illustrating the differences to the L1 metric. Later on they will be looked at in more detail and compared to each other.

### 3.3.1 Dimensional reduction

In the next step we use the information about the distances between cells that we gain from the metrics to order the data set according to the inferred developmental time of the cells. The metrics present us with the opportunity to approach the challenges posed by the high dimensionality of the data. The data suggests that we are actually looking at a low dimensional problem in terms of "developmental dimensions". However, in high dimensions noise poses a severe problem for the reconstruction of this low dimensional developmental manifold.

$$d_{\text{high dimensional}} = d_{\text{low dimensional}} + d_{\text{noise}} \tag{3.3}$$

To obtain the low dimensional description of the data we performed dimensional reduction. We used a direct method to reduce the $N$-dimensional data to a one dimensional developmental time order ($\Re^n \rightarrow \Re^1$). The multidimensional data of the metrics gets projected onto a one dimensional axis. The process is depicted as a cartoon in figure 3.4.

The process of ordering follows a number of steps. In the beginning we have to filter the noise in our cell-to-cell distance data in order to get meaningful results in further analyses. The noise contribution to the metric in this scenario is estimated as the minimal non-zero distance between any of the cells, represented by the value

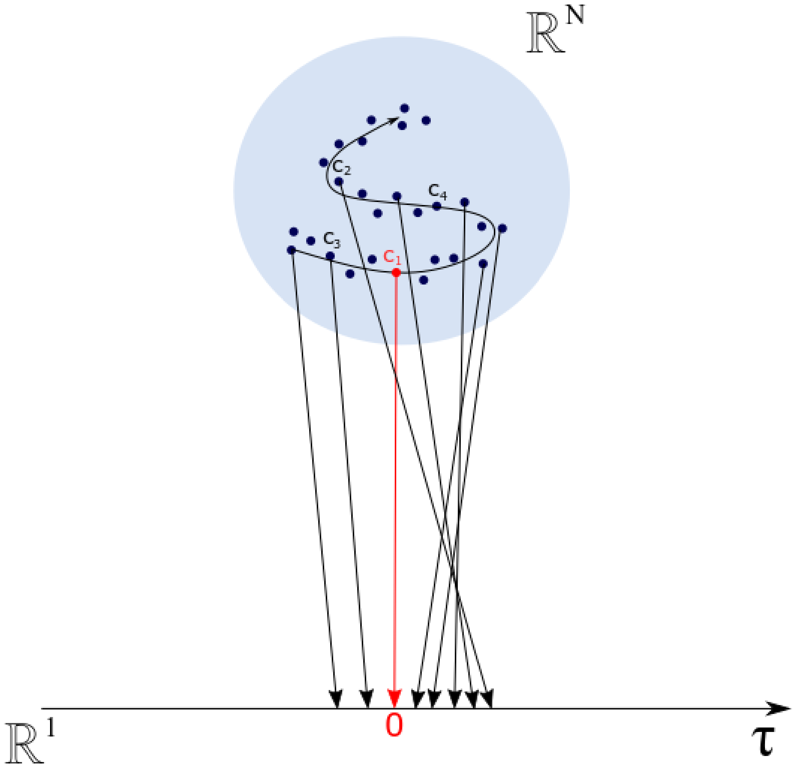


Figure 3.4: Schematic representation of a projection from the N-dimensional gene expression space onto a the one dimensional developmental time axis.

of the lower threshold level (red line) in figure 3.2.
Then we choose a random cell to initialize the ordering process (as we have no idea of the actual order). We can choose freely where on the developmental time axis this first cell is placed (i.e. we can choose the origin of our time axis). We conveniently chose to project the first cell onto zero. In the next step, the second cell (that also can be chosen freely from the rest of the cells) is chosen and placed relative to the first one, according to the metric distance, with a positive value.
Since we have no means of telling the overall orientation of the time-line of the whole process in other words, we do not know whether we are looking at the process from start to finish and or the other way round, we later have to orient differently initialized time order constructions relative to one another. The rest of the cells are then placed relative to the first two cells. Their position on the axis is clear from the distances to the first and the second cell as can be seen in figure 3.5. As the

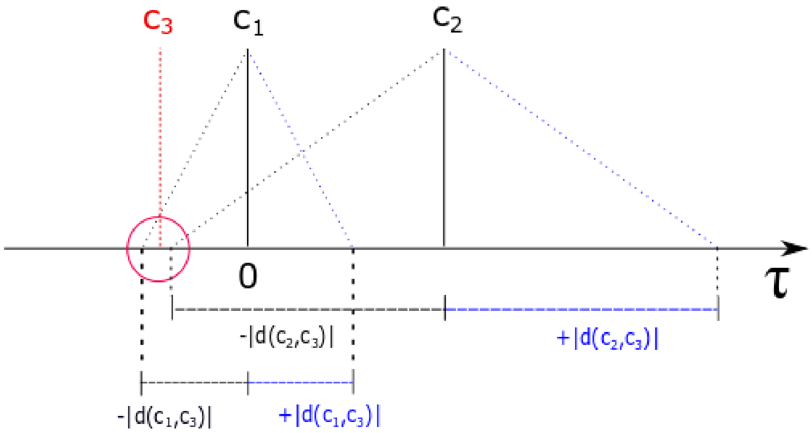


Figure 3.5: Schematic representation of the process of estimating the position of the cells in relation to the first two cells. Cell 1 is set to zero, cell 2 is set in its distance to cell 1 on the right side (positive). It is then looked at the metric distances between cell 1 and cell 3 as well as cell 2 and cell 3. Those are compared and the location of cell 3 in the developmental time ordering is picked as the mean of the distances that are closest to each other.

distances can be assumed to be noisy, we take the average of the positions we would infer with respect to cell 1 and cell 2. This results in a developmental time order for all cells.
This initial time order can iteratively be improved by optimizing the relative positions of the cells to each other. In particular we can calculate the improved position of a cell $m$ in the $(r+1)$'th iteration as the average of the positions for $m$ that we would estimate from the positions of cells $i$ in the previous iteration $r$: The next iteration $(r+1)$ for a cell $m$ is calculated from the previous iteration$(r)$ in two steps:

$$x_m^{(r+1)'} = \frac{1}{N} \sum_{i=1}^{N} \left( x_i^{(r)} + \operatorname{sgn}\left( x_m^{(r)} - x_i^{(r)} \right) g(c_i, c_m) \right) \tag{3.4}$$

$$x_m^{(r+1)} = x_m^{(r+1)'} - x_1^{(r+1)'} \tag{3.5}$$

The first step consists of calculating an intermediate result $(r+1)'$ for cell $m$. This is done by calculating the normalized sum over all cells and their metric distances to cell $m$ while taking the sign on the developmental time axis into consideration. The second step is then to assure that the position of cell $i = 1$ remains tethered to the origin of the time axis $t = 0$. For this $x_1^{(r+1)'}$ is subtracted from the new intermediate cell positions $x_m^{(r+1)}$. This sets the cell that we choose to initialize the time ordering procedure back to $t = 0$. This iteration of x is repeated 100 times.
This is done for all possible choices for the two cells that we require to initialize the time ordering procedure we describe above. In the next step we construct an average time order of the cells and check the robustness of the ordering methods with respect to the possible choices for cell pairs initializing the ordering procedure as well as with respect to the distinct metric measures.

### 3.3.2 Time ordering

We now have $N(N-1)$ possible orders of developmental time corresponding to the distinct possibilities of choosing the two cells initializing the ordering procedure. Before we do any further analyses, we have to remember that for constructing the time order we have decided to position the second cell on the positive side of the timeline. This is a necessary step in creating a time ordering as we have no way of knowing which global orientation (forward or backward) is realized in the actual cell development. However, now we have to take into consideration that this method will not result in a cell ordering of the same orientation for each particular realization of the ordering. Some of the constructed orderings therefore will begin at the start of the developmental process and end at its end, while others will be ordered the other way around.
We cannot know the actual direction of the process. However, for our purposes it is sufficient if all orderings are oriented in the same way. To do this we first choose one particular ordering as reference and orient all other particular orderings accordingly. Hence, we choose the results for the developmental time ordering of the first cell pair $\tau(1,2)$ (with *cell 1* and *cell 2* from the data set) as the reference. We use $\tau(1,2)$ to identify the orientation of a different developmental time order $\tau(x,y)$ (that was calculated using the cell pair *cell x* and *cell y*). In the first step, we calculate the differences in the values of $\tau(1,2)$ and $\tau(x,y)$ for each index in developmental time $i$ and take the sum of the absolute values as stated in equation 3.6.

$$\mu_1(\tau(x,y)) = \sum_{i=1}^{N} |\tau_i(1,2) - \tau_i(x,y)| \tag{3.6}$$

In the second step, we repeat this process but inverse the indices for $\tau(x,y)$ like in equation 3.7.

$$\mu_2(\tau(x,y)) = \sum_{i=1}^{N} |\tau_i(1,2) - \tau_{(N+1)-i}(x,y)| \tag{3.7}$$

Finally, we compare the results of $\mu_1$ and $\mu_2$. As we are looking for the time order that is closer to $\tau(1,2)$ and hence differs less from it, we choose the smaller $\mu$ as

result. We then change the orientation of $\tau(x, y)$ accordingly.
The developmental time ordering procedure described above gives us slightly different results for each cell pair, because we have to deal with noise that affects the ordering. Therefore in the following all the orderings of the developmental time will be used to find the most likely one.

### 3.3.3 Further analysis

Now we have globally oriented the particular time orders we have obtained as described above with respect to one reference order. We continue by constructing the expected time order. Furthermore we check the robustness of this ordering and filter genes from the data which show no improvement in the smoothness of its evolution with respect to developmental time.

**Probability Distribution:**

In order to construct the expected time order of the cells in the sample the first thing to be calculated is the probability distribution cells located in developmental time, i.e. we measure how often a cell $i$ is put into a position $j$.

$$P_i(\tau) = \frac{1}{Z} \sum_{(x,y)} \chi(\tau = \tau_i(x, y)) \quad \forall i \tag{3.8}$$

$$Z = \text{number of pairs } (x, y)$$

$$\chi = \begin{cases} 1 & \text{if cell i on } \tau \\ 0 & \text{else} \end{cases} \tag{3.9}$$

Using formula 3.8 we plotted the probability (histograms) of the cells to be at a certain place in the developmental time in figure 3.6 (b) as well as the image of the probability matrix in figure 3.6 (a). In figure 3.6 (b) we show probability distribution of five random cells over the expected developmental time order, as computed in equation 3.8. In figure 3.6 (a) we plot the same probabilistic information (z-axis) for all cells in where we locate cells in their given order of the original data-set (x-axis) over the expected time order (y-axis). In figure 3.6 (a) it can be seen that the different orderings mainly agree on the place a cell should be put in the overall order. Furthermore, figure 3.6 (b) shows the probability of five specific cells form the sample, namely the cells have the indices 10, 120, 200, 275 and 380 in the original data. We will use the same five cells to exemplify the properties of other measures we look into further below. In particular it can be seen that cell 380 is with a high probability in a certain place in the developmental order, cell 275 and cell 10 also are clearly placed in a certain spot even though with a smaller probability, while the rest of the five cells in the plot do not seem to have a clearly indicated position in the developmental time order. In general figure 3.6 (a) indicates that there are a lot of cells in the beginning and the end of the data set with a well defined temporal location, while in the middle section of the original data set there are cells that can not clearly be located in the developmental time order. This could, for instance, indicate that a one dimensional developmental time manifold is insufficient and the developmental space has a higher dimension. Similarly, it also could mean that those cells less well located in fact belong, in some coarse grained sense, to a single time point, the starting or end point of a process, where the cell expression variability may be larger than in transitional cells, which display a smaller expression variability and match a well defined time order. Other measures below paint a similar picture.

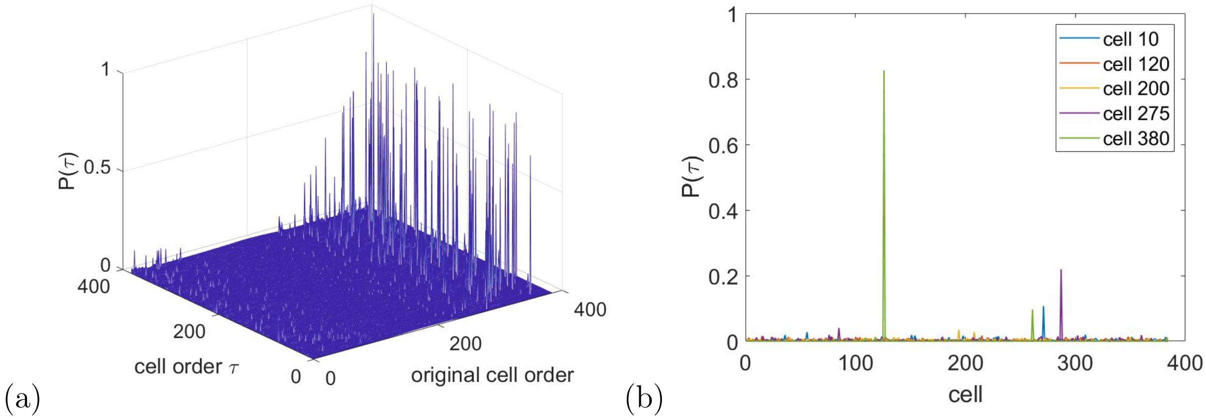


Figure 3.6: Probablilty distribution of a cell $i$ of the original order for being cell $j$ in $\tau$ in a histogram like manner (a) of all cells in a 3D plot, where $P(\tau)$ is given by the z-Axis and (b) of five specific cells with the indices 10, 120, 200, 275 and 380 plotted on top of each other in different colors.

**Probability that $i > j$ and Average Distance of $i$ and $j$:**

After the probability distribution of the developmental time orderings we calculated the probability for each cell $i$ to happen later than cell $j$ in the developmental time

$$P(\tau_i > \tau_j) = \frac{1}{Z} \sum_{\tau(x,y)} \chi(\tau_i(x,y) > \tau_j(x,y)) \tag{3.10}$$

where

$$\chi = \begin{cases} 1 & \text{if cell i} > \text{cell j in } \tau \\ 0 & \text{else} \end{cases} \tag{3.11}$$

as well as the average distance between each 2 cells $i$ and $j$

$$D(i,j) = \frac{1}{Z} \sum_{\tau(x,y)} |\tau_i(x,y) - \tau_j(x,y)|. \tag{3.12}$$

In figure 3.7 the results of the five example cells, already used in the probability distribution, have been plotted for for (a) the probability for cell $i$ being bigger than cell $j$ and (b) the average distances of $i$ to other cells. Both figures 3.7 (a) and (b) demonstrate the same thing as figure 3.6. A few cells in the beginning of the data set and cells with indices larger than roughly 240 show a well defined distance

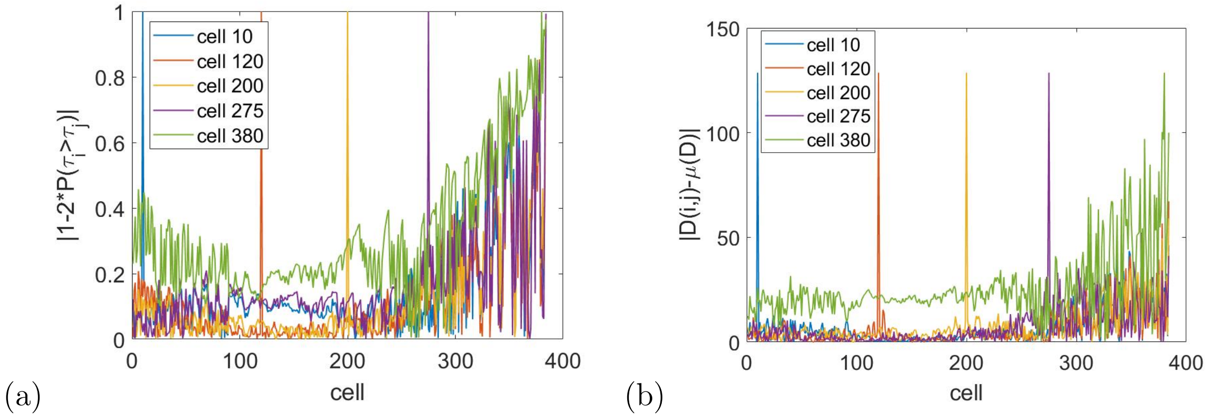


Figure 3.7: (a) Probability of five specific cells to be after each possible cell $j$ in the developmental time order plotted as $|1 - 2 * P(\tau_i > \tau_j)|$ with $i = 1, .., 5$ and $j = 1, ..., N$. (b) Absolute value of the average distance of five specific cells to each other possible cell $j$ minus the mean of the distances: $|D(i,j) - \mu(D)|$ with $i = 1, .., 5$ and $j = 1, ..., N$

with respect to other cells, while the rest of the cells seem to lump together. Coarse graining the temporal time order appropriately may therefore become a challenge in its own right. Each of the cells plotted in 3.7 exhibits one specifically high spike in both (a) and (b). This is where the measures compare the cell to itself and does not contain any information about the time order. In general it seems that the further apart in developmental time a cell is to its neighbours, the clearer its position in any given ordering approach. A valid measure of how well defined certain parts of our data and their orderings are is the entropy of the probability distribution.

**Entropy:**

The entropy of the probability distribution for each developmental time order $\tau_i$ is calculated using the the probability for a cell to be at a certain place in the developmental time $P(\tau_i > \tau_j) = P_{ij}$ and the formula for entropy

$$S_i(P) = -\sum_j P_{ij} \, log \, P_{ij} \tag{3.13}$$

Figure 3.8 shows that the entropy in the beginning and especially the end of the plot is lower. This means that the cells in this area of the sample can be placed more accurate than the ones in the middle as a higher entropy is equivalent to a less ordered state. This further confirms the assumptions we draw from figures 3.6 and 3.7. The reason for this heterogeneity in the sample is not clear, but it could very well be the case that some of the cells are closer to each other in developmental time than others. Different developmental time orders might therefore place them at different positions, which leads to a broadening of the probability distribution and therefore an increase in its entropy.

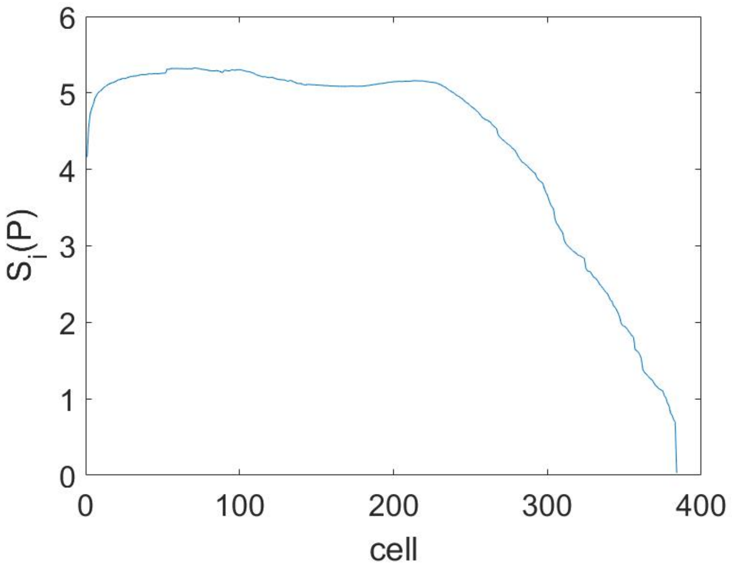


Figure 3.8: Entropy $S(P)$ of the probability distribution $P(\tau)$ as calculated in formula 3.8 for each cell. The cell order in the horizontal axis is the same as the one of the original data set.

### 3.3.4 Expected time ordering

The expected order of the cells can be obtained from using all the individual time orderings. This is done for each metric separately. Specifically, the results for the probability that a cell $i$ is bigger than a cell $j$, as discussed above, are used to achieve this order. We use a basic sorting program to switch the order of the indices. We define a vector that contains the numbers one to N in order. We use this vector to address each entry in the matrix $P_{ij}$ line by line. For each entry that is smaller than 0.5 we switch the according indices of the vector. We start with the first line, displaying the first time order and use the reordered vector to address the second line and so on. This leads to the most likely result of the developmental time ordering by considering each individual order.

### 3.3.5 Robustness

Now that we have several orderings of the cells $i$, each of the orderings corresponding to a distinct metric, we decided to compare their results to have some measure of the quality of our orderings. The expected developmental time ordering of each of the Jaccard metrics has been plotted against the expected developmental time ordering of the L1 metric in figure 3.9. (In our experiments we also produced scatter plots scattering time orders obtained from Jaccard metrices with varying cut-offs. Their results were similar to the plots against the L1 metric with slightly better overlaps. However, we do not show them here as they do not deliver any particularly remarkable results that would add information to this discussion except that the particular choice of the cut-off seems to play only a minor role. Instead we focus on the comparison of two different approaches within this thesis.) The plots in figure 3.9 show that the developmental time orderings of the various considered metrics are quite similar and support the assumption that the cells less well localized

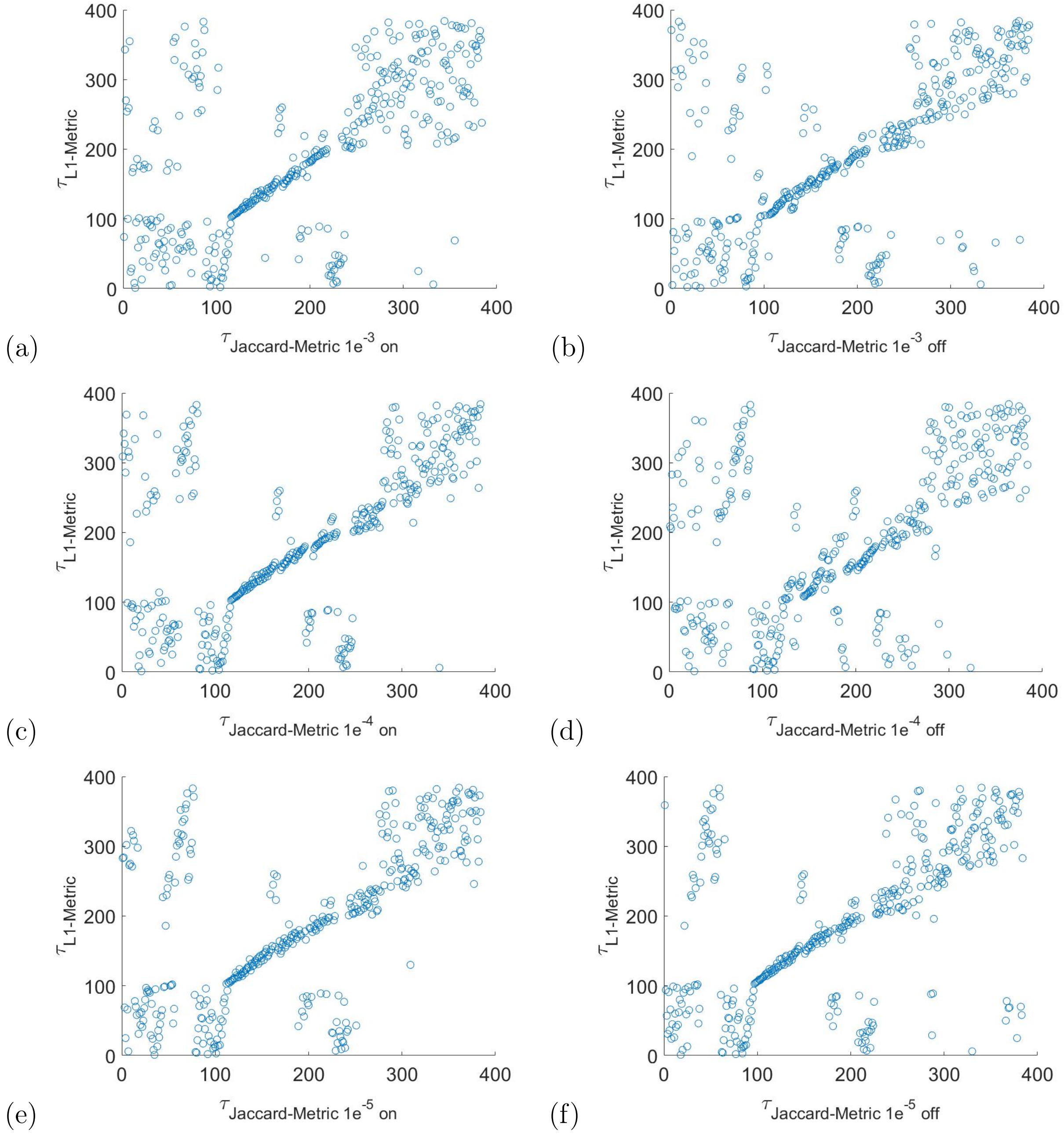


Figure 3.9: Plots of the time ordering gained from the L1 metric $\tau_{L1}$ against the time ordering gained from the different Jaccard metrics $\tau_{Jaccard}$ (The right column (a),(c) and (e) are gained from a Jaccard metric that calculated the overlap of the Jaccard metric of the non-zero values, while the left column (b),(d) und (f) being Jaccard metrics calculated from the zero entries. The first row (a) and (b) sets the threshold for zero at $10^{-3}$ the second row (c) and (d) at $10^{-4}$ and the third row (e) and (f) at $10^{-5}$.

in developmental time belong mainly to two populations of cells that mark the beginning and the end of a transitional process from one to the other population. A purely one dimensional developmental manifold may also be slightly too primitive, yet the assumption that a fairly well defined sub set of cells can be brought into a one dimensional order can be substantiated. Especially the middle part seems to be very well characterized through our developmental time ordering procedure, widely independent from the exact choice of metric. Almost all Jaccard metrics vary only little when compared to the L1 metric based ordering. The diffusion in the lower left and upper right parts of the plots can very likely be attested to a slightly less precise ordering of the cells in that areas. This result is exactly what we would expect from figure 3.8, the part of the cells with low entropy is well ordered and the part with high entropy is scattered. The significant overlap of the cell orders in figure 3.9 attests us that our approach works sufficiently. It also demonstrates that the experimental data has a sufficient signal to noise ratio for dimensional reduction methods to become applicable.

Before performing the steepest descent, we aim to further improve our results. We use plot (f) in figure 3.9 as the basis for further calculations as the developmental time ordering of the Jaccard metric with switched off genes and a threshold of $10^{-5}$ visually has the best overlap with the ordering of the L1 metric. We focus on the cells of the area in the plot with the clearest overlap in all further calculations as they are the most reliable part of the data. To further specify the quality of our developmental time order, we look at its smoothness in comparison to a randomized data set.

# Chapter 4

# Results

## 4.1 Increments

Steepest descent optimization procedures are extremely versatile to use in sufficiently smooth optimization problems with a well defined global optimum. Local optima can in principle trap a steepest decent algorithm which can be counteracted by making the gradient descent steps slightly noisy. We want to filter out cells and genes that may cause the gradient descent algorithm trouble in terms of data and noisy dimensions. To do this we calculated the increments (the absolute values of the differences between consecutive values) that arise from the change in the expression values of each gene throughout the developmental time. We averaged over the results of each gene for all cells chosen above. Then we repeated the process for the same genes (and cells) after randomizing their order. The result is shown in figure 4.1 (a) and (b). It can be seen that the increments of the scrambled gene order are on average larger than the increments of the time ordered genes. The time-ordering we introduced therefore also increased the smoothness of the developmental time evolution of gene expressions captured by the cell sample. Furthermore, there clearly are genes that seem to have large increments even after developmental time ordering. In order to filter out genes that have not merely larger increments due to their larger average expression values, we calculate a vector $rb_i$ describing ratio between the time ordered and scambled increments, that is defined as

$$rb_i(t) = \frac{\Delta x_i^{\text{ordered}}(t)}{\Delta x_i^{\text{scrambled}}(t)} \ . \tag{4.1}$$

This vector was plotted for each gene as well in figure 4.1 (c). The plot also contains the mean of $rb_i$ as straight continuous red line and its first standard deviation on one side of the mean as red dashed lines. The mean of $rb_i$ is around 0.5 while the standard deviation is at around 0.1. None of its entries is bigger than 1, which again means that our developmental time order improved the smoothness of all recorded gene expressions. We now filter out the genes which show an improvement of smoothness by a factor larger than the mean factor plus one standard deviation. We filter those genes out, since we can expect their contribution to the error functional we want to optimize for inferring coefficients of the regulatory dynamics from the data, to be the most noisy ones. We then take the original data set, remove those genes and calculate the L1 metric and the developmental time again in the same way as before. (From here on we only use the L1 metric.) In the next step we finally set up the

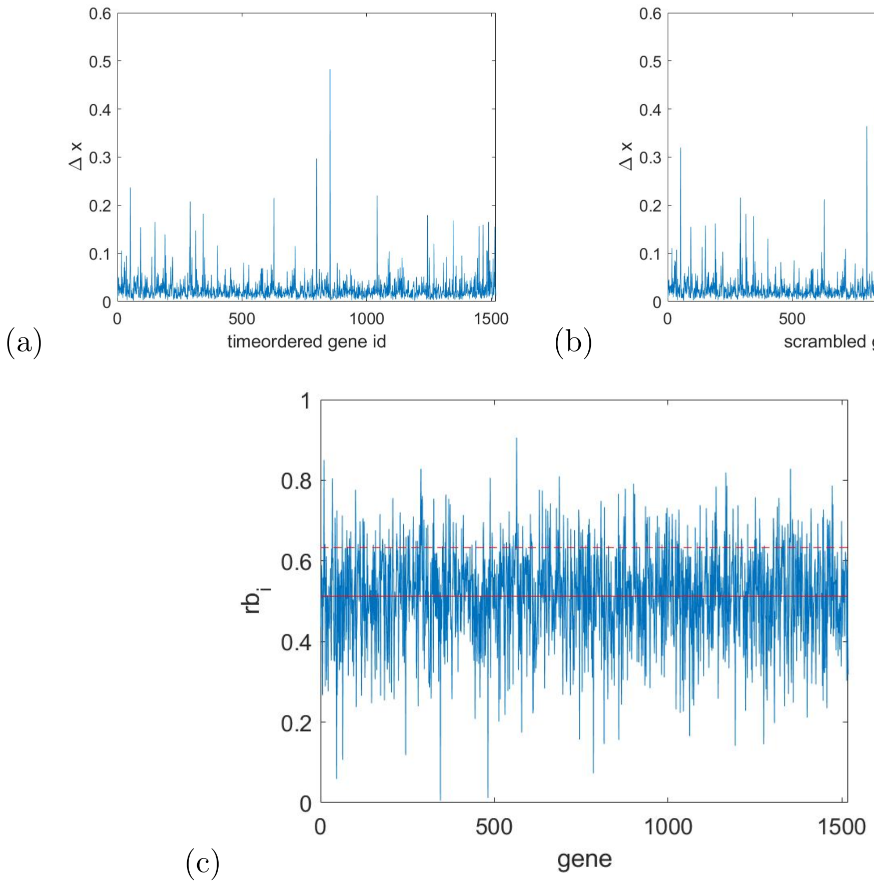


Figure 4.1: Increments $\Delta x$ of each gene (a) when ordered in developmental time, (b) when scrambled and (c) in the ratio $rb_i$ to each other with mean (continuous red line) and fist standard deviation above the mean (dashed red line)

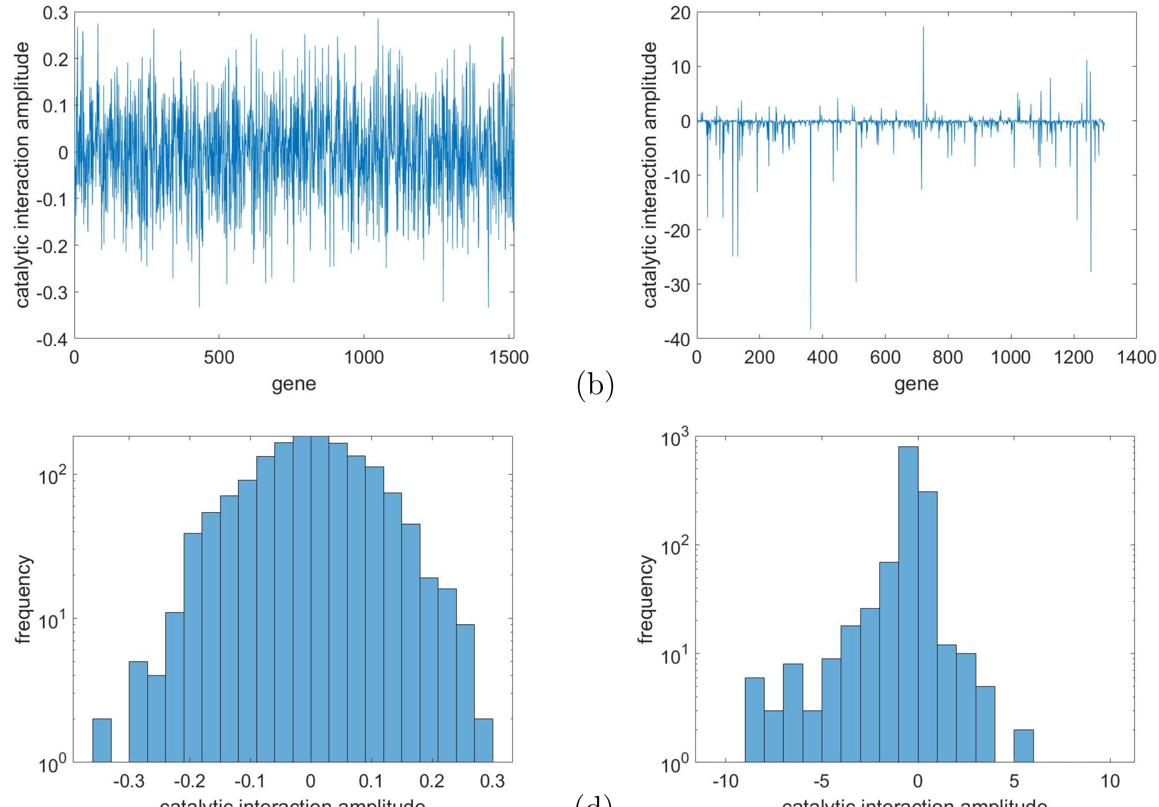
error functional that we want to minimize using steepest descent methodology.

## 4.2 Steepest descent: RNA velocity vs developmental time ordering

We perform the steepest descent to fit $J$ and $A$ in the formula 2.5 for both approximations of $\dot{x}$ and compare the results. To do so we minimize the error functional

$$\sigma^2(J, A) = \sum_t \sum_i (\dot{x}_i - (J_i + A_{ij}x_j))^2 \tag{4.2}$$

with respect to $J$ and $A$. The results for $J$, the diagonal of $A$ and the whole matrix $A$ can be seen in figures 4.2-4.4, respectively. Comparing the two methods it can be seen that the RNA velocity method seems to yield mostly uninformative results for $J$ and $A$. Figures 4.2-4.4 (a) are visually very noisy and the corresponding histograms in figures 4.2-4.4 (c) are washed out to a point, where they look like Gaussian distributions around zero. Especially the matrix $A$ is indistinguishable from a matrix with elements $A_{mn}$ sampled from a normal distribution with zero mean. In comparison, the plots of the developmental time ordering show much more promising results in terms of regulatory structure. Figures 4.2-4.4 (b) and (d)



(a)

(b)

(c)

(d)

Figure 4.2: This figure shows estimates for the vector $J$ with respect to the two different methods of estimating time derivatives, that we have employed. The left column (plots (a) and (c)) shows the results of the RNA velocity methods, while the right column (plot (a) and (c)) depicts the outcome for the developmental time order method. The catalytic interaction amplitudes of the genes of both methods are captured in two different ways. Figures (a) and (b) show the amplitudes of each gene, while figures (c) and (d) show the distribution of the amplitudes in a histogram with logarithmic y-axis. In the histogram of plot (d) there are a few outliers outside the depicted area on both sides with the values being in the range of -40 to 18, as also suggested by the expression value amplitudes in plot (b).

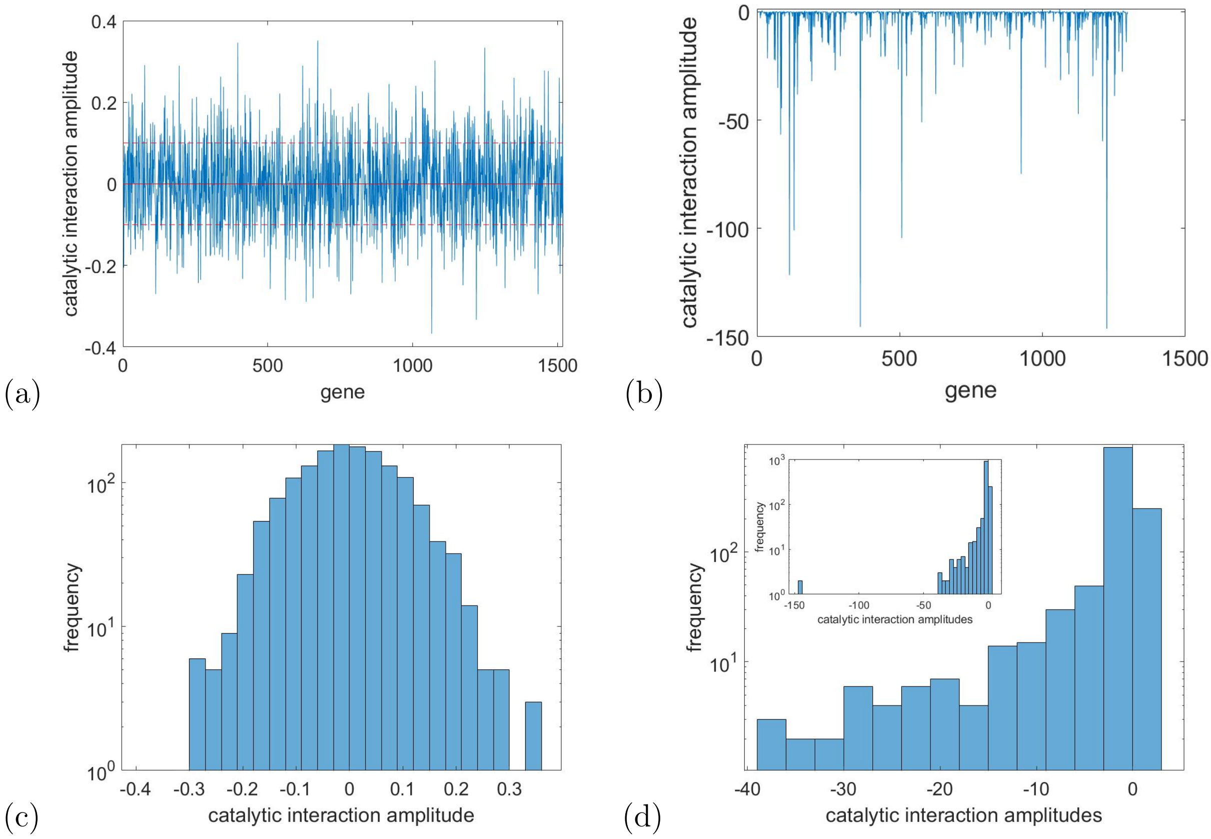


Figure 4.3: The amplitudes of the diagonal entries of the interaction matrix $A$ are shown gene-wise in (a) and (b) as well as by frequency in histograms (c) and (d). (a) and (c) show the RNA velocity approach and (b) and (d) show the developmental time approach. In (a) the mean (continuous red line) and the standard deviation to both sides (dashed red lines) of the amplitudes are plotted. In (d) the major plot shows the part of the histogram with the most values, while the insert shows the whole range of the distribution. Unlike in (c) the diagonal elements are mostly negative in (d) as decay-rates should be.

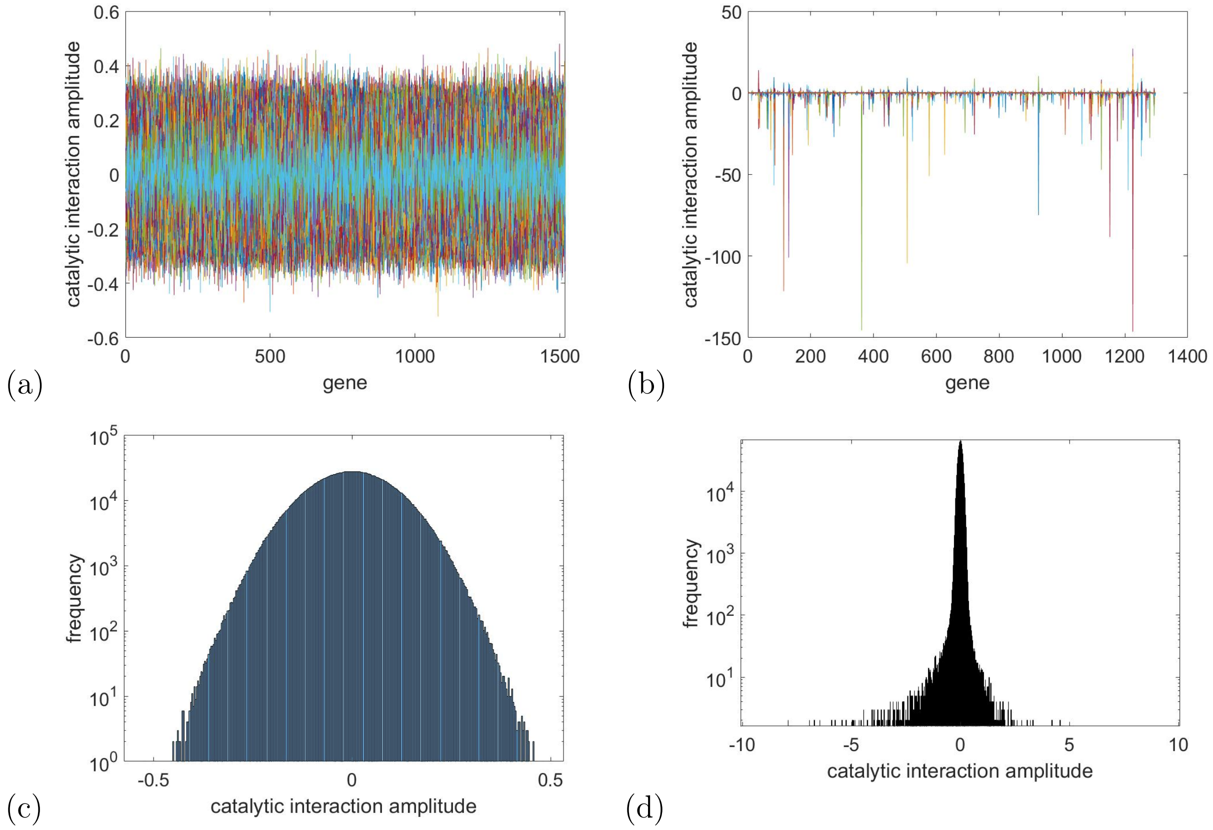


Figure 4.4: For each gene the catalytic interaction amplitudes, that mark the cooperative regulatory effects with the other genes, are plotted in a different color in (a) and (b). Those graphs are plotted on top of each other. The histograms in (c) and (d) show the frequency of the catalytic interaction amplitudes. The results of the RNA velocity method can be found in (a) and (c) and the results of the developmental time order method are shown in (b) and (d). In plot (d) a few outliers on both sides have been cut out. They are placed between -146 and 27.

show a few entries with significant catalytic interaction amplitudes, while the rest of them are close to zero. These results are in accordance hypothizing that there are only a few genes driving the process while regulating each other. The rest of the genes play a subordinate roll and interaction strengths are virtually indistinguishable from noise distributed closely around zero. In developmental time ordering we seem to observe a clear skew of the distribution function towards negative values, which may support what one could also suspect from systemic stability considerations. Specifically, the entries in the diagonal of $A$ are equivalent to the decay rates and hence are expected to be zero or smaller, if the dynamical system should maintain stability, avoiding unchecked exponential growth, that is. This also nicely matches the intuition that in a dynamical system as described by 2.5 only can maintain stable fixed points or limit cycles if the diagonal elements of $A$ have negative values. This condition is sufficiently met by the developmental time order approach, as shown in figure 4.3 (b) and (d). We use the results of the developmental time order to identify the genes with the most significant effect on the process.

## 4.3 Most influencal genes

We use the results of the steepest descent with the developmental time approach to find the genes with the biggest impact on the process. Those genes are the strongest entries in $A$ and $J$. We are interested in those amplitudes that can be sufficiently distinguished from noise. To estimate the number of genes that matter in this manner, we first define a vector $N_{above}$ that defines the number of genes above a certain threshold per line.

$$N^{i}_{above} = f \cdot \mu_i \tag{4.3}$$

with $\mu_i$ being the mean of line i in $A$ and $f$ being a multiplication factor that can be chosen freely. We choose $f = 2$ for this calculation, because our experiments show it to be the most successful setting.

Then we define another measure $N_{below}$ that uses the mean and standard deviation of $N_{above}$ as well as the factor $f$ to estimate the number of genes that significantly differ from noise.

$$N_{below} = \mu(N_{above}) - f \cdot \sigma(N_{above}) \tag{4.4}$$

we use the results from above and $f = 2$ to calculate $N^{below}$. This results in $N_{below} = 80$. The 80 genes with the biggest impact on the process are the ones with the 80 strongest entries in $J$:

Stat1, Abi2, Smarcb1, Dtx3, Rpl41, Dctn2, 0610010F05Rik, Rhbdf1, Hspa4, Chd3, Ogdh, Psme4, Mprip, Psmd3, Ddx42, Psmd12, Sel1l, Hsp90aa1, Itsn2, Rrm2, Dlk1, Ppp2r5c, Fancc, Dapk1, Gmpr2, Trio, Phf20l1, Cyc1, Cct8, Eif4g1, Dnajb11, Rpl35a, Atg3, Ttc3, Tbp, Vars, Eml4, Cdh2, Kdm3b, Sorcs1, Orc4, Mtx2, Naa20, Ctsa, Npepl1, Dhx36, Rabggtb, Csde1, Mast2, Chchd7, Epha5, Gatc, Anapc5, Akap9, Lias, Mlf2, Shkbp1, Ap3b2, Epn1, Arhgef1, Vps33b, Trpc2, Ampd3, Cep44, Rbmxl1, Rad23a, Atp6v1b2, Zfp560, Atm, Map2k1, Hemk1, Prkcsh, Scn3b, C2cd4b, Sltm, Rpl14, Deb1, Flna, Usp9x, Ogt

A matrix that summarizes those specific 80 genes and their interactions from the matrix $A$ can be seen in figure 4.5. Negative values in the matrix mean that the genes suppress the process or interaction described, making them so-called *suppressors*, while genes with positive values strengthen or even enable a process and therefore can be called *catalysts*. This holds up for all non-diagonal entries. The diagonal entries - as discussed above - state the decay rates of the individual genes. The numerical values of the coefficients in $J$ and $A$ with respect to the of the 15 most indicated genes are shown in figure A.1.

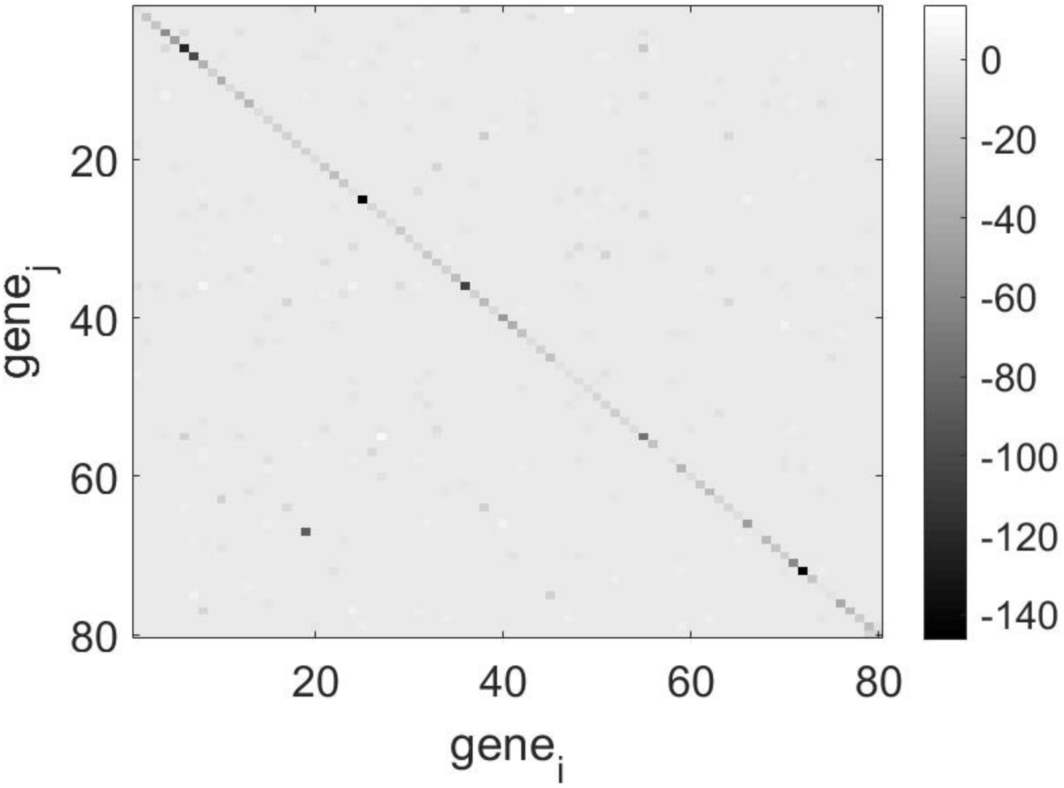


Figure 4.5: Intensity plot of the interaction matrix. It contains the elements of $A$ with amplitudes strong enough to not be considered noise. Each point in the plot is a color coded entry in the matrix, showing the interactions between the genes. The order of the genes in the matrix is given by the order of the top 80 genes on page 35.

# Chapter 5

# Discussion

## 5.1 Fitting the model parameters

Before discussing the for the coefficients of equation 2.5 that we obtain by a steepest descent based minimization of the error-functional

$$\sigma^2(J, A) = \sum_t \sum_i (\dot{x}_i - (J_i + A_{ij}x_j))^2 \tag{4.2}$$

a few remarks are in order. Fitting $J$ and $A$ is the least certain part about the analysis. This is also the first attempt in doing anything like this, so it is likely that the method might not work perfectly yet. As the data is normalized, this would, in theory, require a more complex functional that includes the addition of a normalization term to improve the functioning of the steepest descent. To simplify the problem we also assumed that $\Delta t = 1$ for the time steps at all times, while this might not actually be the case. Furthermore, in our steepest descent we chose not to add noise, because first tests to do so only delivered unusable results. However, in future works this might as well be an option, when investigating this issue deeper and discovering its problems. This master thesis can hence be seen more as a test determining if a method like this can function at all, than as an exact model of the process analyzed in it. To assess the methods we used we discuss their strengths and weaknesses in detail.

### 5.1.1 RNA velocity results

In the first method we estimate the derivative of $x$ in the ODE 2.5 by approximating a time step with data that is projected into the future and then fit $J$ and $A$ of the ODE. To explain the outcome of our experiment in terms of the RNA velocity, we analyze the connection of the actual data and the projected data. To gain a clear insight into the two data sets, we take the mean of the gene expression amplitudes of each cell in the measured values, sort them by their size and plot them. Then we calculate the mean of each cell in the projected data set and plot the results in the same order as the measured values in the same diagram in figure 5.1 (a). We repeat the procedure for the average standard deviation of each cell as shown in figure 5.1 (b). The results for mean and variance of original data and projected data differ considerably, meaning that the mean changes about an order of the standard deviation. In a

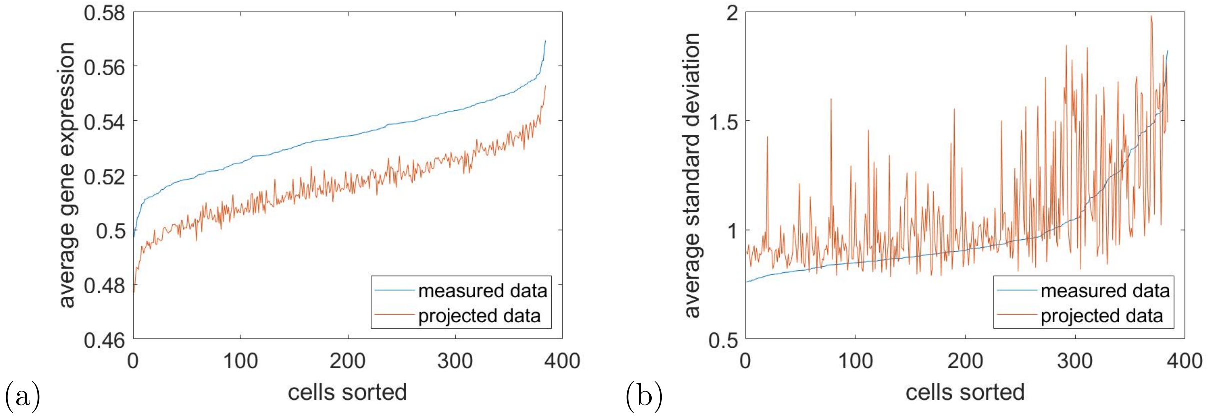


Figure 5.1: (a) Mean and (b) standard deviation of the originally measured and the projected data set for each cell in comparison. The measured data set was ordered according to the size of the mean and standard deviation, respectively, while the projected data set was plotted in the same order as the measured data set.

data set of around 400 cells that changes in time, a shift into the future like that, should not cause a change of the mean and standard deviation by this rate. We may speculate that the increment is too severe for the steepest decent method to find sufficiently well defined minima or that the projected data captures an overall trend with little information about complex regulatory interactions. These graphics might explain, why the results we obtained from the steepest descent yielded no more than white noise. It appears that the RNA velocity approach, despite the valuable insights it yields with respect to estimating RNA turnover rates, does not work well together with the steepest decent methods for estimating the parameters of differential equations describing complex regulatory dynamics. We suspect that the problem lies in the assumptions made in the method of projection. The two models used to determine the future state of the gene expression rates are the model of *constant velocity assumption*

$$s(t) = s_0 + vt \tag{2.21}$$

and the model of *constant unspliced assumption*

$$\frac{ds}{dt} = u_0 - \gamma s(t) \tag{2.22}$$

In both cases $s$ is the amplitude of only one single gene. As there is no info of other gene expressions, the information gained from this calculation could be washed out. Furthermore, if we write the ODE model from equation 2.5 as

$$\frac{dx}{dt} = J + Ax \tag{5.1}$$

we can see that it almost looks the same as equation 2.22. There are a few differences when looking at it in detail. $s$ that would be equivalent to $x$ is, as discussed above, only the amplitude of one gene, while $x$ contains all of the measured genes. In accordance $u_0$ is also only a single value, while $J$ is a vector. From our point of few the most significant difference between the equations, however, is $\gamma$ which is not equivalent to the matrix $-A$, but to the diagonal of $-A$. $\gamma$ only describes the decay rate and therefore does not contain information about the interactions of genes in

the process. The value of $\gamma$ also has been gained experimentally and cooperative effects between genes only enter indirectly into $\gamma$ through the ratios of spliced and unspliced RNA. We therefore may speculate that cooperative effects are washed out in the RNA velocity approach so that methods that try to infer cooperative effects from this data, as we do, by fitting an ODE to the data, get insufficient to extract useful information about cooperative dynamics in this way.
In our second approach we demonstrate that inferring developmental time from data together with optimizing parameters of ODEs describing the developmental dynamics nonetheless has its merits and may provide a viable mathematical methodology for extracting cooperative regulatory information from single cell RNA seqencing data. Integrating RNA velocity techniques adequately into the developmental time based framework may be nonetheless desirable. However, this is work that goes beyond the scope of this thesis.

### 5.1.2 Developmental time ordering and ODE fitting

In this method we order the measured cell data by the small differences in their developmental time and use them to approximate the derivative of $x$ and then use the results to fit the ODE with a steepest descent. We pick the 80 genes with the least random expression values as final results to look at their cooperative regulatory functions. We now compare those results to current knowledge about the cell biological processes they are involved in. The following genes stand out the most, because their change in gene expression values shows the best accordance with the change in cell population during the developmental process:

Dlk1, Ampd3, Rrm2, Scn3b,C2cd4b, Atp6v1b2, Flna, Cep44, Rhbdf1, Epha5, Cdh2, Ap3b2, Npepl1, Sorcs1, Dapk1
[20]

Interestingly and to some extent surprisingly, most of the genes are per se not transcription factors (proteins that influence the gene transcription and therefore have regulatory properties), as one would expect. Two prominent and well researched genes that do have regulatory effects are Dlk1 and Stat1. Dlk1 is known to be a regulator in the Notch signalling pathway [21]. However, in our interaction matrix, Dlk1 does not show a regulatory effect that would support this connection to the Notch signalling pathway. Stat1 is known to be a transcription factor [22], but does not show any change in gene expression that could be linked to a change in the cell population during the process. The method seems to extract enough information from the data to indicate genes that change their expression values along the developmental time line, and therefore have partly been used in other methods for constructing pseudotime lines from data themselves. However, this information seems insufficient for really pinpointing the regulator genes.
Those results are encouraging even though there remains a lot of room for improvement when it comes to estimate ODE model parameters from RNA sequencing data. We conclude that while our method seems to be promising, it performs similarly to other methods [23][24][25][26] currently used in this emerging field, which have to deal with the same or similar uncertainties we encounter in this work, uncertainties that are arising from the a priorily unknown ground truth.

## 5.2 Dimensional embedding and the dimension of the developmental manifold

In the previous analyses we assumed that the actual expression space is one dimensional and the dimensional reduction was performed to project the data onto one dimension. It can of course be the case that the data might as well lie not in one, but in another (low) number of dimensions and an analysis of this low dimensional developmental manifold would in fact be required. We may, for instance, think of a two-dimensional developmental space where the cell-cycle or the circadian cycle serves as one developmental dimension while the differentiation-flow of cells involved in the developmental process constitutes the second dimension. Such a generalization of our methodology could in fact be achieved by using an ordering algorithm similar to the one we used for one dimension. One would just have to project the cells onto a plane (for two dimensions) or a cube (for three dimensions) instead of a one dimensional line. The downside of this method is that it leads to increasing complexity of the code and an increasing run-time. We therefore use another method, the method of *dimensional embedding*, to take a look at a fit of the dimensions $d = 2$ and $d = 3$. A slightly different approach to the subtraction of noise is made here. To estimate the noise in the dimensional embedding we take a closer look at the distribution of the minimal cell distances. In a sample of $N$ cells we define the minimum distance of another cell to cell $i$ as

$$D_i \equiv \min\{d_{ij}|j \neq i\} \tag{5.2}$$

where

$$d_{ij} \equiv ||X_i - X_j|| \tag{5.3}$$

where $X_i$ is a L1-normalized cell from the single cell RNA sequencing data. The distribution can be seen in figure 5.2.

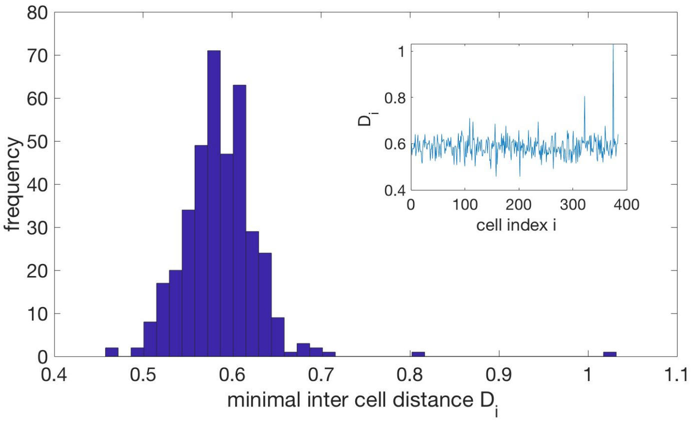


Figure 5.2: The figure shows the distribution of the minimal cell to cell distances $D_i = min\{d_{ij}|j \neq i\}$ for all cells $i$. In the inset we see the actual minimal distances for each cell index $i = 1, ..., N$

If we interpret this minimal distance to other cells as the result of noise we can assume that $d_{ij}$ should be close to zero without noise. If we assume that each cell experiences about the same amount of noise, then we can try to correct the observed distances in at least three different ways:

(i) $d^*_{ij} \equiv |d_{ij} - \lambda\frac{1}{2}(D_j + D_i)|$

(ii) $d^*_{ij} \equiv |d_{ij} - \lambda \mathrm{min} D_j|$

(iii) $d^*_{ij} \equiv |d_{ij} - \lambda \langle D_j \rangle|$

where $0 < \lambda \leq 1$ is likely to be a number close to one. Note that $d_{ij}$ in all three cases remains symmetric. Moreover, $\langle D_j \rangle$ is the sample mean of the minimal distances $D_i$ and $\mathrm{min} D_j$ is the minimum of those values.
Geometric embedding can be achieved again by minimizing an error functional. Suppose that for each cell $i$ we choose some position $y_i \in \Re^d$ , with the representation dimension $d = 1, 2, 3, ...$, then we can define the functional

$$\sigma^2(y) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left( ||y_i - y_j||_1^\alpha - d^*_{ij}(\lambda)^\alpha \right)^2 \tag{5.4}$$

where $\alpha$ is a parameter that can be used to weigh error more severely for large distances ($\alpha > 1$) or smaller distance ($\alpha < 1$). $\lambda > 0$ and $\alpha > 0$ can in principle be chosen freely. In figure 5.3 we set those parameters as $\lambda = 1.0$ and $\alpha = 1.0$, which seem to be the natural choices. (Results of the dimensional embedding ($d = 2, 3$) for different parameter choices can be found in the appendix (figures B.1 - B.8).) Plot 5.3 compares the measured data in pane (1) to randomly re-sampled data with the same parameters in pane (2). In the re-sampled data it was randomized which cell a gene entry belongs to. It therefore shows what random noise would look like when using our sample. It clearly can be seen that the dimensional embedding shows structure for the measured data, when compared to the randomized data in two dimensions as well as in three. With certain parameter choices the structure of the dimensional embedding is clearly non-random. This is a , all considered, reassuring result.
A few plots like 5.3 (c) and (d) as well as B.1 (b) and (e) and B.4 (c) and (d) even show results that are almost one dimensional, but either washed out, or with the addition of a small cluster. What becomes obvious, however, is that the answers that dimensional embedding techniques will provide may crucially depend on details of how dimensional embedding is achieved. One crucial point being how noise in the high dimensional RNA expression space is dealt with. More detailed analyses will have to be done in order to understand those issues and to unlock the potential of dimensional reduction approach.
The dimensional reduction maps, e.g. 5.31(c), indicate that also a appropriate coarse-graining of the data may be indicated for instance along the one dimensional sub manifold. This could for example be done by averaging over a few cells within the same neighborhood in order to calculate a local mean over the RNA expression values of cells, to construct a "mean cell" at this location. This corresponds to averaging over noise, leading to a noise reduction in the signal, leading to a smoother representation of the process. Also one could treat possible occurring clusters or one dimensional sub-manifolds separately. This gives room for methodological improvement in future work.

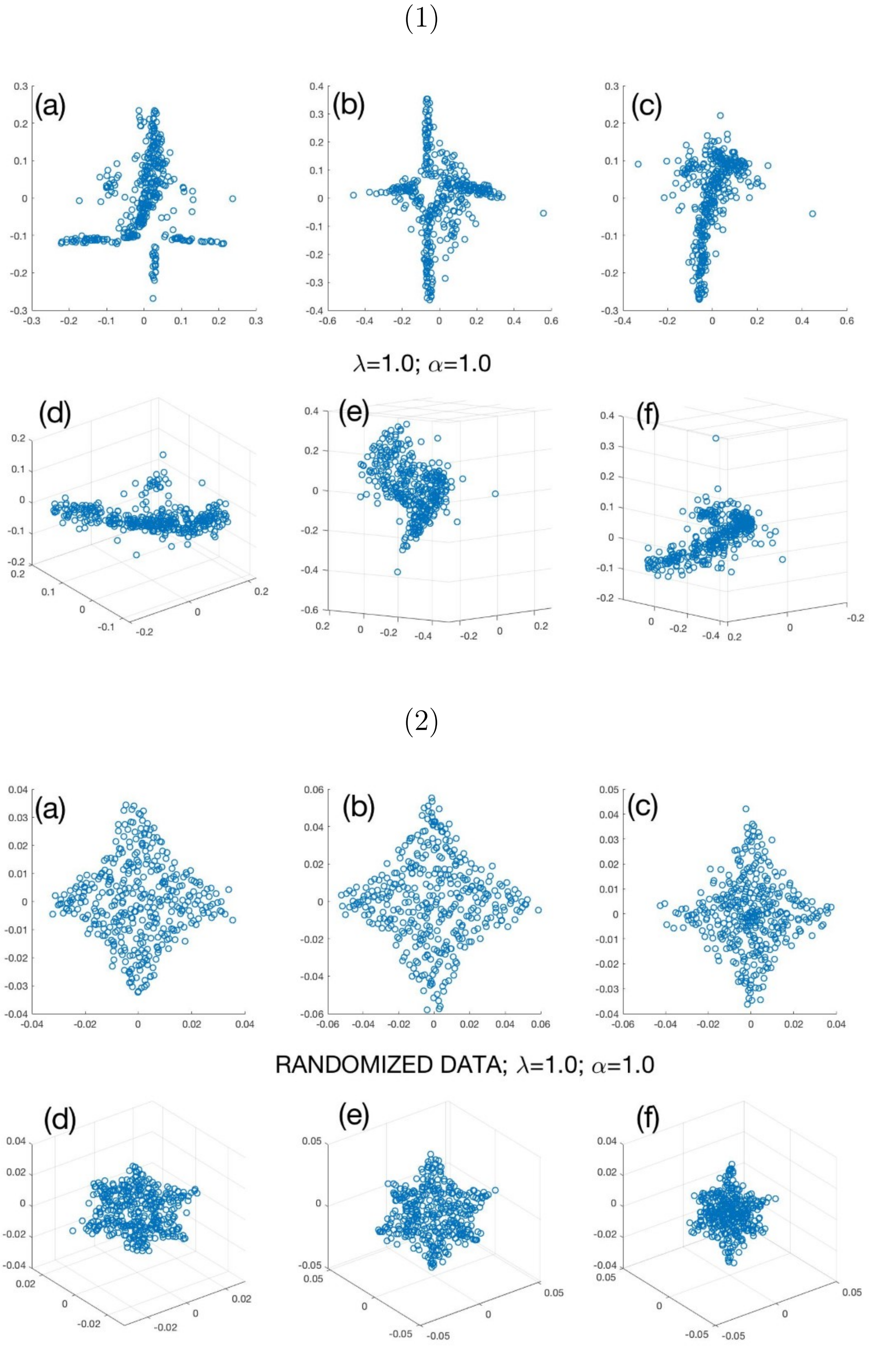


Figure 5.3: Comparison of the measured data in pane (1) and randomly re-sampled data in pane (2). Dimensional embeddings for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; (i) is shown in (a,d), (ii) in (b,e), and (iii) in (c,f). The parameters were chosen as $\lambda = 1$ and $\alpha = 1$.

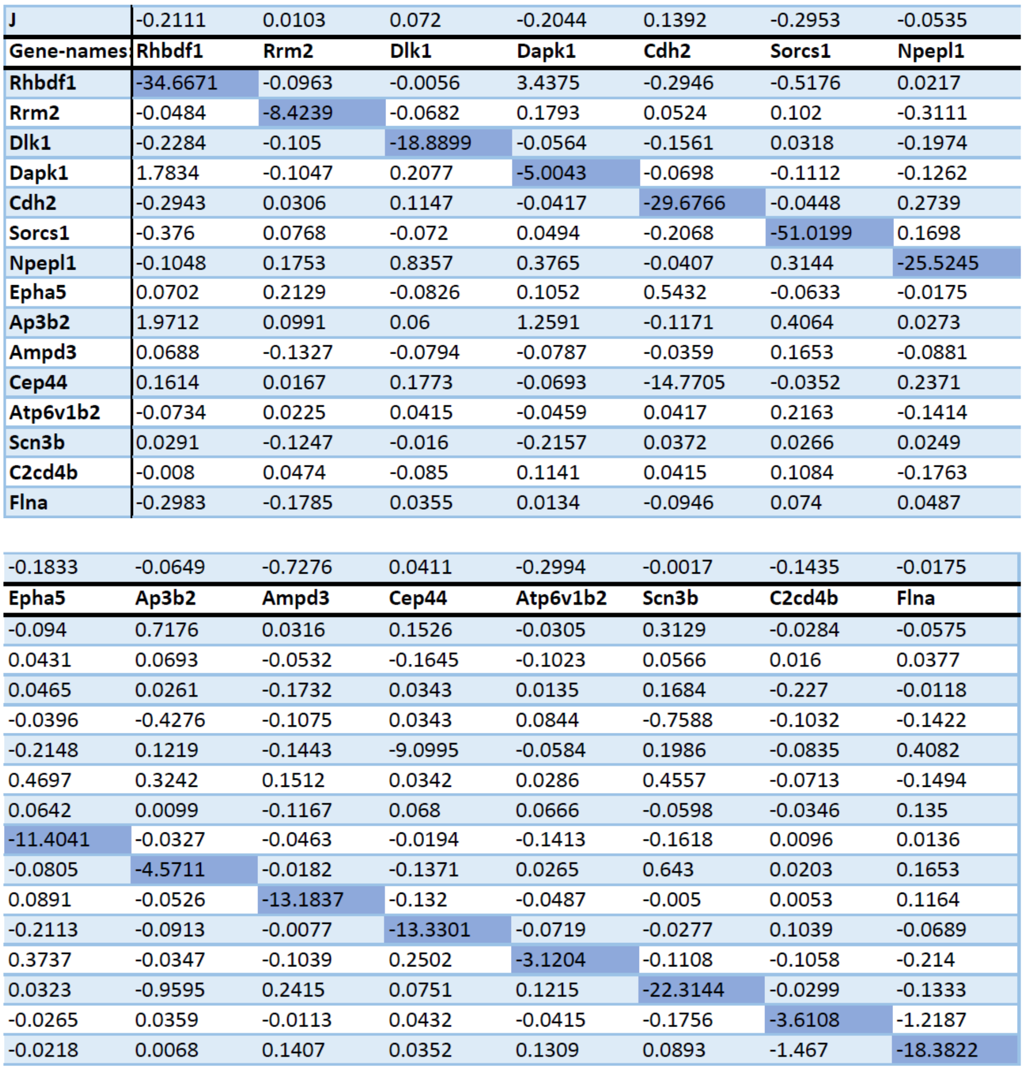
## 5.3 Outlook

This thesis is an attempt to understand the process explored and the methods used to do so better. However, there are still many things left to be researched, improved and tested in future work.
In this thesis it has been shown that dimensional embedding approaches for two or three dimensions bare a lot of potential. This approach should be examined more closely in future research.
To improve the methods employed several measures could be taken. For one, the methodology of subtracting high dimensional noise contributions from the metric distances in expression space should be improved. Additionally, possible one dimensional sub-manifolds could be identified and extracted, noise could be reduced by local averaging and the error functional for normalized data could be adapted by adding a normalization term.
In future work an improvement of the RNA velocity method should be aspired. Moreover, we need to understand how to adequately include info on RNA velocity, i.e. RNA turnover, in the approximation of differential equations. Furthermore, additional filter parameters could be included to reduce the influence of unrelated processes on the outcome. It can for example be filtered for known genetic cellular processes, such as cell cycles or circadian cycles.

# Appendix A

# Data

Due to the sheer amount of data we picked the results of the 15 most promising genes to be depicted here in figure A.1 representative for the 80x80 Matrix, that was our result after the steepest descent and filtering process.



| **J** | -0.2111 | 0.0103 | 0.072 | -0.2044 | 0.1392 | -0.2953 | -0.0535 | -0.1833 | -0.0649 | -0.7276 | 0.0411 | -0.2994 | -0.0017 | -0.1435 | -0.0175 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Gene-names** | **Rhbdf1** | **Rrm2** | **Dlk1** | **Dapk1** | **Cdh2** | **Sorcs1** | **Npepl1** | **Epha5** | **Ap3b2** | **Ampd3** | **Cep44** | **Atp6v1b2** | **Scn3b** | **C2cd4b** | **Flna** |
| **Rhbdf1** | -34.6671 | -0.0963 | -0.0056 | 3.4375 | -0.2946 | -0.5176 | 0.0217 | -0.094 | 0.7176 | 0.0316 | 0.1526 | -0.0305 | 0.3129 | -0.0284 | -0.0575 |
| **Rrm2** | -0.0484 | -8.4239 | -0.0682 | 0.1793 | 0.0524 | 0.102 | -0.3111 | 0.0431 | 0.0693 | -0.0532 | -0.1645 | -0.1023 | 0.0566 | 0.016 | 0.0377 |
| **Dlk1** | -0.2284 | -0.105 | -18.8899 | -0.0564 | -0.1561 | 0.0318 | -0.1974 | 0.0465 | 0.0261 | -0.1732 | 0.0343 | 0.0135 | 0.1684 | -0.227 | -0.0118 |
| **Dapk1** | 1.7834 | -0.1047 | 0.2077 | -5.0043 | -0.0698 | -0.1112 | -0.1262 | -0.0396 | -0.4276 | -0.1075 | 0.0343 | 0.0844 | -0.7588 | -0.1032 | -0.1422 |
| **Cdh2** | -0.2943 | 0.0306 | 0.1147 | -0.0417 | -29.6766 | -0.0448 | 0.2739 | -0.2148 | 0.1219 | -0.1443 | -9.0995 | -0.0584 | 0.1986 | -0.0835 | 0.4082 |
| **Sorcs1** | -0.376 | 0.0768 | -0.072 | 0.0494 | -0.2068 | -51.0199 | 0.1698 | 0.4697 | 0.3242 | 0.1512 | 0.0342 | 0.0286 | 0.4557 | -0.0713 | -0.1494 |
| **Npepl1** | -0.1048 | 0.1753 | 0.8357 | 0.3765 | -0.0407 | 0.3144 | -25.5245 | 0.0642 | 0.0099 | -0.1167 | 0.068 | 0.0666 | -0.0598 | -0.0346 | 0.135 |
| **Epha5** | 0.0702 | 0.2129 | -0.0826 | 0.1052 | 0.5432 | -0.0633 | -0.0175 | -11.4041 | -0.0327 | -0.0463 | -0.0194 | -0.1413 | -0.1618 | 0.0096 | 0.0136 |
| **Ap3b2** | 1.9712 | 0.0991 | 0.06 | 1.2591 | -0.1171 | 0.4064 | 0.0273 | -0.0805 | -4.5711 | -0.0182 | -0.1371 | 0.0265 | 0.643 | 0.0203 | 0.1653 |
| **Ampd3** | 0.0688 | -0.1327 | -0.0794 | -0.0787 | -0.0359 | 0.1653 | -0.0881 | 0.0891 | -0.0526 | -13.1837 | -0.132 | -0.0487 | -0.005 | 0.0053 | 0.1164 |
| **Cep44** | 0.1614 | 0.0167 | 0.1773 | -0.0693 | -14.7705 | -0.0352 | 0.2371 | -0.2113 | -0.0913 | -0.0077 | -13.3301 | -0.0719 | -0.0277 | 0.1039 | -0.0689 |
| **Atp6v1b2** | -0.0734 | 0.0225 | 0.0415 | -0.0459 | 0.0417 | 0.2163 | -0.1414 | 0.3737 | -0.0347 | -0.1039 | 0.2502 | -3.1204 | -0.1108 | -0.1058 | -0.214 |
| **Scn3b** | 0.0291 | -0.1247 | -0.016 | -0.2157 | 0.0372 | 0.0266 | 0.0249 | 0.0323 | -0.9595 | 0.2415 | 0.0751 | 0.1215 | -22.3144 | -0.0299 | -0.1333 |
| **C2cd4b** | -0.008 | 0.0474 | -0.085 | 0.1141 | 0.0415 | 0.1084 | -0.1763 | -0.0265 | 0.0359 | -0.0113 | 0.0432 | -0.0415 | -0.1756 | -3.6108 | -1.2187 |
| **Flna** | -0.2983 | -0.1785 | 0.0355 | 0.0134 | -0.0946 | 0.074 | 0.0487 | -0.0218 | 0.0068 | 0.1407 | 0.0352 | 0.1309 | 0.0893 | -1.467 | -18.3822 |

Figure A.1: The results of the 15 most intriguing genes and their interactions

# Appendix B

# Dimensional embedding

Additionally to the dimensional embedding plots with the parameters set to $\lambda = 1.0$ and $\alpha = 1.0$ that can be seen in figure 5.3, we looked at different values and value combinations in those parameters that can be seen in figures B.1 - B.8.

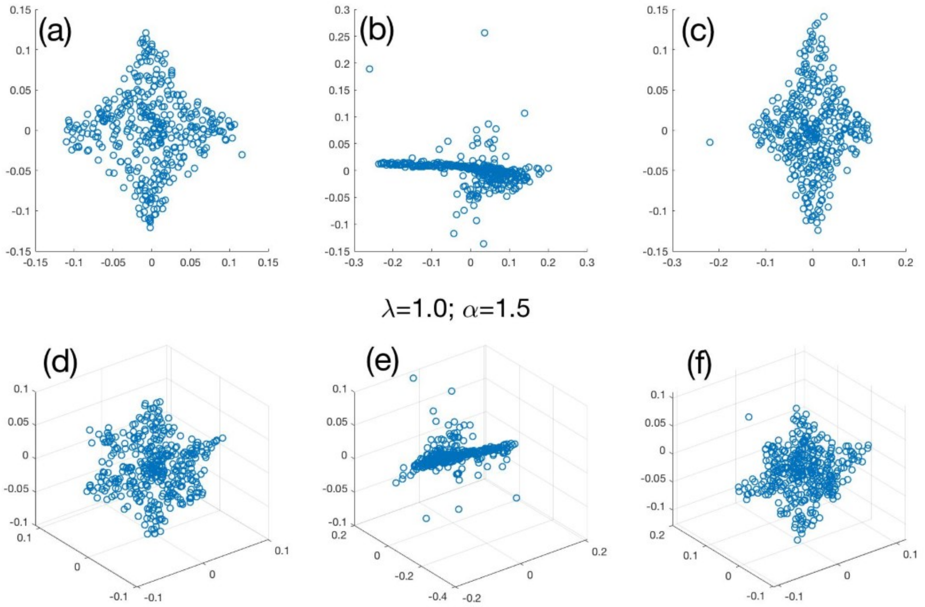


Figure B.1: Dimensional embedding with the parameters $\lambda = 1.0$ and $\alpha = 1.5$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

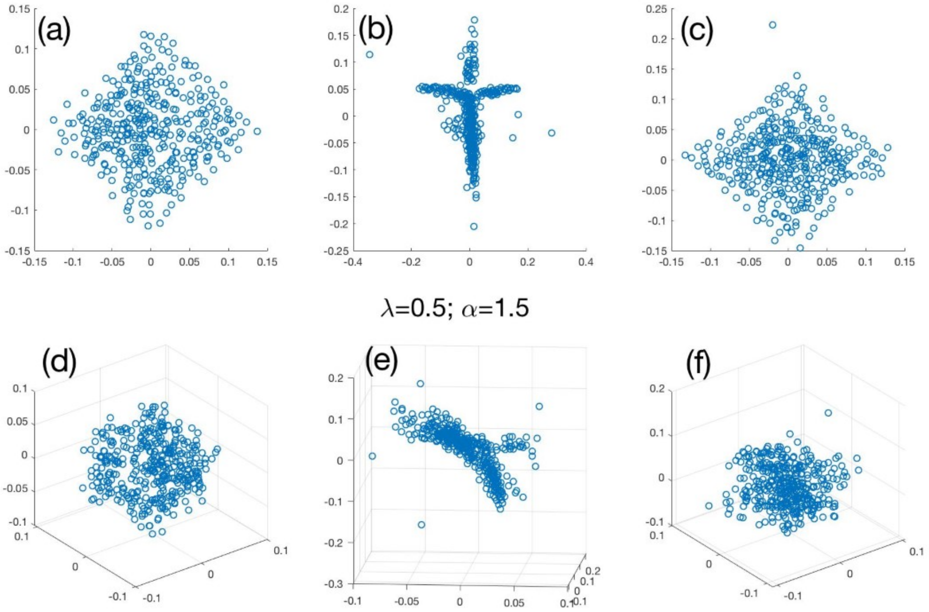


Figure B.2: Dimensional embedding with the parameters $\lambda = 0.5$ and $\alpha = 1.5$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

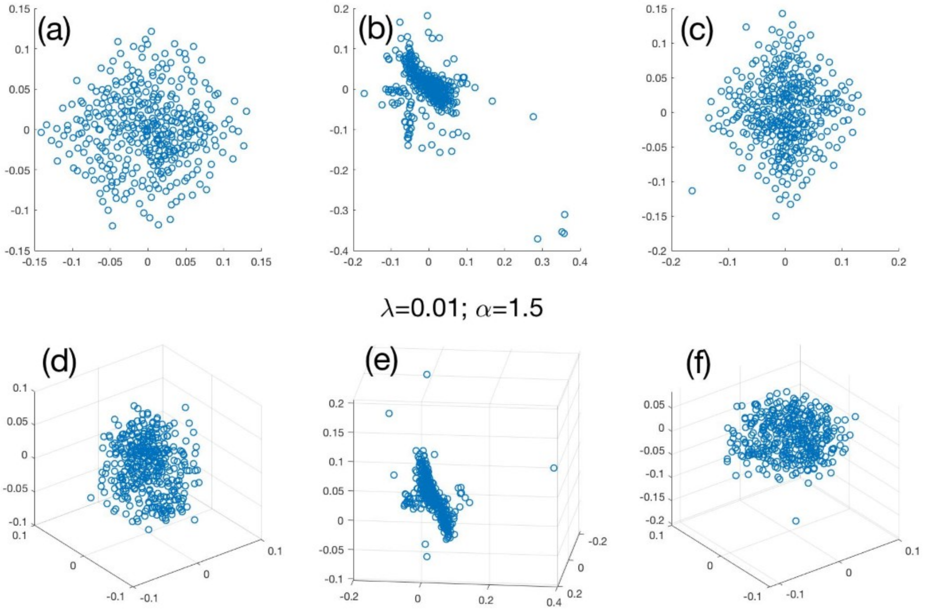


Figure B.3: Dimensional embedding with the parameters $\lambda = 0.5$ and $\alpha = 0.001$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

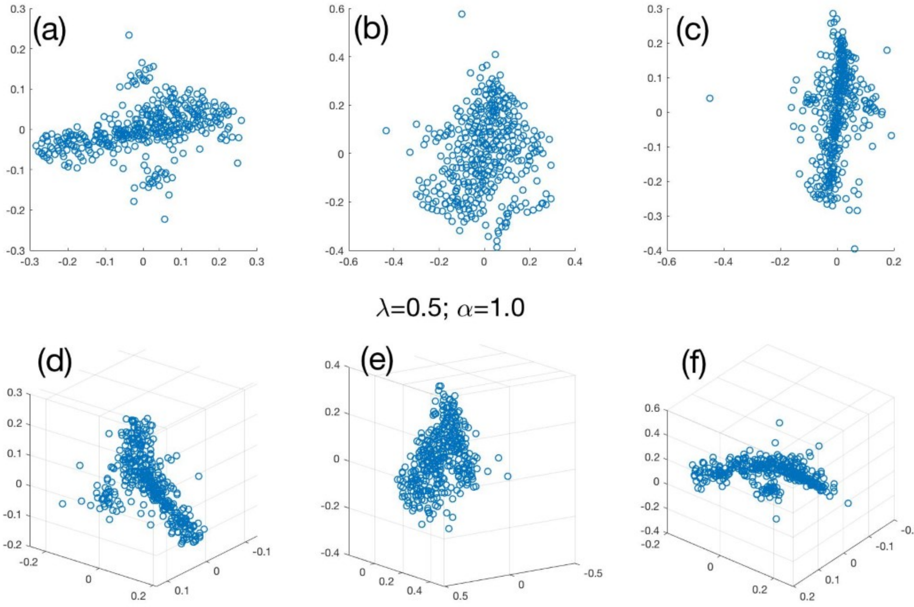


Figure B.4: Dimensional embedding with the parameters $\lambda = 0.5$ and $\alpha = 1.0$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

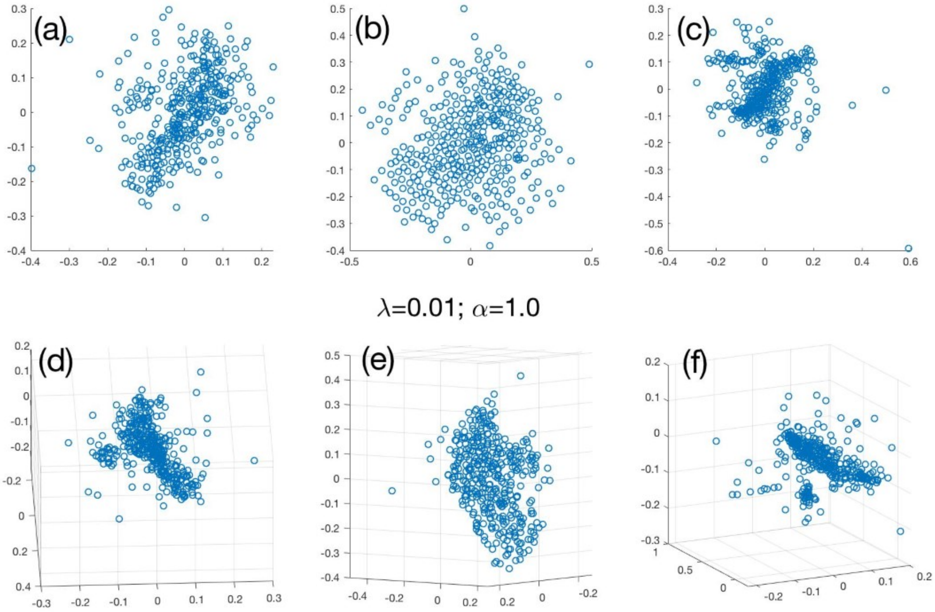


Figure B.5: Dimensional embedding with the parameters $\lambda = 0.01$ and $\alpha = 1.0$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

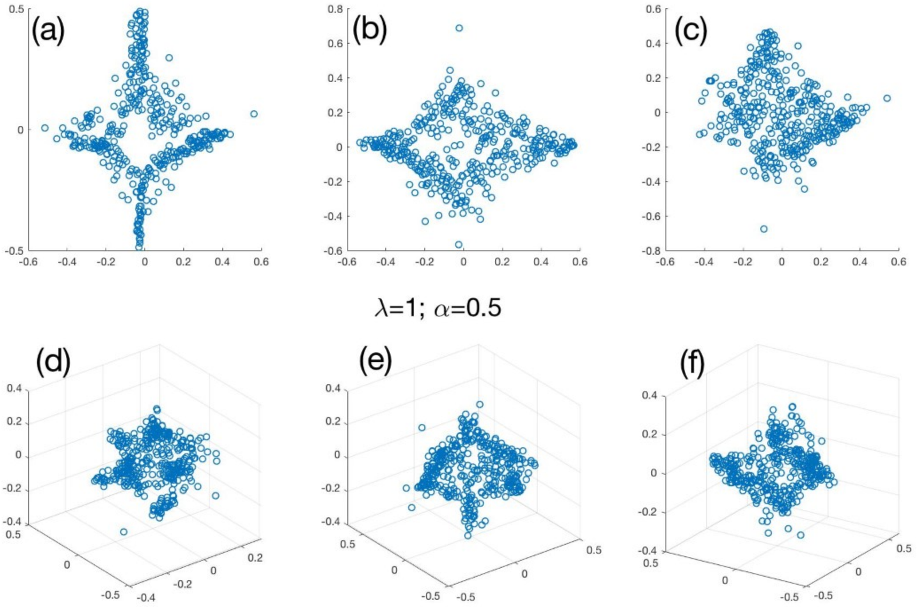


Figure B.6: Dimensional embedding with the parameters $\lambda = 1.0$ and $\alpha = 0.5$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

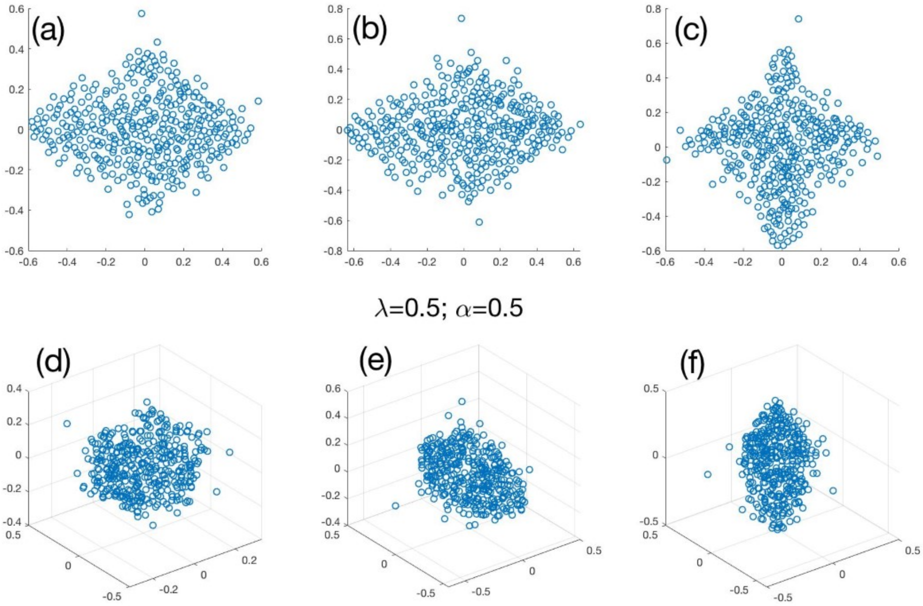


Figure B.7: Dimensional embedding with the parameters $\lambda = 0.5$ and $\alpha = 0.5$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

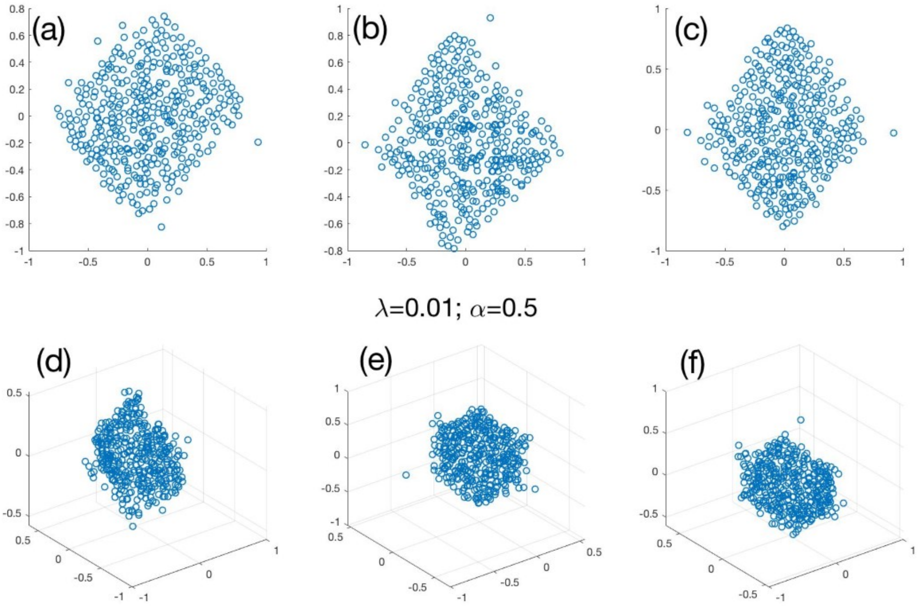


Figure B.8: Dimensional embedding with the parameters $\lambda = 0.01$ and $\alpha = 0.5$ for $d = 2$ (a,b,c) and $d = 3$ (e,f,g) and for the three cases to subtract noise; i) is shown in (a,d), ii) in (b,e), and iii) in (c,f).

# Bibliography

[1] Alessandro Furlan et al. “Multipotent peripheral glial cells generate neuroendocrine cells of the adrenal medulla”. In: *Science* 357 (2017). DOI: `https://doi.org/10.1126/science.aal3753`.

[2] Mollie B. Woodworth, Kelly M. Girskis, and Christopher A. Walsh. “Building a lineage from single cells: genetic techniques for cell lineage tracking”. In: *Nature Reviews Genetics* 18 (2017), pp. 230–244. DOI: `https://doi.org/10.1038/nrg.2016.159`.

[3] Caleb Weinreba et al. “Fundamental limits on dynamic inference from single-cell snapshots”. In: *PNAS* 115.10 (2018), E2467–E2476. DOI: `10.1073/pnas.1714723115`.

[4] Gioele La Manno et al. “RNA velocity in single cells”. In: *Nature* 560 (2018), pp. 494–498. DOI: `https://doi.org/10.1038/s41586-018-0414-6`.

[5] Cole Trapnell et al. “Pseudo-temporal ordering of individual cells reveals dynamics and regulators of cell fate decisions”. In: *Nature Biotechnology* 32 (2014), pp. 381–386. DOI: `10.1038/nbt.2859`.

[6] Manu Setty et al. “Wishbone identifies bifurcating developmental trajectories from single-cell data”. In: *Nature Biotechnology* 34 (2016), pp. 637–645. DOI: `10.1038/nbt.3569`.

[7] Robrecht Cannoodt, Wouter Saelens, and Yvan Saeys. “Computational methods for trajectory inference from single-cell transcriptomics”. In: *European journal of immunology* 46 (2016), pp. 2496–2506. DOI: `https://doi.org/10.1002/eji.201646347`.

[8] URL: `http://www.adameykolab.eu/` (visited on 05/19/2020).

[9] URL: `https://www.meduniwien.ac.at/hp/phd-neuroscience/research-laboratories/igor-adameyko/` (visited on 05/14/2020).

[10] B.Alberts et al. “Molecular Biology of the cell”. In: Garland Science, 2008, p. 345.

[11] Xavier Rambout, Franck Dequiedt, and Lynne E. Maquat. “Beyond Transcription: Roles of Transcription Factors in Pre-mRNA Splicing”. In: *Chemical Reviews* 118.8 (2018), pp. 4339–4364. DOI: `10.1021/acs.chemrev.7b00470`.

[12] Guido Walz. “Lexikon der Mathematik: Band 3”. In: Springer, 2017, p. 419.

[13] Krause and Eugene F. *Taxicab geometry : an adventure in non-Euclidean geometry*. New York, NY : Dover Publ., 1986.

[14] Hans Friedrich Eckey, Reinhold Kosfeld, and Martina Rengers. “Multivariate Statistik”. In: Gabler, 2002, p. 219.

[15] RJ Lipton and KW Regan. *Explaining The Jaccard Metric*. 2018. URL: https://rjlipton.wordpress.com/2018/12/14/explaining-the-jaccard-metric/ (visited on 05/08/2020).

[16] H. R. Schwarz. “Numerical Analysis: A Comprehensive Introduction”. In: Wiley, 1989, p. 324.

[17] Sam T. Roweis and Lawrence K. Saul. “Nonlinear Dimensionality Reduction by Locally Linear Embedding”. In: *Science* 290.5500 (2000), pp. 2323–2326. DOI: 10.1126/science.290.5500.2323.

[18] Pavel Pudil and Jana Novovičová. “Novel Methods for Feature Subset Selection with Respect to Problem Knowledge”. In: *Feature Extraction, Construction and Selection: A Data Mining Perspective*. Ed. by Huan Liu and Hiroshi Motoda. Springer, Boston, MA, 1998, pp. 101–116.

[19] URL: https://www.sciencedirect.com/topics/medicine-and-dentistry/principal-component-analysis (visited on 05/08/2020).

[20] URL: http://pklab.med.harvard.edu/cgi-bin/R/rook/nc.chromaf.cE12_E13/index.html (visited on 04/03/2020).

[21] Gunnhildur ÁstaTraustadóttir et al. “Evidence of non-canonical NOTCH signaling: Delta-like 1 homolog (DLK1) directly interacts with the NOTCH1 receptor in mammals”. In: *Cellular Signalling* 28.4 (2016), pp. 246–254. DOI: https://doi.org/10.1016/j.cellsig.2016.01.003.

[22] Wang H et al. “STAT1 activation regulates proliferation and differentiation of renal progenitors.” In: *Cellular Signalling* 22.11 (2010), pp. 1717–1726. DOI: 10.1016/j.cellsig.2010.06.012.

[23] Wouter Saelens et al. “A comparison of single-cell trajectory inference methods”. In: *Nature Biotechnology* 37 (2019), pp. 547–554. DOI: https://doi.org/10.1038/s41587-019-0071-9.

[24] Trapnell C et al. “The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells”. In: *Nature Biotechnology* 32 (2014), pp. 381–386. DOI: 10.1038/nbt.2859.

[25] Xiaojie Qiu et al. “Single-cell mRNA quantification and differential analysis with Census”. In: *Nature Methods* 14 (2017), pp. 309–315. DOI: https://doi.org/10.1038/nmeth.4150.

[26] Qiu X et al. “Reversed graph embedding resolves complex single-cell trajectories”. In: *Nature Methods* 14 (2017), pp. 979–982. DOI: 10.1038/nmeth.4402.