Technical University of Munich

Department of Mathematics

Bachelor's Thesis

Interaction Preserving Discretization Methods and Applications in Single-Cell RNA Sequencing Data Analysis

Niclas Popp

Supervisor: Prof. Dr. Dr. Fabian Theis

Advisor: Dr. Antonio Scialdone, Dr. Jonathan Fiorentino

Submission Date: 15.12.2020

I assure the single-handed composition of this bachelor's thesis only supported by declared resources.

Garching, 15.12.2020

Zusammenfassung

Diskretisierung beschreibt die Transformation einer kontinuierlichen Zufallsvariable in eine diskrete stochastische Größe. Dieser Prozess ermöglicht die Anwendung allgemeiner informationstheoretischer Methoden auf kontinuierlichen experimentellen Daten. Bei der Diskretisierung von hochdimensionalen Datensätzen ist es wichtig, strukturelle Interaktionen innerhalb der Daten zu erhalten. Im Rahmen dieser Arbeit werden Ähnlichkeitsmaße für Wahrscheinlichkeitsverteilungen eingeführt, um derartige Interaktionsstrukturen zu quantifizieren und verschiedene Diskretisierungsschemata bezüglich ihrer Fähigkeiten, diese zu erhalten, verglichen. Darauf aufbauend werden Vorteile interaktionserhaltender Diskretisierungsverfahren anhand von zwei Anwendungen in der Analyse von Einzelzell-RNA-Sequenzierungsdaten dargestellt.

Summary

Discretization is the process of transforming a continuous random variable into a discrete one. This procedure enables the application of general information theoretic methods on continuous experimental data. When discretizing high-dimensional datasets it is important to preserve structural interactions within the data. In this thesis we will introduce measures for distribution similarity to capture such interactions and examine different discretization schemes regarding their capabilities to preserve them. Subsequently, we exemplify advantages of interaction preserving discretization on two applications in single-cell RNA sequencing data analysis.

Contents

1 Introduction

Discretization is the process of transforming a continuous random variable into a discrete one. Apart from the fact that some data mining methods require discrete data, discretization can provide several advantages such as noise reduction and improvements in accuracy and speed for classification algorithms [1]. When discretizing high-dimensional sets the goal should be to retain as much of the original interactions within the data as possible. Although many real-world applications require multivariate discretization, *Interaction Preserving Discretization* is very much an open research problem.
In the first part of the thesis we will introduce measures for distribution similarity that can be used to quantify interactions between the dimensions of empirical datasets. Subsequently, discretization schemes, of which some are designed to preserve specific types of interaction, will be presented. The algorithms we will be looking at are Uniform Width Discretization (UW), Distance Based Clustering, Bayesian Blocks (BB) [2], Correlation Preserving Discretization (CPD) [3] and Interaction Preserving Discretization (IPD) [4].

A current field of research that can benefit from improvements in discretization methods is the analysis of single-cell RNA sequencing data. Ribonucleic acid (RNA) molecules play a fundamental role in various metabolic processes such as protein synthesis and can be involved in the transmission of genetic information. Analysing the transcriptome, the entirety of RNA in a cell, yields insight into cellular functions in different developmental stages or physiological conditions. This allows for a characterisation of a cell's identity. RNA sequencing measures the type and quantity of RNA molecules present in a biological sample at a fixed timepoint. The introduction of methods for capturing gene expression on the level of individual cells [5] represented a major breakthrough since previous data was limited to averages across all cells in the analysed population. Thus, single-cell RNA sequencing enables the investigation of cellular phenomena that had previously been very hard or even impossible to address, such as early development where there are naturally only few cells.
In the second part of the thesis we will benchmark our open-source implementations of the discretization schemes named above on single-cell RNA sequencing data and use the results for two applications, namely Critical Variable Selection (CVS) [6] and a gene regulatory network inference algorithm based on partial information decomposition (PIDC) [7]. The PIDC algorithm uses information theoretic measures for discrete random variables to identify regulatory relationships between genes. Critical Variable Selection aims to extract relevant variables in complex systems. In particular, CVS has been used to identify subsets of most informative sites in protein sequences [6]. Our goal is to apply this method on single-cell RNA sequencing data in order to determine subsets of relevant genes.

The structure of the thesis is as follows. For better readability all necessary theoretical foundations are introduced in chapter 2. The main discretization methods are discussed in chapter 3. In chapter 4 we benchmark the algorithms on representative single-cell RNA sequencing datasets. These results are applied to CVS and the PIDC inference scheme in chapter 5. We conclude with a comparison of the different discretization methods and a summary of the results.

2 Foundations

2.1 General Notions

In the following we consider a database $\mathbf{D}$ consisting of n datapoints in m dimensions. The set of dimensions is denoted as $\mathbf{A} = \{X_1, ..., X_m\}$ and each X_i is interpreted as a continuous random variable with domain $[\min_i, \max_i]$. The joint space is given by $\Omega = [\min_1, \max_1] \times ... \times [\min_m, \max_m]$. The probability density function of all points projected onto a certain X_i is written as $p(X_i)$ with corresponding expectation $E[X_i]$ and variance $Var[X_i]$. The covariance between X_i and X_j is denoted as $Cov(X_i, X_j)$.
The goal is to partition X_i into k_i bins and find a set of cutpoints $K_i = \{c_i^1, ..., c_i^l\}$. If we only use connected bins then $l = k_i - 1$ and the domain of X_i is partitioned into $\{[\min_i, c_i^1], (c_i^1, c_i^2], ..., (c_i^{k_i-1}, \max_i]\}$. The set of possible discretization models will be denoted by $\mathcal{M}$. The data discretized by $M \in \mathcal{M}$ is $M(\mathbf{D})$.
For clarity of notation $\log(\cdot)$ is the logarithm with base 2 while $\ln(\cdot)$ is the natural logarithm. Vectors and databases that contain empirical data will be denoted in bold.

2.2 Probabilistic Measures

In order to compare the results of different discretization methods, it is important to examine informative ways to quantify interaction in $\mathbf{D}$. Therefore, we start by introducing probabilistic measures for this setup.

Definition 2.1. The *Pearson Correlation* between two random variables X and Y is

$$Cor(X, Y) = \frac{Cov(X, Y)}{Var(X)Var(Y)} \tag{1}$$

For two random vectors X and Y the correlation matrix C is given by

$$c_{i,j} = Cor(X_i, Y_j) \tag{2}$$

Pearson Correlation detects linear relations between random variables. The Spearman Rank Correlation coefficient is able to assess more general monotonic dependencies in empirical data.

Definition 2.2. Assume $\mathbf{X} = (x_1, ..., x_n)$ and $\mathbf{Y} = (y_1, ..., y_n)$ are two data vectors. The *rank* $r(x_i)$ is the position of x_i when sorting X in descending order. We define

$$r(\mathbf{X}) := (r(x_1), ..., r(x_n)) \tag{3}$$

The *Spearman Rank Correlation* between $\mathbf{X}$ and $\mathbf{Y}$ is given by

$$\rho_{\mathbf{X},\mathbf{Y}} = Cor(r(\mathbf{X}), r(\mathbf{Y})) \tag{4}$$

Due to spurious relationships, correlation is not sufficient to describe relations within a dataset. Thus, we will use additional quantities to analyse similarity between dimensions of $\mathbf{D}$.

Definition 2.3. Assume $p(\cdot)$ and $q(\cdot)$ are two probability densities over the same probability space $\mathcal{X}$. The *Kullback-Leibler divergence*, also known as *relative entropy*, is defined as

$$KL(p||q) = \int_{\mathcal{X}} p(x)\log\Big(\frac{p(x)}{q(x)}\Big)dx \tag{5}$$

Theorem 2.4 justifies the definition of $KL(p||q)$ as divergence measure.

Theorem 2.4. *$KL(p||q) \geq 0$ with equality if and only if $p(\cdot) = q(\cdot)$ almost everywhere.*

Proof. Define the support set of $p(\cdot)$ as $\mathcal{A} = \{x \in \mathcal{X} : p(x) > 0\}$

$$\begin{aligned}
KL(p||q) &= \int_{\mathcal{X}} p(x)\log\Big(\frac{p(x)}{q(x)}\Big)dx \\
&= \int_{\mathcal{A}} p(x)\log\Big(\frac{p(x)}{q(x)}\Big)dx \\
&= -\int_{\mathcal{A}} p(x)\log\Big(\frac{q(x)}{p(x)}\Big)dx \\
&\overset{(I)}{\geq} -\log\int_{\mathcal{A}} p(x)\frac{q(x)}{p(x)}dx \\
&= -\log\int_{\mathcal{A}} q(x)dx \\
&\overset{(II)}{\geq} -\log\int_{\mathcal{X}} q(x)dx \\
&= -\log(1) = 0
\end{aligned} \tag{6}$$

(I) follows from Jensen's inequality and (II) from the definition of $\mathcal{A}$. □

The Kullback-Leibler divergence is bounded by 0 from below but there does not exist any upper bound i.e. $KL(p||q) \in [0,\infty)$. The Jensen-Shannon divergence is an alternative divergence measure that is symmetric and can be bounded by 1 from above as shown by J. Lin [8].

Definition 2.5. Given two probability measures with densities $p(\cdot)$ and $q(\cdot)$ over the same probability space $\mathcal{X}$ the *Jensen-Shannon divergence* is introduced as

$$JS(p||q) = \frac{1}{2}KL(p||m) + \frac{1}{2}KL(q||m) \tag{7}$$

where $m = \frac{1}{2}(p+q)$

The Kullback-Leibler and Jensen-Shannon divergences require knowledge about the probability density functions. In practice this is not always available for empirical datasets or can be hard to estimate. Nguyen et. al. [9] introduced the Cumulative Jensen Shannon distance which can be applied to empirical data.

Definition 2.6. The *Cumulative Jensen Shannon distance* (CJS) of $p(\cdot)$ and $q(\cdot)$ is

$$CJS(p||q) = \int_{\mathcal{X}} P(x)\log\Big(\frac{2\,P(x)}{P(x)+Q(x)}\Big)dx + \frac{1}{2\ln 2}\int_{\mathcal{X}}\Big(Q(x)-P(x)\Big)dx \tag{8}$$

where $P(\cdot)$ and $Q(\cdot)$ are the cumulative distribution functions corresponding to $p(\cdot)$ and $q(\cdot)$.

Induced by the following lemma CJS defines a divergence measure.

Lemma 2.7. *$CJS(p||q) \geq 0$ with equality if and only if $p(\cdot) = q(\cdot)$ almost everywhere.*

Proof.

$$\begin{aligned}\int_{\mathcal{X}} P(x)\log\Big(\frac{2\,P(x)}{P(x)+Q(x)}\Big)dx &\overset{(I)}{\geq} \log\Big(\frac{2\int_{\mathcal{X}} P(x)dx}{\int_{\mathcal{X}} P(x)+Q(x)dx}\Big)\int_{\mathcal{X}} P(x)dx \\ &\overset{(II)}{\geq} \frac{1}{2\ln 2}\int_{\mathcal{X}}\Big(P(x)-Q(x)\Big)dx\end{aligned} \tag{9}$$

(I) holds because of the log sum inequality. In (II) it was used that

$$a\log\Big(\frac{a}{b}\Big) \geq \frac{1}{2\ln 2}(a-b) \tag{10}$$

□

CJS is not symmetric by definition but can be symmetrized by considering

$$CJS_{sym}(p||q) = \frac{1}{2}\Big(CJS(p||q) + CJS(q||p)\Big) \tag{11}$$

All distribution similarity measures introduced so far capture *pairwise* interactions. The Total Correlation, also known as Multi-Information, is a commonly used measure of multivariate correlation.

Definition 2.8. Given a set of random variables $\{X_i\}_{i=1}^n$ the *Total Correlation* is

$$C_T(X_1,...,X_n) = KL(p(X_1,...,X_n)||p(X_1)...p(X_n)) \tag{12}$$

2.3 Information Theoretic Measures

Discretization can be seen as a way of compressing and encoding data. Hence several discretization methods such as IPD [4] and others [10] make use of information theoretic concepts. Apart from that, the PIDC algorithm [7] for inferring gene regulatory networks and Critical Variable Selection [6] which will be applied in chapter 5 are based on information theory. In this section the information theoretic measures that are necessary for these purposes will be introduced.

Definition 2.9. For a discrete random variable X with probability mass function $p(\cdot)$ the *entropy* is defined as

$$H(X) = \sum_{x\in X} p(x)\log(p(x)) \tag{13}$$

Entropy quantifies the uncertainty in the probability distribution of a single random variable. When analysing the interaction between two random variables the mutual information provides a non-negative and symmetric measure for the statistical dependency.

Definition 2.10. Let X and Y be two discrete random variables. The *mutual information* is given by

$$I(X;Y) = H(X) + H(Y) - H(X,Y) = \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(\frac{p(x,y)}{p(x)p(y)}\Big) \tag{14}$$

The *conditional mutual information* of X and Y given a third discrete random variable Z is

$$I(X;Y|Z) = H(X,Z) + H(Y,Z) - H(X,Y,Z) - H(Z) \tag{15}$$

Aside from conditional mutual information it is difficult to define multivariate measures that quantify statistical dependencies. A first attempt was made by McGill [11].

Definition 2.11. For three discrete random variables X, Y and Z the *McGill interaction distance* is defined as

$$\begin{aligned} II(X;Y;Z) &= I(X;Y|Z) - I(X;Y) \\ &= I(X;Z|Y) - I(X;Z) \\ &= I(Y;Z|X) - I(Y;Z) \end{aligned} \tag{16}$$

Timme et al. [12] pointed out that the McGill interaction distance can be negative and even zero in some non-trivial cases. William and Beer [13] expanded the concept and introduced the partial information decomposition. The PIDC algorithm makes use of this measure in the case of three random variables.

Definition 2.12. Given three discrete random variables X, Y and Z consider the set of source variables $S = \{X, Y\}$ and a target variable Z.
The *specific information* is a measure for the amount of information that can be determined from one random variable about another one.

$$I_{spec}(z;X) = \sum_{x\in X} p(x|z)\bigg(\log\Big(\frac{1}{p(z)}\Big) - \log\Big(\frac{1}{p(z|x)}\Big)\bigg) \tag{17}$$

The *redundancy* $R(Z;X,Y)$ is the portion of information provided by Z that is given by either component of S.

$$R(Z;X,Y) = \sum_{z\in Z} p(z)\min_{S} I_{spec}(z;S) \tag{18}$$

$U_Y(Z;X)$ is the *unique information* of Z provided by Y.

$$U_Y(Z;X) = I(X;Z) - R(Z;X,Y) \tag{19}$$

The information from Z that can only be determined by X and Y simultaneously is called *synergy*.

$$S(Z;X,Y) = II(X;Y;Z) + R(Z;X,Y) \tag{20}$$

The *partial information decomposition* (PIDC) is

$$I(Z;X,Y) = S(Z;X,Y) + U_Y(Z;X) + U_X(Z;Y) + R(Z;X,Y) \tag{21}$$

2.4 Shannon Source Coding Theorem

In this section we briefly explain the Shannon source coding theorem that is used to construct the IPD algorithm [4] in chapter 3.2.2.
The setup for this theorem is a discrete random variable X that we aim to encode using an alphabet consisting of D letters. Such alphabet is called *D-ary*. In the following we will set $D = 2$.

Definition 2.13. A *source code* C is a mapping from $\mathcal{X}$, the range of a discrete random variable X, to the set of finite-length strings from a D-ary alphabet.
$C(x)$ is the *codeword* corresponding to $x \in \mathcal{X}$ with respective length $l(x) \in \mathbb{N}$. The *expected description length* of a source code is

$$L(C) = \sum_{x \in \mathcal{X}} p(x) l(x) \tag{22}$$

The goal is to find a bound on the optimal description length when encoding X.

Lemma 2.14. (Shannon source coding theorem) *For a discrete random variable X with range $\mathcal{X}$ the minimal expected description length $L^\star$ is bounded by*

$$H(X) \leq L^\star \tag{23}$$

Proof. The presented proof will closely follow [14].

$$\begin{aligned} L^\star - H(X) &= \sum_{x \in \mathcal{X}} p(x) l(x) - \sum_{x \in \mathcal{X}} p(x) \log\Big(\frac{1}{p(x)}\Big) \\ &= -\sum_{x \in \mathcal{X}} p(x) \log(2^{-l(x)}) + \sum_{x \in \mathcal{X}} p(x) \log(p(x)) \end{aligned} \tag{24}$$

Introducing $q(x) = \frac{2^{-l(x)}}{\sum_{x \in \mathcal{X}} 2^{-l(x)}}$ and $c = \sum_{x \in \mathcal{X}} 2^{-l(x)}$ we obtain

$$\begin{aligned} L^\star - H(X) &= \sum_{x \in \mathcal{X}} p(x) \log\Big(\frac{q(x)}{p(x)}\Big) - \log(c) \\ &= KL(p||q) + \log\Big(\frac{1}{c}\Big) \\ &\overset{(I)}{\geq} 0 \end{aligned} \tag{25}$$

In (I) we used theorem 2.4 and $c \leq 1$ which is a consequence of the Kraft inequality shown in [14]. □

As a consequence of Shannon's source coding theorem, the entropy is a lower bound for the smallest codeword length that is theoretically possible for a given alphabet with associated weights.

2.5 Interaction Distance

Nguyen et al. [4] introduce another interaction distance ID that is used to measure similarity between two multivariate distributions. The IPD method utilises ID to approximate multivariate heterogeneity within bins of a discretization model.

Definition 2.15. The *interaction distance* ID between $p(\mathbf{A})$ and $q(\mathbf{A})$ is defined as

$$ID(p(\mathbf{A})||q(\mathbf{A})) = \sqrt{\int_{\Omega} \big(P(\mathbf{a}) - Q(\mathbf{a})\big)^2 d\mathbf{a}} \tag{26}$$

where

$$P(\mathbf{a}) = \int_{min_1}^{a_1} ... \int_{min_m}^{a_m} p(x_1, ..., x_m) dx_1 ... dx_m \tag{27}$$

and $Q(\mathbf{a})$ similarly.

Theorem 2.16. *The interaction distance ID defines a distance metric:*

(i) $ID(p(\boldsymbol{A})||q(\boldsymbol{A})) = ID(q(\boldsymbol{A})||p(\boldsymbol{A}))$

(ii) $ID(p(\boldsymbol{A})||q(\boldsymbol{A})) \geq 0$ *with equality if and only if* $p(\boldsymbol{A}) = q(\boldsymbol{A})$

(iii) $ID(p(\boldsymbol{A})||r(\boldsymbol{A})) + ID(r(\boldsymbol{A})||q(\boldsymbol{A})) \geq ID(p(\boldsymbol{A})||q(\boldsymbol{A}))$

Proof. (i) and (ii) follow directly from the definition of the interaction distance 2.15. (iii): Define $H(\mathbf{A}) = P(\mathbf{A}) - R(\mathbf{A})$ and $G(\mathbf{A}) = R(\mathbf{A}) - Q(\mathbf{A})$. The inequality can be written as:

$$\sqrt{\int_{\Omega} H(\mathbf{a})^2 d\mathbf{a}} + \sqrt{\int_{\Omega} G(\mathbf{a})^2 d\mathbf{a}} \geq \sqrt{\int_{\Omega} \big(H(\mathbf{a}) + G(\mathbf{a})\big)^2 d\mathbf{a}} \tag{28}$$

which is a consequence of Hölder's inequality:

$$\sqrt{\int_{\Omega} H(\mathbf{a})^2 \cdot \int_{\Omega} G(\mathbf{a})^2 d\mathbf{a}} \geq \int_{\Omega} H(\mathbf{a}) d\mathbf{a} \cdot \int_{\Omega} G(\mathbf{a}) d\mathbf{a} \tag{29}$$

□

In theory the interaction distance is defined using probability density functions. In our scenario we aim to discretize empirical measurements for which the correct distributions are unknown. The following theorem shows how the interaction distance can be utilized in this setting.

Theorem 2.17. *Assume $p(\boldsymbol{A})$ is formed by empirical data $\{R_1, .., R_k\}$ and $q(\boldsymbol{A})$ respectively by $\{S_1, ..., S_l\}$. Then the interaction distance is given by*

$$\begin{aligned} ID(p(\boldsymbol{A})||q(\boldsymbol{A})) = \Big(& \frac{1}{k^2} \sum_{j_1}^{k} \sum_{j_2}^{k} \prod_{i=1}^{m} (\max_i - \max\{R_{j_1}^i, R_{j_2}^i\}) \\ & - \frac{2}{kl} \sum_{j_1}^{k} \sum_{j_2}^{l} \prod_{i=1}^{m} (\max_i - \max\{R_{j_1}^i, S_{j_2}^i\}) \\ & + \frac{1}{l^2} \sum_{j_1}^{l} \sum_{j_2}^{l} \prod_{i=1}^{m} (\max_i - \max\{S_{j_1}^i, S_{j_2}^i\}) \Big)^{\frac{1}{2}} \end{aligned} \tag{30}$$

Proof. For empirical data it holds:

$$P(a) = \frac{1}{k}\sum_{j=1}^{k}\prod_{i=1}^{m}\mathbb{1}_{\{R_j^i \leq a_i\}} \qquad Q(a) = \frac{1}{l}\sum_{j=1}^{l}\prod_{i=1}^{m}\mathbb{1}_{\{S_j^i \leq a_i\}} \tag{31}$$

Consequently $[ID(p(\mathbf{A})||q(\mathbf{A}))]^2$ is given by

$$\int_{\min_1}^{\max_1}\dots\int_{\min_m}^{\max_m}\Big(\frac{1}{k}\sum_{j=1}^{k}\prod_{i=1}^{m}\mathbb{1}_{\{R_j^i \leq a_i\}} - \frac{1}{l}\sum_{j=1}^{l}\prod_{i=1}^{m}\mathbb{1}_{\{S_j^i \leq a_i\}}\Big)^2 da_1\dots da_m \tag{32}$$

By expanding this equation and using the additivity of the integral it is possible to derive the result

$$\begin{aligned}
ID(p(\mathbf{A})||q(\mathbf{A})) = \Big(&\frac{1}{k^2}\sum_{j_1}^{k}\sum_{j_2}^{k}\prod_{i=1}^{m}\int_{\min_1}^{\max_1}\mathbb{1}_{\max\{R_{j_1}^i, R_{j_2}^i\}\leq a_i}da_i \\
&- \frac{2}{kl}\sum_{j_1}^{k}\sum_{j_2}^{l}\prod_{i=1}^{m}\int_{\min_1}^{\max_1}\mathbb{1}_{\max\{R_{j_1}^i, S_{j_2}^i\}\leq a_i}da_i \\
&+ \frac{1}{l^2}\sum_{j_1}^{l}\sum_{j_2}^{l}\prod_{i=1}^{m}\int_{\min_1}^{\max_1}\mathbb{1}_{\max\{S_{j_1}^i, S_{j_2}^i\}\leq a_i}da_i\Big)^{\frac{1}{2}} \\
= \Big(&\frac{1}{k^2}\sum_{j_1}^{k}\sum_{j_2}^{k}\prod_{i=1}^{m}(\max_i - \max\{R_{j_1}^i, R_{j_2}^i\}) \\
&- \frac{2}{kl}\sum_{j_1}^{k}\sum_{j_2}^{l}\prod_{i=1}^{m}(\max_i - \max\{R_{j_1}^i, S_{j_2}^i\}) \\
&+ \frac{1}{l^2}\sum_{j_1}^{l}\sum_{j_2}^{l}\prod_{i=1}^{m}(\max_i - \max\{S_{j_1}^i, S_{j_2}^i\})\Big)^{\frac{1}{2}}
\end{aligned} \tag{33}$$

□

2.6 Principal Component Analysis

The CPD algorithm [3] makes use of the concept of *Principal Component Analysis* (PCA) [15] which is frequently used in modern data analysis. PCA is a linear method that reduces the dimensionality of a dataset by translating each datapoint into a representation that captures the "most important" features. For the method of principal components, "most important" is interpreted as the direction of largest variability.

Definition 2.18. Given a random variable X in $\mathbb{R}^m$ with covariance matrix Σ the *principal components* of X are the eigenvectors of Σ.

The covariance matrix Σ of X is symmetric and positive semi-definite. Thus, definition 2.18 is justified by the Spectral theorem.

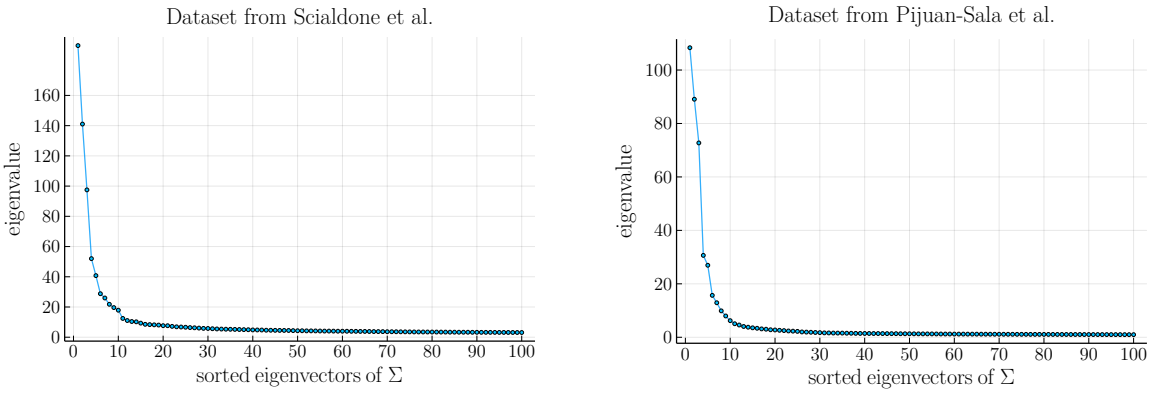


Figure 1: Scree plots for datasets [25] and [28]. The eigenvectors are sorted by descending eigenvalues. In both cases we choose the first $k = 10$ principle components.

Given an orthonormal basis $v_1, ..., v_m$ of eigenvectors of Σ it is possible to select the ones that describe the highest portion of variability. In practice this is done by choosing the eigenvectors with the $k < m$ largest eigenvalues. The selection process of k is non-trivial and a number of different approaches have been proposed [15]. For the results presented in chapter 4 we chose to analyse the Scree plots displayed in figure 1. To obtain the lower dimensional representation of the datapoints, they are projected onto the subspace spanned by these k principal components. In order to perform robust PCA it is useful to normalize the data to prevent influences from the specific scales of the variables.

2.7 Relevance and Resolution

It is impossible to consider all variables when modelling a complex system since some of them are in fact unknown or cannot be measured. Critical Variable Selection [6] provides an information theoretic framework for extracting the most informative variables of a model.

In our setup we consider data composed of M sequences with length L. Given a subset $I \subseteq \{1, ..., L\}$ we split $\vec{s}^{\,\alpha} = (\underline{s}_I^\alpha, \overline{s}_I^\alpha)$ into two subsequences $\underline{s}_I^\alpha = \{a_i^\alpha : i \in I\}$ and $\overline{s}_I^\alpha = \{a_i^\alpha : i \notin I\}$. The frequency of a subsequence $\underline{s}$ in the sample $\vec{s}^{\,\alpha}$ is

$$k_I(\underline{s}) = \sum_{\alpha=1}^{L} \delta_{\underline{s},\underline{s}_I^\alpha} \tag{34}$$

Marsili et al. [16] observe that a broader distribution of frequencies captures more information of the function being optimized. Hence, they suggest to use the entropy of the frequency distribution as a quantitatve measure for *relevance*.

Definition 2.19. The entropy of the frequency distribution is given by

$$H(K_I) = -\sum_k \frac{km_I(k)}{M} \log\Big(\frac{km_I(k)}{M}\Big) \tag{35}$$

where $m_I(k)$ is the number of subsequences that occur k times.

$$m_I(k) = \sum_{\underline{s}} \delta_{k,k_I(\underline{s})} \tag{36}$$

$H(K_I)$ is different from the entropy $H(\underline{s}_I)$ of a sequence $\underline{s}_I$. $H(\underline{s}_I)$ measures *resolution*.

Definition 2.20. The entropy of a sequence $\underline{s}_I$ is

$$H(\underline{s}_I) = -\sum_{\underline{s}} \frac{k_I(\underline{s})}{M} \log\Big(\frac{k_I(\underline{s})}{M}\Big) = -\sum_{k} \frac{k m_I(k)}{M} \log\Big(\frac{k}{M}\Big) \tag{37}$$

It is possible to derive a theoretical bound on $H(K_I)$ as a function of $H(\underline{s}_I)$. An upper constraint on the feasible region is given by the *Data Processing Inequality* [14].

$$H(K_I) \leq H(\underline{s}_I) \tag{38}$$

For a fixed $H(\underline{s}_I) = \tilde{H}$ we are interested in the most informative sample that yields

$$\mathbf{m}^* = \underset{m_I(k):\, \hat{H}(K_I) \leq \tilde{H}}{\operatorname{argmax}} \hat{H}(K_I) \tag{39}$$

subject to the constraint $\sum_{k=1}^{M} k m_I(k) = M$ with $\mathbf{m} = \{m_I(k), k > 0\}$. The solution can be found by maximizing

$$\hat{H}(K_I) + \mu \hat{H}(\underline{s}_I) + \lambda \sum_{k>1} k m_I(k) \tag{40}$$

over $m_I(k) \in \mathbb{R}^+$ where μ and λ are Lagrange multipliers used to enforce the constraints $\hat{H}(\underline{s}_I) = \tilde{H}$ and $\sum_{k=1}^{M} k m_I(k) = M$. The solution is

$$m_k^* = c k^{-1-\mu} \tag{41}$$

where c is a normalizing constant. As μ varies we obtain the upper bound on $H(K_I)$ shown in figure 11.

3 Discretization Algorithms

Discretization methods can be classified according to different criteria. A discretization algorithm can be *univariate* or *multivariate*. Univariate schemes discretize each $X_i \in A$ independently and do not take into account the distributions of $\mathbf{A} \setminus \{X_i\}$. Multivariate procedures regard the distribution of the entire dataset $\mathbf{D}$. A further distinction can be drawn between *global* discretizers wich operate on all $X_i \in \mathbf{A}$ simultaneously and *local* discretizers which edit only one dimension at a time.
Discretization methods used in single-cell RNA sequencing data analysis should be *unsupervised*, meaning that the user does not have to specify any parameters for the algorithm to function. *Supervised* procedures can be adjusted better to specific use cases but require user input.

In the following sections we will present three univariate and two multivariate schemes. Due to their complexity, some multivariate algorithms can be computationally expensive and slow on large, high-dimensional datasets. Thus, we introduce a novel method based on hierarchical clustering to address this problem.

3.1 Univariate Discretization

Univariate methods do not have any specific properties to preserve multivariate interactions. However, they are likely to have lower computational complexity and thus can be faster on large datasets. Furthermore, both multivariate methods presented in chapter 3.2 use univariate algorithms as part of their procedure.

3.1.1 Uniform Width

Uniform Width discretization is the simplest method for binning a univariate dataset. The domain of $X_i \in \mathbf{A}$ is split into k_i intervals of equal length where k_i is the pre-defined number of bins. The datapoints are classified according to their location in one of these bins.
While a larger number of intervals is likely to capture the distribution in more detail, Reshef et al. [17] have shown that too many bins can introduce artificial correlation. The following formulas are commonly used for selecting appropriate k_i.

k-proportional choice is the most common way to determine the number of bins. It does not take the data distribution into account, but it is unlikely to overestimate the number of bins. This choice is particularly suitable for larger datasets due to its simplicity and speed.

$$k_i = \lfloor \sqrt{n} \rfloor \tag{42}$$

Rice rule is a simple alternative to the k-proportional choice that works better for lower dimensional data.

$$k_i = \lceil 2\sqrt[3]{n} \rceil \tag{43}$$

Sturges' formula [18] assumes normal data and is specifcally developed for larger datasets.

$$k_i = \lceil \log(n) \rceil + 1 \tag{44}$$

Doane's rule [19] is an expansion of Sturges' formula. It drops the assumption of normal distributed data and takes into account the specific distribution of X_i.

$$k_i = 1 + \log(n) + \log\Big(1 + |s_i|\frac{(n+1)(n+3)}{6(n-2)}\Big) \tag{45}$$

where s_i estimates the skewness of X_i.

3.1.2 Bayesian Blocks

Bayesian Blocks is an univariate and unsupervised discretization method introduced by Scargle et al. [2] that simultaneously determines the number and position of cutpoints. The goal of this algorithm is to find the discretization that best models the distribution of the data when assuming a constant model for each bin. For this purpose, a block fitness function is determined by a Bayesian approach.
Assume a discretization model $M \in \mathcal{M}$ with bins $B_1, ..., B_n$ where n_j denotes the number of datapoints in B_j and w_j is the binwidth. With reference to Cash [21] the following maximum log-likelihood for a single bin is derived

$$\ln(L_{max}(B_j)) = n_j(\ln(n_j) - \ln(w_j)) - n_j \tag{46}$$

For single-cell RNA sequencing data Chan et al. [7] propose the following prior for the number of bins k

$$p(k) = \exp(4 - \ln(3.6765\ k^{-0.478})) \tag{47}$$

The fitness of a discretization model M for a data vector $\mathbf{X}$ consisting of blocks $B_1, ..., B_k$ is defined as the logarithm of the posterior

$$F[M(X)] = \sum_{j=1}^{k} \ln(L_{max}(B_j)) + \ln(p(k)) \tag{48}$$

In practice a dynamical programming approach is used to find the optimal binning. At the start of the algorithm the data is partitioned into intervals containing single datapoints. Consecutive bins are subsequently merged to find the discretization maximizing the overall fitness function. In chapter 4 we use the implementation contained in the Julia package Discretizers.jl [36] with complexity $\mathcal{O}(n^2)$ [2].

3.1.3 Distance Based Clustering

Clustering algorithms are intentionally designed to identify groupings in higher dimensional data. In chapter 3.2 we will use this to full extent. However, in a univariate setting distance based clustering schemes can be used to discretize data.
Given a data vector $\mathbf{X} = (x_1, ..., x_n)$ and a fixed number of clusters k the goal is to find a partition $B_1 \cup B_2 \cup ... \cup B_k$ of the range of $\mathbf{X}$ with bin centers $\mu_1, .., \mu_k$ that minimizes

$$J(\{B_1, ..., B_k\}, \{\mu_1, .., \mu_k\}) = \sum_{i=1}^{k} \sum_{x_l \in B_i} dist(x_l - \mu_i) \tag{49}$$

In practice it is necessary to choose a distance measure for equation 49. In chapter 4 we will examine the Euclidean distance and the Pearson Correlation distance.
In general it is NP-hard to find the optimal clustering. Thus, various iterative methods have been proposed to approximate solutions [22]. For its simplicity and speed we choose to use *k-means* clustering. The pseudo code is given as Algorithm 1.
Since there are only $\mathcal{O}(n^k)$ possible partitions of points, the k-means algorithm converges in a finite number of steps. At convergence the cluster centers $\hat{\mu_1}, .., \hat{\mu_k}$ are local minima of equation 49.
Nevertheless, the speed of k-means comes at a cost. The final clustering is sensitive to the initialization of $\mu_1, .., \mu_k$. Finding a robust procedure for this problem is non-trivial. Different approaches are shown in [22].

Algorithm 1 One-dimensional k-means

Require: Data vector $\mathbf{X}$ consisting of n datapoints, number of clusters k
1: Initialize $\mu_1, .., \mu_k$
2: **while** $\{B_1, ..., B_k\}, \{\mu_1, .., \mu_k\}$ change **do**
3: Assign each point to the closest cluster center

$$B_j = \{x \in \mathbf{D} : j = \underset{l=1,...,k}{\operatorname{argmin}} dist(x - \mu_l)\}$$

4: Update the cluster centers

$$\mu_j = \frac{1}{|B_j|} \sum_{x \in B_j} x$$

5: **end while**
6: **return** $\{B_1, ..., B_k\}, \{\mu_1, .., \mu_k\}$

3.2 Multivariate Discretization

In contrast to the univariate methods introduced above, multivariate discretization algorithms take the entire distribution structure in $\mathbf{D}$ into account. In this section we present two procedures based on selection criteria that are targeted towards retaining specific multivariate interactions.

3.2.1 Correlation Preserving Discretization

Mehta et al. [3] propose a discretization scheme called *Correlation Preserving Discretization* (CPD) which is designed to preserve the structure of the Pearson Correlation matrix. The algorithm expands on the concepts of principal component analysis and distance based clustering. No external information other than the number of bins for the clustering step and the number of principal components is required. Since the cutpoints are determined simultaneously for all dimensions the algorithm is global.
The intuition behind the procedure is to project all datapoints in the subspace spanned by the principal components. This basis is orthogonal and the eigenvectors are linearly uncorrelated. Therefore, independent discretization in this representation does not disturb the correlation structure.

The exact pseudocode for the algorithm is given as Algorithm 2. Figure 2 displays a two-dimensional example using k-means as distance based clustering scheme.

Algorithm 2 CPD

Require: Database **D** consisting of n datapoints in m dimensions.

1: Normalize and mean centralize for each dimension.
2: Calculate the $m \times m$ correlation Matrix C
3: Determine the principal components as eigenvectors of C
4: Select $k < m$ eigenvectors $v_1, ..., v_k$ as principal components
5: Project the datapoints onto the subspace $\mathcal{S} = \langle \{v_1, ..., v_k\} \rangle$
6: Independently discretize each dimension of $\mathcal{S}$ by using distance based clustering
7: Determine all cutpoints $\tilde{c}^1_{v_1}, ..., \tilde{c}^{N_1}_{v_1}, \tilde{c}^1_{v_2}, ..., \tilde{c}^{N_k}_{v_k}$ in the entire subspace $\mathcal{S}$
8: Reproject $\tilde{c}^1_{v_1}, ..., \tilde{c}^{N_1}_{v_1}, \tilde{c}^1_{v_2}, ..., \tilde{c}^{N_k}_{v_k}$ onto the original dimensions to obtain the final binedges
9: **return** Cutpoints $c^i_{1,1}, ..., c^i_{1,N_1}, c^i_{2,1}, ..., c^i_{k,N_k}$ for each X_i.

Lemma 3.1. *The CPD algorithm has the following properties:*

(i) The discretization in step 6 preserves correlations of order 2.

(ii) When using distance based clustering with negligible computational complexity, the cost can be bounded by $\max\{\mathcal{O}(m^2 \cdot n), \mathcal{O}(m^3)\}$.

Proof. (i) As discussed in chapter 2.6 the eigenvectors of the correlation matrix form an orthogonal basis. Thus, all correlations of order 2 between the principal components vanish. Consequently, independent discretization along each basis vector does retain the correlation structure.
(ii) The computational complexity of the algorithm can be determined by the complexity of the dominating steps. The computation of the correlation matrix costs $\mathcal{O}(m^2 \cdot n)$ and the determination of the principal components costs $\mathcal{O}(m^3)$. However, in case one uses a distance based clustering scheme with non-negligible complexity, the number of bins per principal component k must be taken into account as well. □

3.2.2 Interaction Preserving Discretization

CPD is specifically constructed to preserve linear correlations. As discussed in chapter 2 correlation, is not the only interaction structure within datasets that should be retained. Therefore, CPD might miss interactions that cannot be captured by the Pearson correlation matrix. Nguyen et al. [4] introduce an unsupervised, local discretization approach called *Interaction Preserving Discretization* (IPD) that is designed to capture more general interactions.
The setup for the algorithm is as follows. Each dimension $X_i \in \mathbf{A}$ is partitioned into T_i fine-grained *micro bins* using univariate discretization methods. These micro bins are merged into k_i *macro bins* $B^1_i, ..., B^{k_i}_i$ that form the final discretizer.

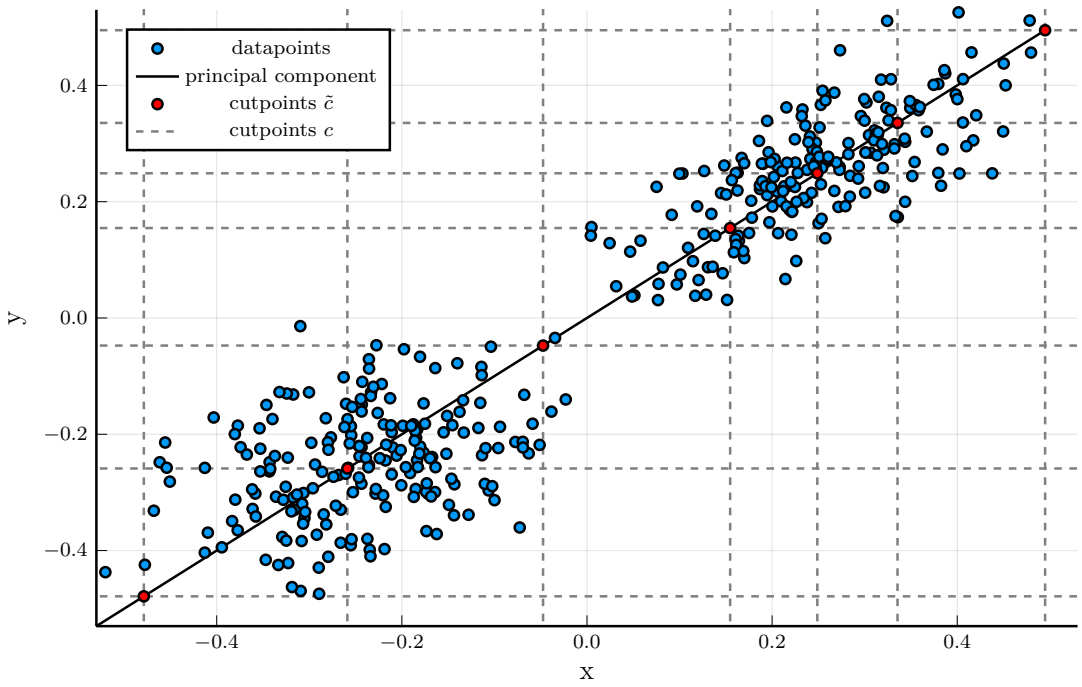


Figure 2: In the two dimensional case there exist only two principal components, hence $k = 1$. The datapoints are projected onto this vector and 7 cutpoints $\tilde{c}^1_{v_1}, ..., \tilde{c}^7_{v_1}$ are determined using k-means. The final cutpoints $c^1_{1,1}, ..., c^1_{1,7}, c^2_{1,1}, ..., c^2_{1,7}$ are displayed as dashed lines.

The model selection criterion for IPD is based around the *Minimum Description Length Principle* (MDL) which was proposed by Rissanen [23]. Given a database $\mathbf{D}$ the goal of the algorithm is to find the discretization model $M \in \mathcal{M}$ that minimizes

$$L(\mathbf{D}, M) = L(M) + L(\mathbf{D}|M) \tag{50}$$

where $L(M)$ is the description length of the model M and $L(\mathbf{D}|M)$ is the length of the encoding of $\mathbf{D}$ by M, both measured in bits.
The motivation behind the application of the MDL principle is to find the discretization model that obtains the best compression of the data without losing multivariate interactions. Thus, in addition to the encoding lengths a penalty term for multivariate heterogeneity will be introduced.
In order to be able to apply the MDL principle to the given setting, it is necessary to determine the description length of an optimal encoding for the IPD algorithm. Since this requires finding an optimal compressor, which is not possible in most cases, Nguyen et al. [4] introduce a practically applicable MDL measure called the *practical score*. The practical score incorporates a penalty term for macro bins with heterogeneous distribution that makes use of the interaction distance defined in chapter 2.5. It factorizes per dimension and hence is particularly well suited for high-dimensional datasets. To find bounds for the minimum description lengths we will make use of Lemma 2.14. The practical score comprises three components: encoding the discretization, encoding the discretized data with an additional penalty term and encoding the errors.

Encoding the discretization consists of encoding the number of bins and the corresponding cutpoints for each dimension X_i. The number of bins are encoded based on the optimal MDL-encoding for integers proposed by Risannen [23]. For $k \in \mathbb{N}$ it is given by

$$L_{\mathbb{N}}(k) = \log^*(k) + \log(c_0) \tag{51}$$

where $\log^*(k) = \log(k) + \log(\log(k)) + ...$, only including the positive terms, and $c_0 = 2.8654$ is the normalizing constant. Given T_i micro bins, encoding k_i macro bins is equivalent to choosing one out of $\binom{T_i-1}{k_i-1}$ possibilities. Thus, we get

$$L(M) = L_{\mathbb{N}}(k_i) + \log\binom{T_i - 1}{k_i - 1} \tag{52}$$

Encoding the discretized data is done in two steps. At first the micro bin ids are encoded per macro bin. This involves

$$L_{bid}(M(X_i)) = \sum_{j=1}^{k_i} \left(L_{\mathbb{N}}(|B_i^j|) - \log\frac{|B_i^j|}{T_i} - |B_i^j|\log\frac{|B_i^j|}{T_i} \right) \tag{53}$$

Additionally, a penalty term $L_{mh}(M(\mathbf{D}))$ for multivariate heterogeneity is introduced. Simply encoding does not take interaction structures into account. The function of the penalty term is to account for micro cutpoints within a macro bin B_i^j where the interaction distance between two consecutive micro bins is large.

$$P(B_i^j) = \left\{ k \in \{1, ..., |B_i^j| - 1\} |\ ID(\mathbf{A} \setminus X_i | b_i^{j,k} \ ||\ \mathbf{A} \setminus X_i | b_i^{j,k+1}) \text{ is large} \right\} \tag{54}$$

The definition of "large" in $P(B_i^j)$ can be adjusted to retain the desired amount of detail during the discretization process. While Nguyen et al. [4] chose to set the threshold at a tertile of all ordered interaction distances between two consecutive bins, we chose to set it at the first percentile for the datasets presented in chapter 4.
To make sure we only penalize macro bins in which interactions are broken, the penalty term is formally given by

$$L_{mh}(M(X_i)) = \sum_{\substack{j=1 \\ |P(B_i^j)|>0}}^{k_i} L_{\mathbb{N}}(|P(B_i^j)|) + |P(B_i^j)|\log\big(|B_i^j| - 1\big) \tag{55}$$

Consequently, the term describing the entire encoding of the discretized data is

$$L(M(X_i)) = L_{mh}(M(X_i)) + L_{bid}(M(X_i)) \tag{56}$$

Encoding the errors is necessary since the MDL-principle requires lossless compression.

$$L_{error}(M(X_i)) = \sum_{j=1}^{k_i} |B_i^j| \log |B_i^j| \tag{57}$$

The above defined encoding lengths enable us to reconstruct the practical score for a dimension variable X_i and a model M.

Definition 3.2. For $X_i \in \mathbf{A}$ and $M \in \mathcal{M}$ the *practical score* is defined as

$$S_{pr}(X_i, M) = L(M) + L(M(X_i)) + L_{error}(M(X_i)) \tag{58}$$

By using dynamical programming, it is possible to find the overall optimal classifier by searching through all possible merges. However, Nguyen et al. [4] have shown that a greedy approach to this optimal implementation gives a good approximation. Hence, we restrict ourselves to using this realization for our purposes. The pseudocode for the *Greedy IPD algorithm* is given as Algorithm 3. Figure 3 visualizes a two-dimensional example.

Lemma 3.3. *The theoretical complexity of the IPD algorithm is* $\mathcal{O}(m^2 n^{1.5})$ *when choosing* $T_i = \sqrt{n}$ *i.e. using the k-proportional choice.*

Proof. The complexity of IPD per dimenson $X_i \in \mathbf{A}$ is made up by three parts.

1. Sorting the data costs $\mathcal{O}(n \log n)$
2. Computing the interaction distances costs $\mathcal{O}(mn^{1.5})$. The complexity per bin is $\mathcal{O}(\frac{mn^2}{T_i^2})$ and there are $T_i - 1 = n^{0.5} - 1$ bins.
3. Merging the micro bins costs $\mathcal{O}(T_i^2) = \mathcal{O}(n)$

Thus, the overall complexity is $\mathcal{O}(m^2 n^{1.5})$ □

Algorithm 3 Greedy IPD

Require: Database **D** consisting of n datapoints in m dimensions.

```
for each dimension X_i ∈ A do
    Discretize X_i into T_i micro bins
    Initialize the practical score S_pr by using the micro binned data
    while S_pr can be minimized do
        Find two consecutive macro bins B_i^j and B_i^l that minimize the practical score for
        the current iteration
        Merge B_i^j and B_i^l
        Determine the new practical score S_pr
    end while
end for
```

3.3 Correlation Based Hierarchical Clustering

In chapter 4 we notice disadvantages of the more complicated multivariate methods when applying them in practice. The runtime of IPD can become impractical and CPD can introduce artificial correlation when the number of principal components is small compared to the number of dimensions. The fundamental reason behind this behaviour is that the number of possible discretization models grows exponentially with the number of dimensions.

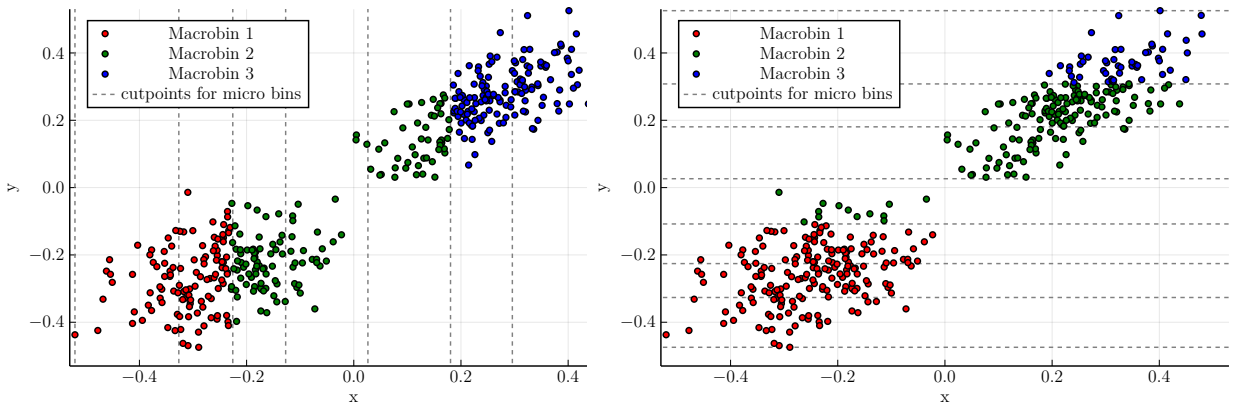


Figure 3: For this simple dataset we used k-means clustering to initialize 7 microbins and applied the IPD algorithm. The color coding indicates to which macro bin the datapoints belong.

To address these problems, we introduce correlation based hierarchical clustering and split the dataset into smaller clusters that will be discretized separately. This step is not essential but it can be used to compensate the speed of IPD and the correlation structure of CPD. Additionally, count matrices from large single-cell RNA sequencing datasets can be sparse, which can be problematic when dealing with distances due to numerical inaccuracies. Clustering reduces the dimension of the sets on which the discretization algorithms are applied and thus reduces numerical errors.

We aim to find a partition of the set of dimensions $\mathbf{A}$ that has high similarity within the clusters but low similarity between the clusters. For this purpose, we expand on the concept of clustering from chapter 3.1.3. However, when grouping dimensions instead of datapoints we perform *hierarchical clustering* to make use of the structural dependencies in $\mathbf{D}$. Compared to k-means, hierarchical clustering does not need any primary information. Initially each dimension represents one cluster. Repeatedly the two clusters with the lowest merging cost are identified and merged until the final number of partitions is achieved. As a measure for merging cost we use Ward's distance.

Definition 3.4. Given two clusters $\mathbf{A}$ and $\mathbf{B}$ the *Ward's distance* is defined as

$$\Lambda(\mathbf{A},\mathbf{B}) = \sum_{x_i \in \mathbf{A}\cup\mathbf{B}} dist(x_i - m_{\mathbf{A}\cup\mathbf{B}})^2 + \sum_{x_i \in \mathbf{A}} dist(x_i - m_{\mathbf{A}})^2 + \sum_{x_i \in \mathbf{B}} dist(x_i - m_{\mathbf{B}})^2 \quad (59)$$

where $m_{\mathbf{A}\cup\mathbf{B}}, m_{\mathbf{A}}, m_{\mathbf{B}}$ denote the cluster centers.

Again, it is possible to choose different distance metrics for this step. In order to cover monotonous relations, we use the following distance based on Spearman Rank Correlation

$$dist(\mathbf{x},\mathbf{y}) = \sqrt{\frac{1-\rho_{\mathbf{x},\mathbf{y}}}{2}} \quad (60)$$

The choice of the optimal number of clusters is non-trivial and cannot even be uniquely determined in some cases.

We restrict ourselves to selecting the number for which the partition minimizes the Total Correlation between the clusters. Using the explicit definition of the Kullback-Leibler divergence, the Total Correlation for clusters $\{X_{1,1}, ..., X_{1,n_1}\}, ..., \{X_{k,1}, ..., X_{k,n_k}\}$ can be simplified to

$$C_T(X_{1,1}, ..., X_{k,n_k}) = \Big[\sum_{i=1}^{k} H(X_{i,1}, ..., X_{i,n_i})\Big] - H(X_{1,1}, ..., X_{k,n_k}) \tag{61}$$

4 Benchmarking

In this chapter we benchmark the discretization methods from chapter 3 on single-cell RNA sequencing data. We compare Uniform Width, Bayesian Blocks, CPD and IPD regarding runtime, the number of non-empty bins as well as probabilistc interaction preserving capabilities. The algorithms have been implemented into an open-source Julia package called MultivariateDiscretization.jl [37] which expands on Discretizers.jl [36] that contains univariate schemes. All benchmarks were conducted on an Intel i7-8550U machine with 16GB RAM.

4.1 Runtime

We use in silico single-cell gene expression data from [24] to benchmark the runtimes. The samples were simulated based on 6 different gene regulatory networks using the BoolODE approach. For each setting we discretize sets consisting of 100, 200, 500, 1500 and 2000 cells for 8 and 16 genes.
As discussed in chapter 3, the complexity of IPD, CPD and Bayesian Blocks with regard to the number of cells is quadratic while uniform width binning displays linear dependence. The number of genes induces linear complexity on CPD, Bayesian Blocks and Uniform Width. Only IPD has polynomial complexity with respect to this parameter. Since CPD is faster than Bayesian Blocks on the tested datasets, we conclude that multivariate methods do not necessarily have to be slower than univariate schemes. In addition, we were able to optimize the speed of the CPD in comparison to the implementation in [4]. The runtime plots are displayed in figure 4.

4.2 Number of Bins

For the remaining benchmark analysis we use two experimental single-cell RNA sequencing datasets from gastrulating mouse embryos. Dataset 1 from Scialdone et al. [25] includes 1205 good quality cells that were sequenced through the Smart-seq2 protocol [26]. They represent a transcriptome-wide in vivo view of early mesoderm formation during mammalian gastrulation. We will restrict our analysis to 2085 *highly variable genes* (HVGs) which were identified in [25]. Genes are called highly variable if they contribute strongly to cell-to-cell variation within a homogeneous cell population. Details about the selection process can be found in [27].
The measurements from dataset 2 by Pijuan-Sala et al. [28] were performed using the 10x sequencing protocol. After selecting HVGs from sample 21, expression values from 831 genes and 4651 cells remain. Due to intrinsic features of the 10x protocol the count matrix of this dataset is sparse. Thus, clustering is a practically essential pre-processing step to avoid numerical errors. Referring to chapter 3.3 we use hierarchical clustering and minimize the Total Correlation between the clusters. We retain 20 clusters for dataset 1 and 10 clusters for dataset 2. Moreover, genes with less than 5 expression values are not discretized but regarded as already discrete. For the following results all schemes have been used in their optimal configuration described in chapter 3.

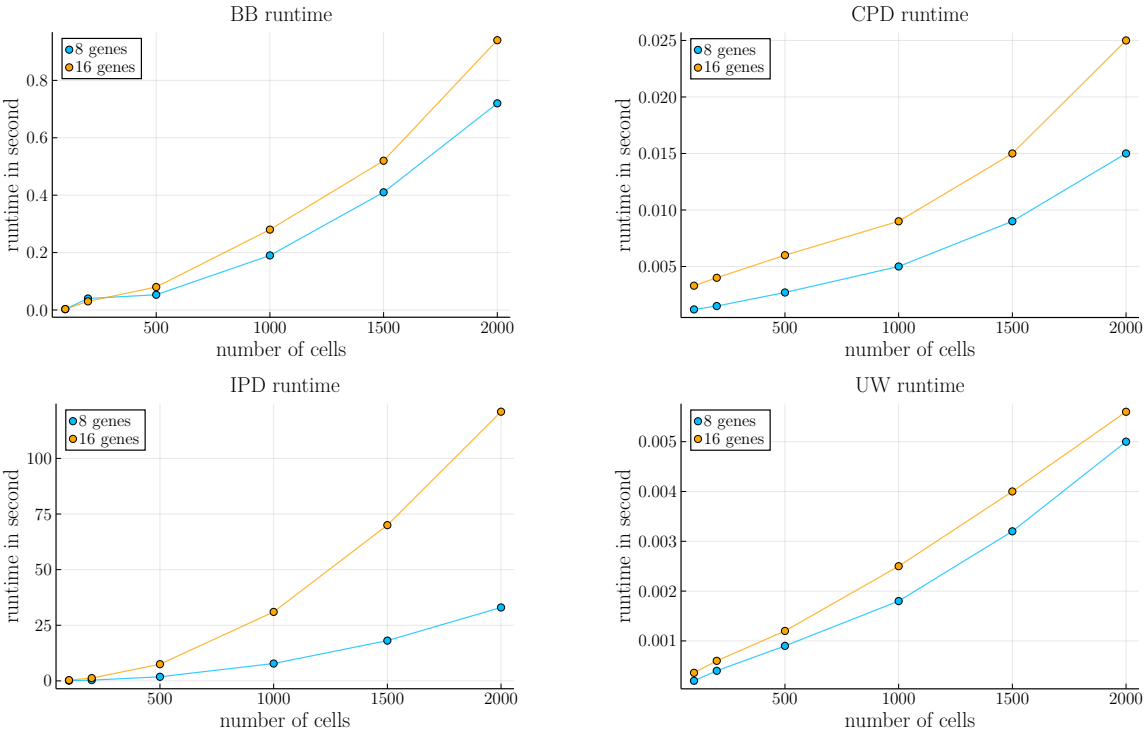


Figure 4: The mean runtimes of the four discretization schemes were benchmarked using synthetic data from [24] that was sampled on the basis of gene regulatory networks. This data is comparable to in vivo gene expression data yet the ground truth structure is known.

The different constructions of the discretization schemes result in significantly different distributions of the number of non-empty bins which are shown in figure 5 and supplementary figure 1. The number of micro bins for IPD is determined by the k-proportional choice. For CPD we apply the Rice rule since the uniform discretization is performed in a lower dimensional subspace where the k-proportional choice would underestimate an applicable number of bins. For reference Uniform Width discretization is performed with 10 intervals per gene.
The maximal number of non-empty bins for IPD is bounded by the number of micro bins. Since IPD is based on the minimum description length principle, it compresses the data the most, resulting in the smallest average number of bins for both datasets. The majority of genes from dataset 1 are partitioned into 7 intervals while the number of non-empty bins from dataset 2 is more evenly distributed. For CPD the maximum possible amount of non-empty bins is given by the product of the number of principle components and the number of bins per principle component. Hence, CPD outputs the largest average number of bins per gene. As the amount of unique expression values from certain genes is smaller than the maximum number of non-empty bins, some intervals remain empty. This is particularly common in the data from Pijuan-Sala et al. [28]. The number of bins that are selected by the Bayesian Blocks procedure is constantly higher than the amount of macro bins from IPD but undercuts 10 for some genes. The bin distribution for dataset 1 is more right-skewed with a higher average compared to dataset 2 while the maximum number of bins is larger for this data.

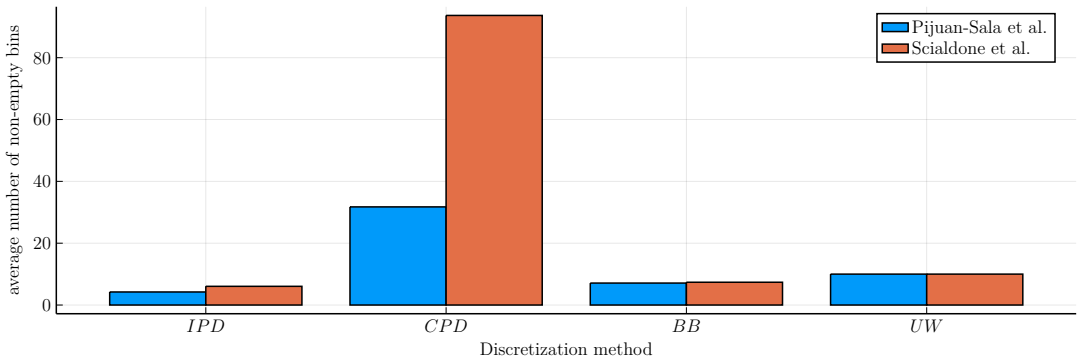


Figure 5: Average number of non-empty bins per gene for Interaction Preserving Discretization (IPD), Correlation Preserving Discretization (CPD), Bayesian Blocks (BB) and Uniform Width (UW)

4.3 Interaction Measures

As discussed in the introduction, the main property of the discretization methods we are interested in is the capability to preserve the interactions within the datasets. We investigate three of the measures for distribution similarity introduced in chapter 2, namely Pearson Correlation, Spearman Rank Correlation and the Cumulative Jensen Shannon Distance. In conjunction with the distribution of non-empty bins we can draw conclusions regarding the performance of the discretization schemes on datasets with different properties.

In figure 6 and supplementary figures 2 to 5 the distribution of the entries in the Pearson and the Spearman Rank Correlation matrices from the original and the discretized data are compared. Uniform Width Discretization preserves both correlations significantly less than the other schemes. Regarding dataset 1, IPD performs better compared to CPD and Bayesian Blocks on gene pairs with larger correlation. The opposite holds for dataset 2. In this case IPD does not preserve the Pearson Correlation structure to a sufficient extent. For both datasets CPD shows the best preservation capabilities for the Spearman Rank Correlation, being almost ideal for the second set.

Even though the Cumulative Jensen Shannon distance is a theoretically well-founded interaction measure, it can be hard to draw conclusions from its practical computation. It is necessary to normalize the CJS values for each dataset in order to be able to compare empirical and discretized data. The corresponding distributions are shown in figure 7.

We observe that regarding CJS all methods perform better on the second dataset. CPD shows the most consistent CJS preservation capabilities on both sets. Similar to the distribution of pairwise correlations between genes, Bayesian Blocks and IPD underestimate smaller and overestimate larger distances in the first dataset.

We conclude that IPD performs better on dataset 1 while CPD and Bayesian Blocks show more consistent benchmarks especially on dataset 2. An explanation for these observations could be that the data from Pijuan-Sala et al. [28] is less compressible due to intrinsic features of the measurement process. CPD and Bayesian Blocks feature a larger number of non-empty bins and thus retain more variation in the dataset. The smaller number of partitions proposed by IPD could potentially overcompress the data and therefore disturb the interaction structures.

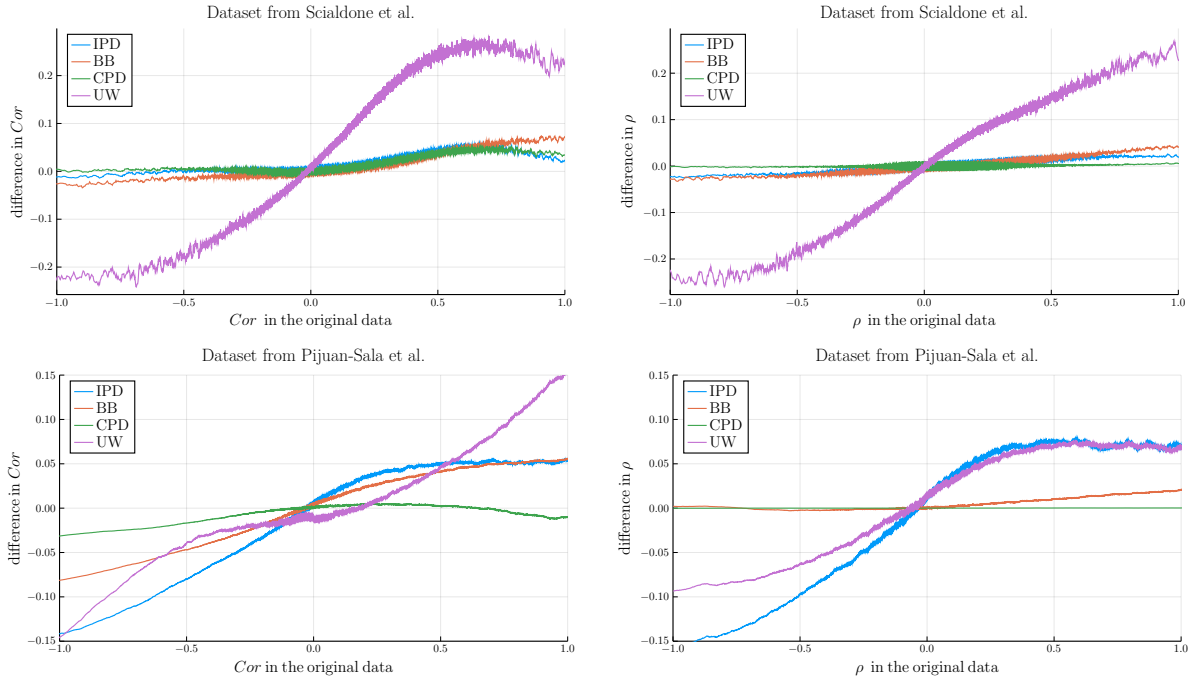


Figure 6: Difference in Pearson Correlation Cor and Spearman Rank Correlation ρ between the discretized data and the original dataset

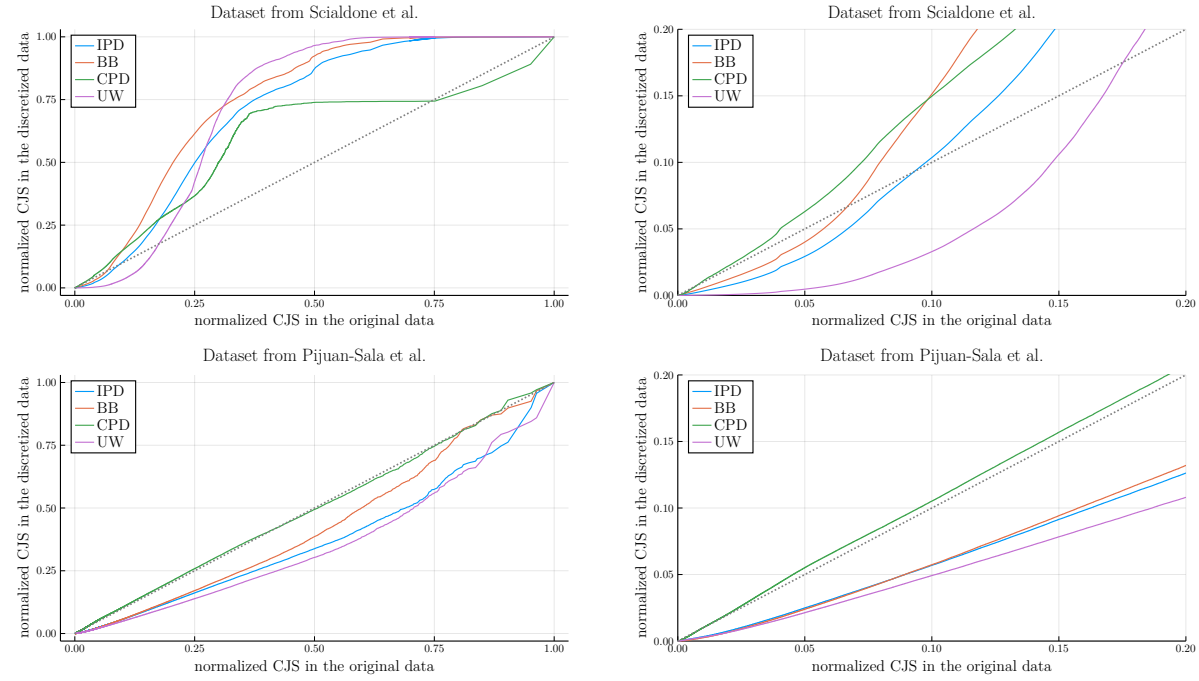


Figure 7: The Cumulative Jensen Shannon distance for each pair of genes in both discretized datasets [25] [28] can be directly calculated using the formula given in chapter 2.2. For the empirical measurements the cumulative distribution function was estimated by numeric integration.

5 Applications in Single-Cell RNA Sequencing Data Analysis

5.1 Inference of Gene Regulatory Networks

An emerging problem in systems biology is to identify the relationships between the various components of a natural system and infer how one component influences another. The goal of *gene regulatory networks* (GRN) is to structure this complex interplay of regulatory interactions between genes and small molecules that determine the expression level of a gene.
The introduction of single-cell RNA sequencing enables the observation of stochastic variations between individual cells in homogeneous cell populations. Theoretically this makes it possible to uncover the underlying gene regulatory network of a cellular process. Even though GRN inference has been an active area of research for over 20 years, it is still an open research question [29].

The *Benchmarking gene regulatory network inference from single-cell transcriptomic data* (BEELINE) [24] project by Pratapa et al. provides a pipeline to evaluate the performance of algorithms for gene regulatory network inference based on transcriptomic data. One of the inference schemes that shows promising results is the *PIDC algorithm* introduced by Chan et al. [7] which requires discretized single-cell data. The standard implementation of PIDC only considers Bayesian Blocks and Uniform Width discretization. Our goal is to discretize the gene expression data with the multivariate methods from chapter 3 and investigate if it is possible to improve the inference process. In addition to the measures for prediction accuracy proposed by Pratapa et al. we analyse graph theoretic properties of the resulting networks as well.

In practice gene regulatory networks are inferred based on measurements of gene co-expression. The proposed measure of co-expression for the PIDC scheme expands on the concept of PIDC introduced in chapter 2.3.
In the following the set of dimensions $\mathbf{A}$ describes the set of genes and each datapoint corresponds to a cell.

Definition 5.1. The *proportional unique contribution* (PUC) between $X, Y \in \mathbf{A}$ is

$$u_{X,Y} = \sum_{\mathbf{A}\setminus\{X,Y\}} \frac{U_Z(X;Y) + U_Z(Y;X)}{I(X;Y)}$$

Referring to the distribution of the PUC, the PIDC algorithm assigns weights to each possible combination of two genes. The pairs with a score that is higher than a pre-specified threshold are connected by edges. The pseudocode is given as Algorithm 4.
Estimating the distribution of the PUC scores in step 3 can be executed by multiple approaches. We assume a Gaussian distribution and determine the parameters by maximum likelihood estimation.

Algorithm 4 PIDC Inference Algorithm

Require: Set of discrete genes $\mathbf{A}$, threshold t

1: Calculate $U_Z(X;Y)$ and $U_Z(Y;X)$ for every possible triplet $\{X,Y,Z\} \in \mathbf{A}$
2: Determine $u_{X,Y}$ for every pair $\{X,Y\} \in \mathbf{A}$
3: For each $X \in \mathbf{A}$ estimate the cumulative distribution function $F_X(\cdot)$ of all PUC scores involving X
4: The confidence of an edge between two genes $X, Y \in \mathbf{A}$ is given by $c_{X,Y} = F_Y(u_{X,Y}) + F_Y(u_{X,Y})$
5: Generate the network from all edges with $c_{X,Y} \geq t$ for $X, Y \in \mathbf{A}$
6: **return** Gene regulatory network $G = (V, E)$

The first measure of prediction accuracy that we examine is the *area under the precision recall curve* (AUPRC). The adjacency matrix of the underlying ground truth matrix is regarded as binary in order to compute the precision and recall values for each discretization scheme.
Since the precision recall curve does not take any structural properties of networks into account, we additionally consider the following graph specific metrics.

Definition 5.2. Let $G = (V, E)$ be an undirected graph consisting of m vertices and n edges. The *eccentricity* $\epsilon(v)$ of a vertex $v \in V$ is the greatest pathlength between v and any other vertex in V.
The *diameter* of G is

$$d(G) = \max_{v \in V} \epsilon(v) \tag{62}$$

Apart from these distance properties of a graph G we investigate the clustering structure that can be important to identify groupings of genes.

Definition 5.3. Given a graph $G = (V, E)$ the *local clustering coefficient* of node $v \in V$ with degree k_v is defined as

$$C_v = \frac{2L_v}{k_v(k_v - 1)} \tag{63}$$

where L_v represents the number of links between the k_v neighbors of vertex v. The *average clustering coefficient* of a graph G is

$$\langle C \rangle = \frac{1}{m} \sum_{i=1}^{m} C_i \tag{64}$$

The average clustering coefficient is the probability that two neighbours of a randomly selected node are linked to each other and thereby captures the degree of clustering of an entire network.

BEELINE provides sample data for the purpose of benchmarking GRN inference. We selected synthetic data from *trifurcating* (TF) and *hematopoetic stem cell differentiation* (HSC) networks. TF networks are purely artificial while HSC networks describe a simple model for myeloid differentiation by Krumsiek et al. [30]. All sample networks which correspond to one type are isomorphic. Example representations are given in supplementary

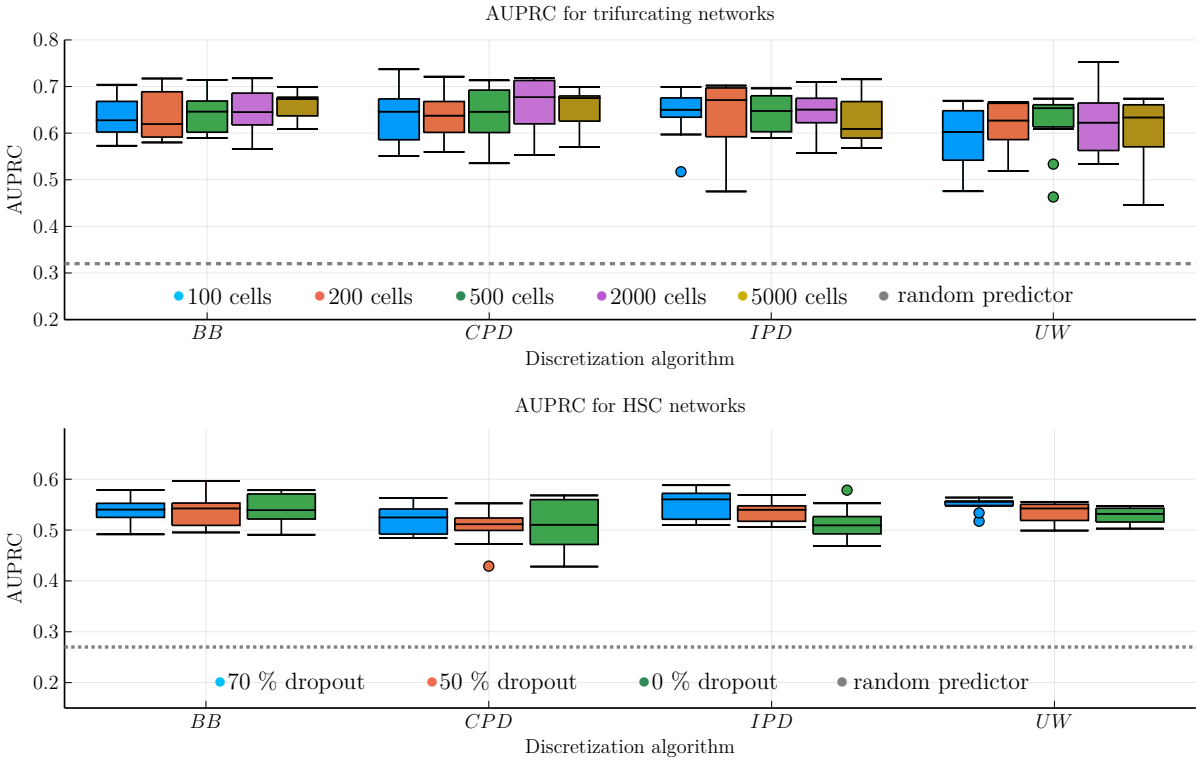


Figure 8: The AUPRC values were calculated by varying the threshold parameter t in Algorithm 4. The distributions are grouped by the discretization algorithms from chapter 3. The dashed line indicates the performance of a random binary classifier.

figure 6. Samples from TF networks consist of 8 genes and 100, 200, 500 and 2000 cells. Synthetic data from HSC networks was created by a boolean model that involves 11 genes or 2000 cells with either 70%, 50% or no dropout. A dropout rate of q indicates that if a cell c belongs to the lowest qth percentile of cells ordered by increasing expression values for a gene g, the expression of g in c is 0 with a q% chance.
For the following results the discretization methods were applied with the parameter settings explained in chapter 3.

We note that the AUPRC was only improved slightly by any of the discretization schemes. On average all four discretization algorithms perform similarly compared to a random predictor which is shown in figure 8. The network diameter is mostly estimated correctly independent of the underlying schemes. This can be seen in figure 9.

However, when considering the average clustering coefficient there are more significant differences. $\langle C \rangle$ was estimated noticeably less accurately by Bayesian Blocks and Uniform Width. Additionally to figure 10 we performed Wilcoxon signed-rank tests [31] to confirm this hypothesis. The corresponding p-values are shown in supplementary table 1. Since $\langle C \rangle$ captures structural properties of the clustering structure of a network, this observation is important in the context of gene regulatory networks.

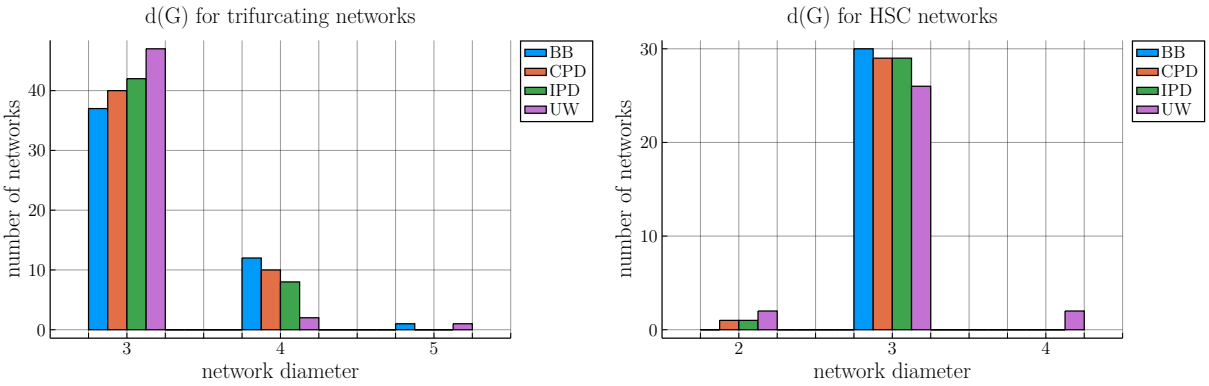


Figure 9: The histograms show the distribution of the network diameter $d(G)$ corresponding to the graphs that were inferred using the PIDC algorithms based on the discretization algorithms from chapter 3. The true network diameter is 3 in both cases.

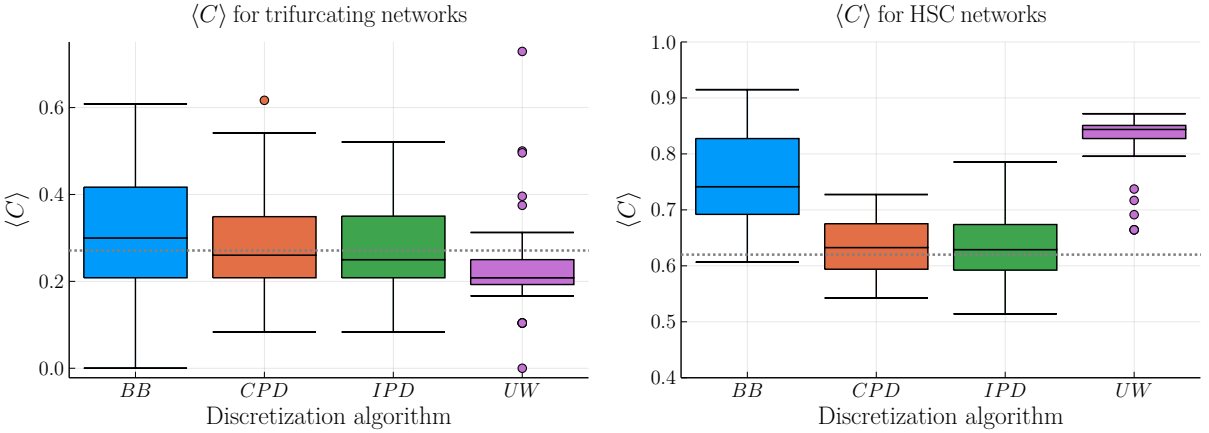


Figure 10: The boxplots display the distribution of the average clustering coefficient $\langle C \rangle$ for trifurcating and HSC networks inferred by the PIDC algorithm based on the discretization schemes from chapter 3. Since the ground truth networks are isomorphic for each type, the true value of $\langle C \rangle$ is constant which is indicated by the dashed line.

To consolidate the observations from synthetic data where the exact ground truth networks are known, we analyse the performance of PIDC with different discretization methods on experimental single-cell RNA sequencing data from Nestorowa et al. [32]. The set consists of 1072 good quality mouse hematopoetic stemcells (mHSC). This data is intersected with the STRING database [33] by Szklarczyk et al. that contains protein-protein associations and serves as reference network. The *early precision ratios* (EPR) are displayed in table 1. Similar to the synthetic case there are no significant improvements in binary classification. Since the EPR only considers the top k edges, where k is the number of activating genes, this could indicate that the differences between the networks resulting from different discretization schemes lie mainly among the lower ranked edges. It will be possible to further investigate this observation, as soon as more precise reference networks related to experimental datasets are available.

Discretization algorithm	Bayesian Blocks	Uniform Width	CPD	IPD
Early precision ratio	7.3	7.4	7.4	6.9

Table 1: We selected 500 HVGs and all 204 transcription factors from the dataset by Nestorowa et al. [32]. Subsequently, this data was intersected with the gene interactions that are collected in the STRING database [33]. The *early precision* (EP) is defined as the precision among the top k edges in the ground truth network, where k is the number of activating genes. The *early precision ratio* is the ratio of the EP for the model and the EP for a random binary predictor.

5.2 Critical Variable Selection

Expanding on the concepts introduced in chapter 2.7, we apply *Critical Variable Selection* on single-cell RNA sequencing data. CVS was conceptualised by Grigolon et al. [6] to identify relevant positions in proteins. In our setting we consider discrete data from L genes and M cells. For single-cell RNA sequencing data it is a priori not clear how to practically apply CVS and how to interpret the results.

For cells numerated by $\alpha \in \{1, ..., M\}$ the gene expression is denoted by $\vec{s}^{\,\alpha} = (a_1^\alpha, ..., a_L^\alpha)$ where a_i^α is the discrete value for cell i and gene α. Each sequence $\vec{s}^{\,\alpha}$ is regarded as the solution of an unknown biological optimization problem. Subsequences in the gene expression that occur very often are assumed to be optimal under broader conditions compared to subpatterns that occur rarely. Thus, the frequency with which a specific subsequence occurs provides an estimate of the function being optimised.
Referring to chapter 2.7 the goal of CVS is to find the subset $I \subseteq \{1, ..., L\}$ of length n that maximizes $H(K_I)$.

$$I_n^\star = \underset{I:|I|=n}{\operatorname{argmax}}\, H(K_I) \tag{65}$$

Algorithm 5 Critical Variable Selection

Require: Discrete expression values for m genes and n cells, number of critical sites n
1: Determine an initial subset $I \subseteq \{1, ..., L\}$ with $|I| = n$
2: Calculate the initial entropy $H(K_I)$
3: **repeat**
4: Construct I' by changing a random position $i \in I$ to a randomly chosen position $i' \notin I$
5: Calculate $H(K_{I'})$
6: **if** $H(K_{I'}) \geq H(K_I)$ **then**
7: $I \leftarrow I'$
8: **end if**
9: **until** $H(K_I)$ does not change for a sufficiently large number of steps m
10: **return** critical sites I

In practice this maximization task is approximated by a greedy gradient ascent algorithm. Since the solutions converge to local maxima, we run the method R times to receive a distribution of results. The number of times that position i is selected when evaluating subsequences of length n is denoted as $c_i(n)$. The procedure for a single iteration of this implementation of CVS is displayed as Algorithm 5.

We analyse the performance of CVS on the experimental dataset from Scialdone et al. [25] which was discretized in chapter 4. We increase the number of CVS runs in steps of 5 until for $R = 50$ and $m = 500$ the distribution of $c_i(n)$ stabilises. The resulting entropies corresponding to $n \in \{5, 10, 20\}$ are displayed in figure 11. Figure 12 shows the distribution of $c_i(n)$.

Using the marker genes available from [25] we perform Fisher's exact tests [34] to examine whether the selected critical genes are significantly more likely to be marker genes. Genes with $c_i(n) > 1$ are defined as critical. The resulting p-values are shown in table 2. We observe that Bayesian Blocks, CPD and IPD, which show the best interaction preserving capabilities in chapter 4, result in more significant enrichment. Figure 13 indicates that the marker genes among the critical genes are not distributed differently than the marker

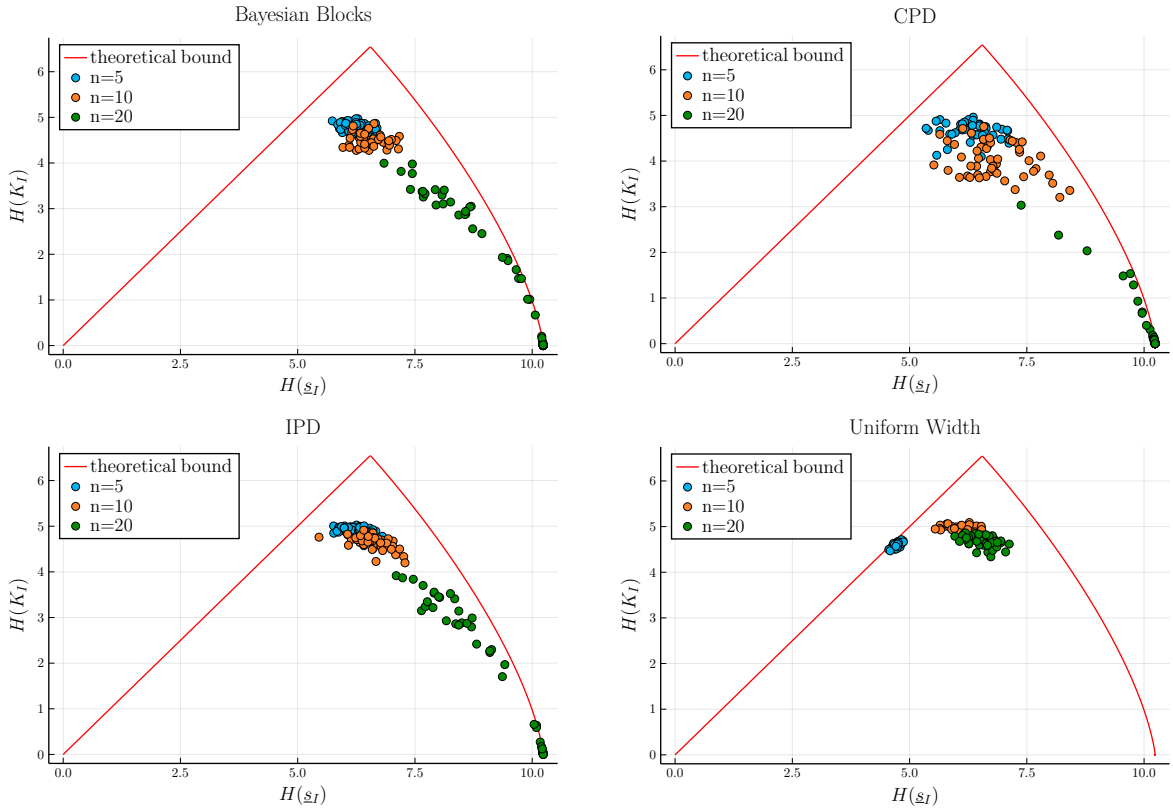


Figure 11: The theoretical bound was calculated using the approach explained in chapter 2.7. The datapoints represent the entropies $H(\underline{s}_I)$ and $H(K_I)$ of the resulting sequences after $R = 50$ iterations of CVS with $m = 500$ on the discretized dataset by Scialdone et al. [25].

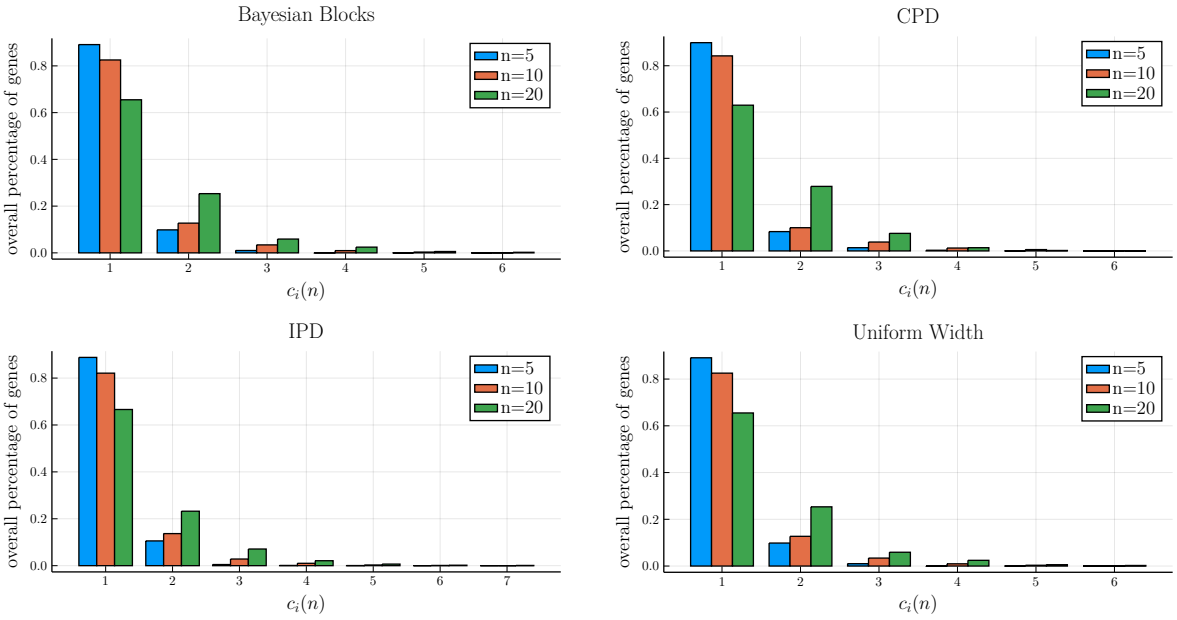


Figure 12: The histograms show the normalized distribution of $c_i(n)$ resulting from $R = 50$ iterations of CVS on the discretized dataset by Scialdone et al. [25].

p-values	Bayesian Blocks	Uniform Width	CPD	IPD
n=5	0.05	0.45	0.01	0.83
n=10	<0.001	0.46	0.07	0.04
n=20	0.01	0.29	0.12	0.02

Table 2: The p-values are calculated using the Fisher's exact test [34]. The success probability is given by the ratio of the number of marker genes and the overall amount of genes. The null hypothesis is that success probability among the critical genes is equal to the success probability in the data.

genes in the entire dataset. We confirm this hypothesis by Anderson-Darling tests [35]. Supplementary table 2 shows the corresponding p-values.

To analyse the similarity between the subsets of critical genes, we calculate the Jaccard Similarity Coefficient between the L-dimensional vectors containing the $c_i(n)$ values per gene.

Definition 5.4. For two vectors $\mathbf{x} = (x_1, ..., x_M)$ and $\mathbf{y} = (y_1, ..., y_M)$ with $x_i, y_i \geq 0$ the *Jaccard Similarity Coefficient* is defined as

$$J(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{M} \max(x_i, y_i)}{\sum_{i=1}^{M} \min(x_i, y_i)} \tag{66}$$

From figure 14 we conclude that the subsets of critical genes do not show larger set similarities.

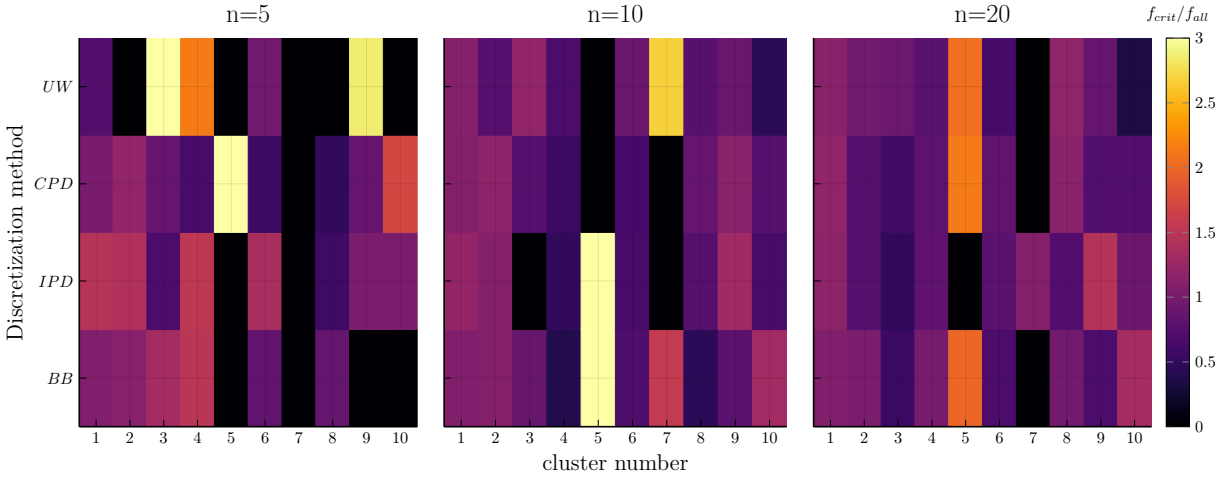


Figure 13: We calculated the distribution f_{crit} of marker genes between the different clusters among the critical genes and compared it to the distribution f_{all} of marker genes between the different clusters in the original data. The deviations in clusters 5 and 7 can be explained by the small number of corresponding marker genes in the dataset.

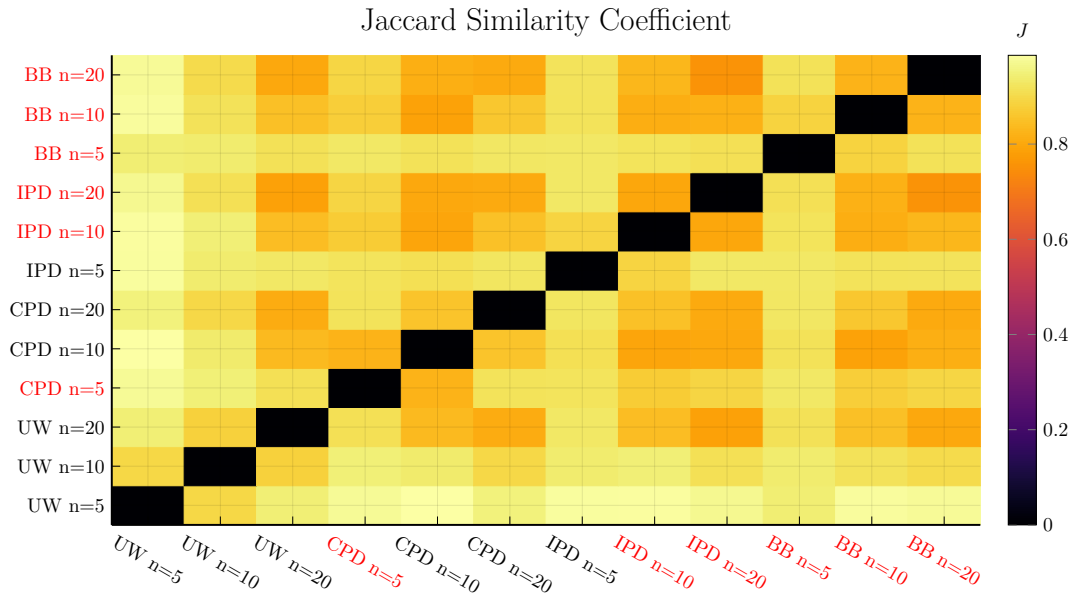


Figure 14: The entries in the heatmap correspond to the Jaccard Similarity Coefficients between the L-dimensional vectors containing the $c_i(n)$ values per gene. Red labels indicate significance at a level of $\alpha = 5\%$ corresponding the Fisher tests shown in table 2.

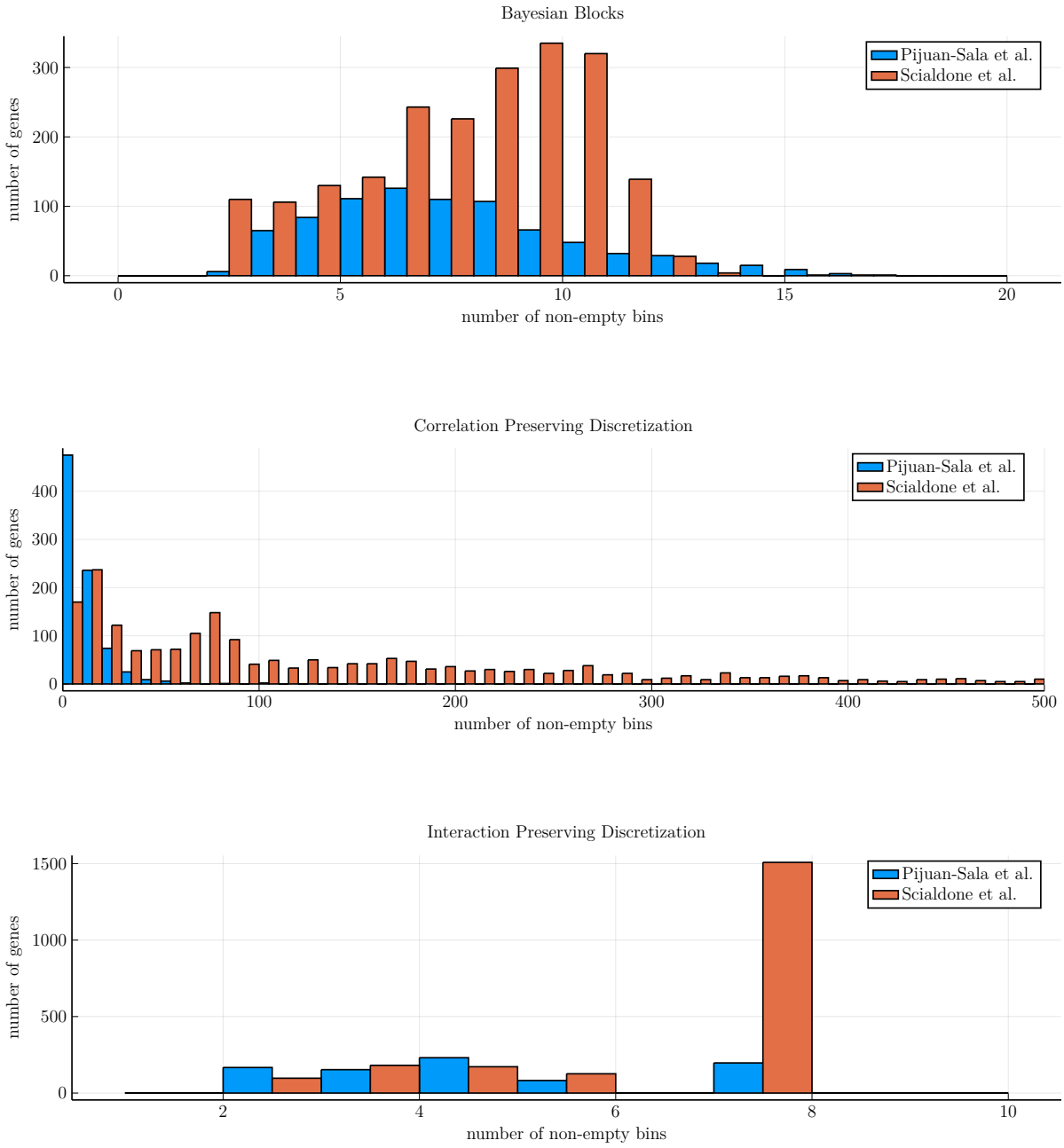
6 Conclusion

Based on chapters 2 to 4 of this thesis we were able to quantitatively benchmark Uniform Width Discretization, Bayesian Blocks, Correlation Preserving Discretization and Interaction Preserving Discretization on experimental single-cell RNA sequencing data with respect to their interaction preserving capabilities. We conclude that in practice the optimal choice of discretization scheme is dependent on the specific properties of a dataset and the requirements of the user.
For larger datasets with strong interaction structures we recommend the usage of multivariate algorithms. If fast runtime on high-dimensional data is required, we advise against IPD due to its computational complexity. Nevertheless, IPD in its optimal configuration accurately preserves interactions when a dataset is compressible in an information theoretic sense. However, this can be difficult to measure a priori. If the discretization process is targeted towards the preservation of linear or monotonic correlations, we found that CPD produces the best output. CPD also was the fastest algorithm we tested in Julia. In addition, we introduced a novel method based on hierarchical clustering to reduce the runtime of computationally expensive, multivariate schemes on high-dimensional datasets. In case the data mainly contains weaker interaction structures, univariate algorithms can be applicable as well. Bayesian Blocks discretization can be useful in a setting where very little knowledge about the dataset is available since it is entirely unsupervised [2] [7].
Our implementations of the discretization methods that were analysed in this thesis are available in an open-source Julia package [37] which can easily be extended by additional methods in the future.

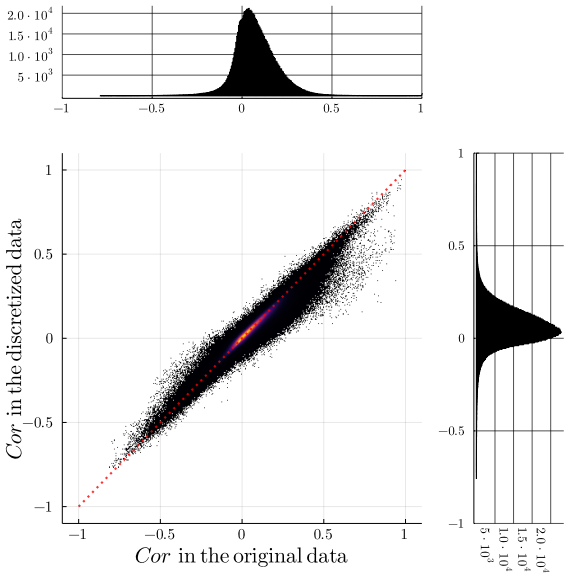
In chapter 5 we could observe benefits of interaction preserving discretization on two information theory based algorithms for single-cell RNA sequencing data analysis.
Even though it was not possible to improve the binary classification accuracy of the PIDC algorithm, on synthetic data we found that structural properties of the underlying ground truth networks such as the average clustering coefficient can be estimated significantly better when using multivariate discretization schemes. As soon as more precisely annotated gene regulatory networks related to experimental datasets are available for reference, it will be possible to further investigate this observation.
Using different discretization methods together with CVS on single-cell RNA sequencing data, we could show that interaction preserving discretization does significantly increase the enrichment of marker genes among the selected critical genes.

Since our results in chapter 5 are promising and the field of single-cell RNA sequencing data analysis is fast-growing, it is likely that in the future more applications can benefit from the work laid out in this thesis.

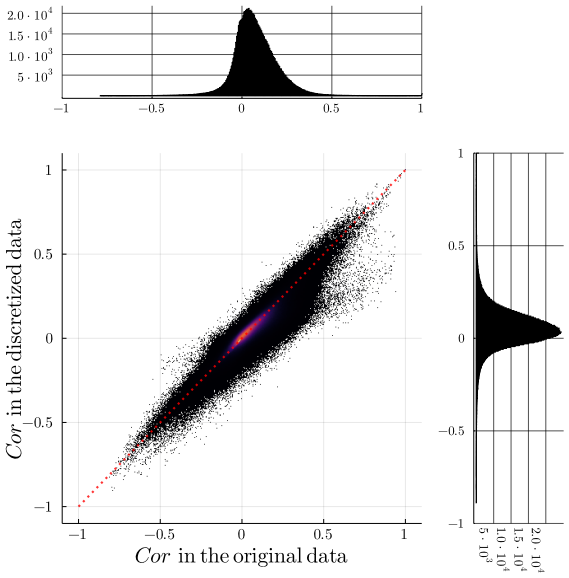
7 Supplementary Figures and Tables



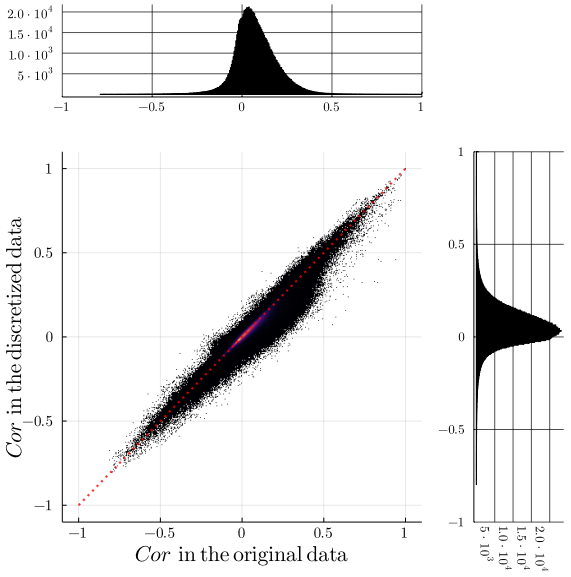
Supplementary figure 1: The histograms display the distribution of non-empty bins for the three discretization methods with a non-constant number of bins on both datasets [25] [28]. The overall number of bins for the second set is lower since it consists of fewer highly variable genes.



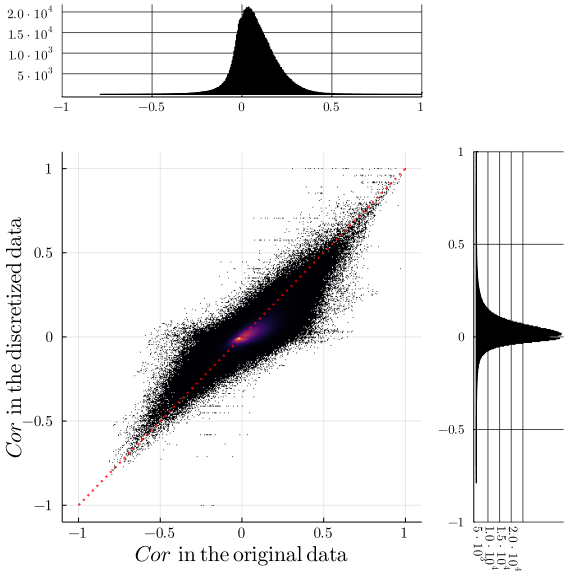
(a) Bayesian Blocks



(b) Correlation Preserving Discretization

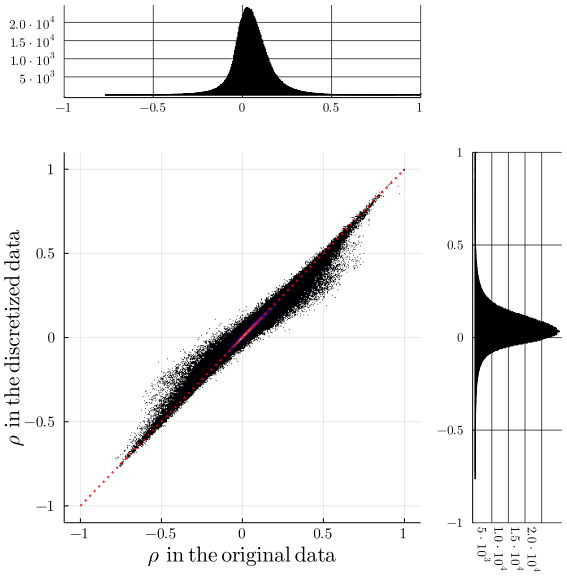


(c) Interaction Preserving Discretization

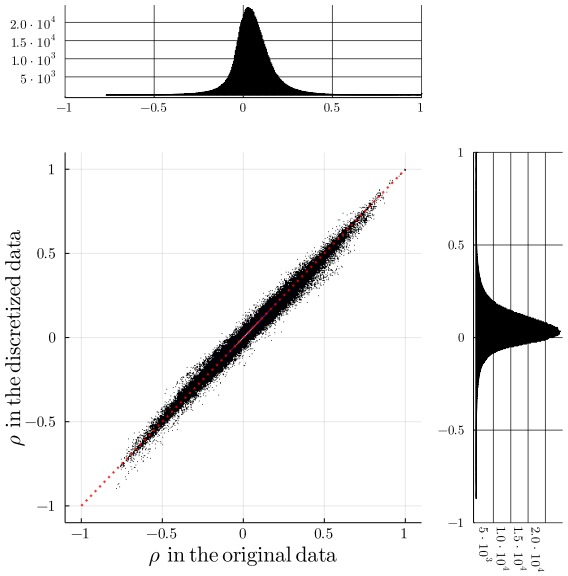


(d) Uniform Width

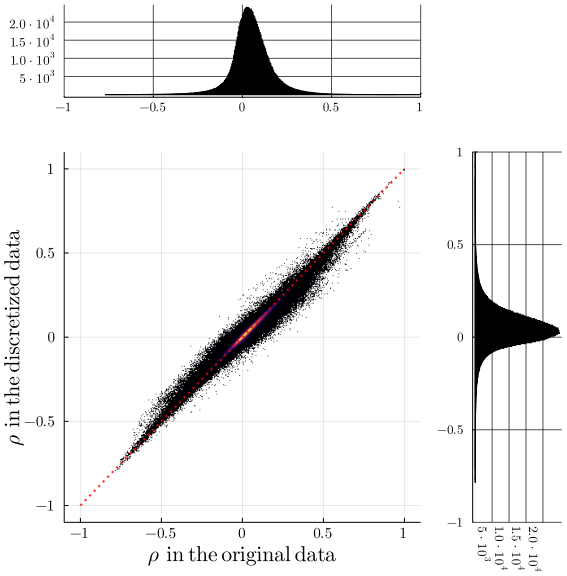
Supplementary figure 2: The histograms show the dependency of the Pearson Correlation *Cor* between genes in the original and the discretized data corresponding to the set by Scialdone et al. [25]. In case of an optimal discretization algorithm the points would be distributed along the bisecting line displayed in red.



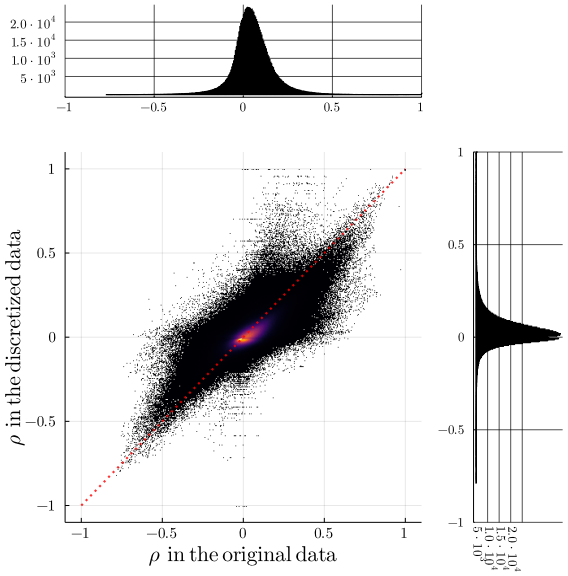
(a) Bayesian Blocks



(b) Correlation Preserving Discretization

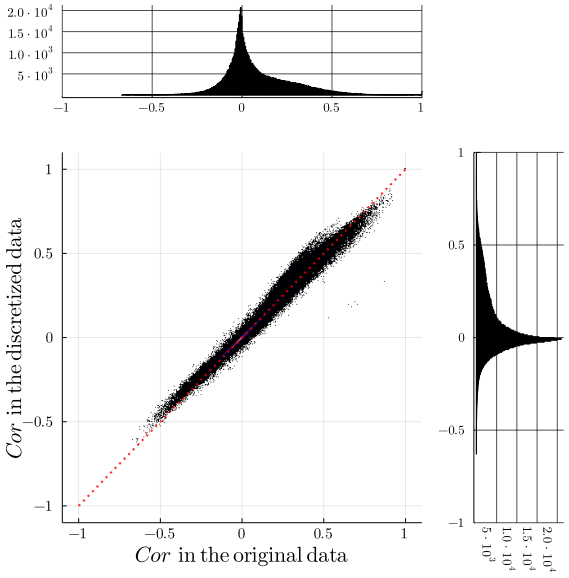


(c) Interaction Preserving Discretization

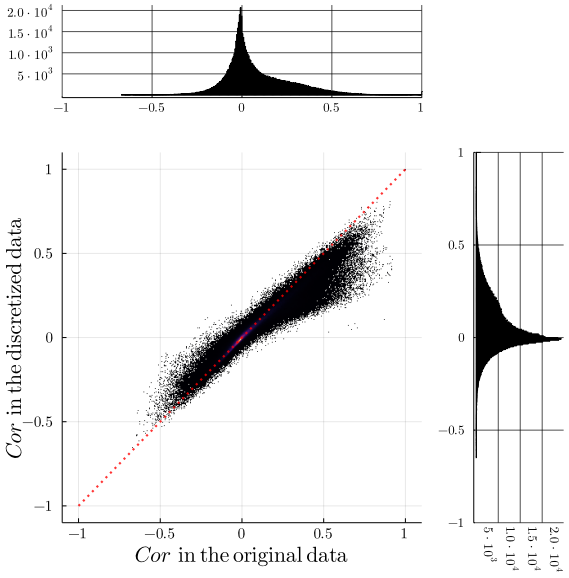


(d) Uniform Width

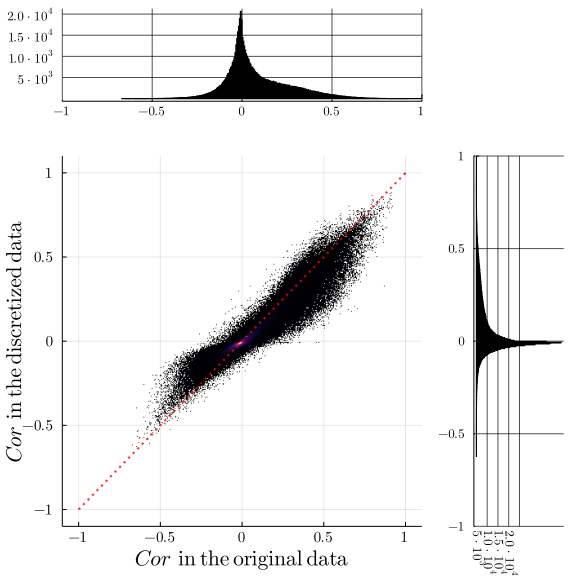
Supplementary figure 3: Similar to supplementary figure 2 the histograms show the dependency of the Spearman Rank Corrleation ρ between genes in the original and the discretized data corresponding to the set by Scialdone et al. [25].



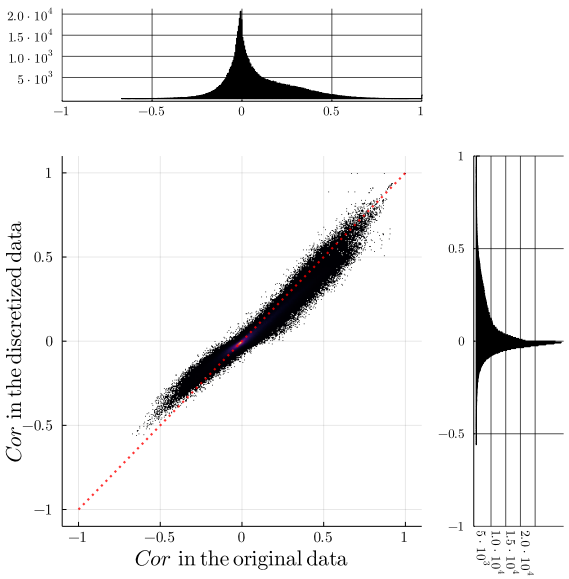
(a) Bayesian Blocks



(b) Correlation Preserving Discretization

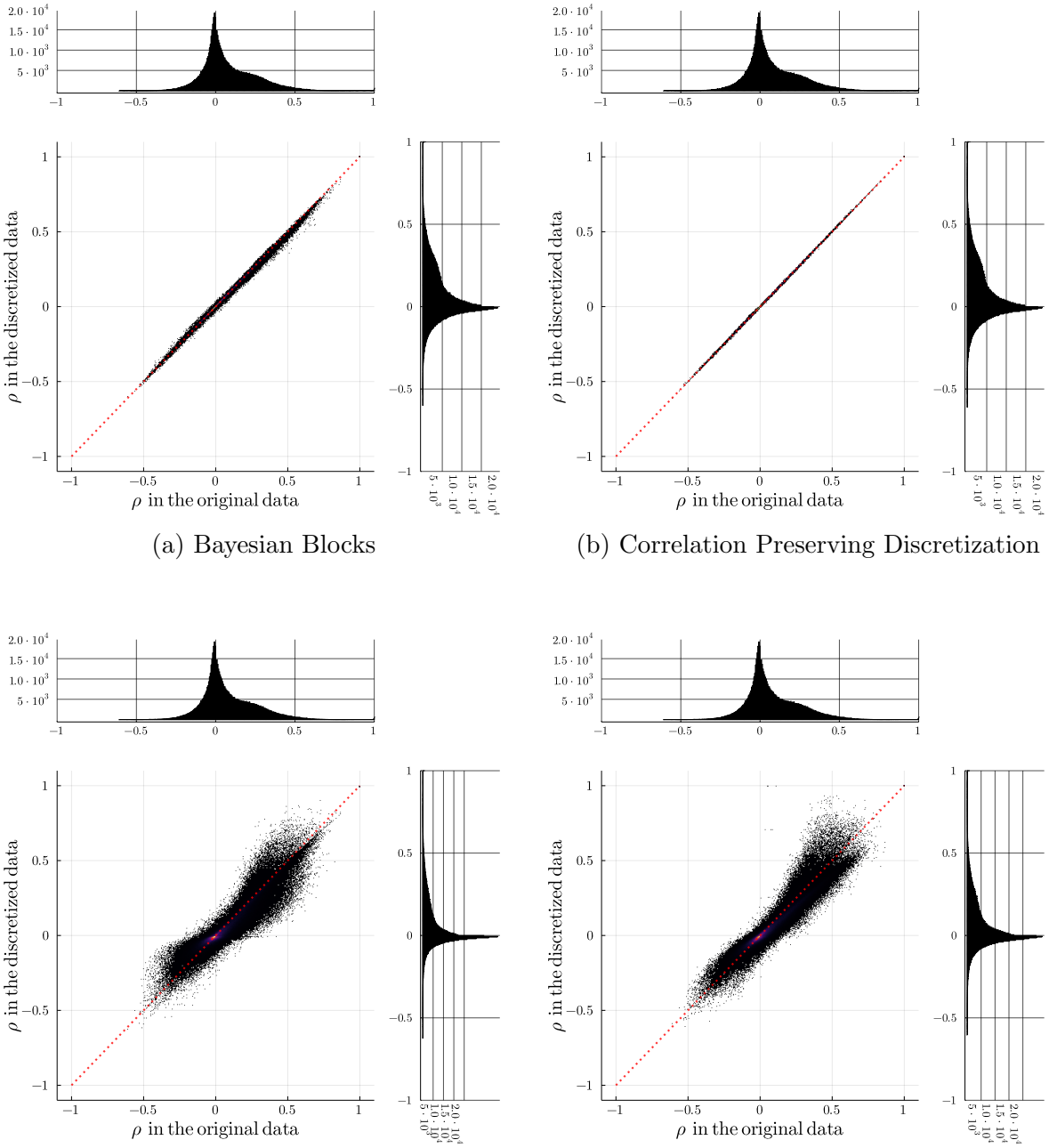


(c) Interaction Preserving Discretization



(d) Uniform Width

Supplementary figure 4: Similar to supplementary figure 2 the histograms show the dependency of the Pearson Correlation *Cor* between genes in the original and the discretized data corresponding to the set by Pijuan-Sala et al. [28].



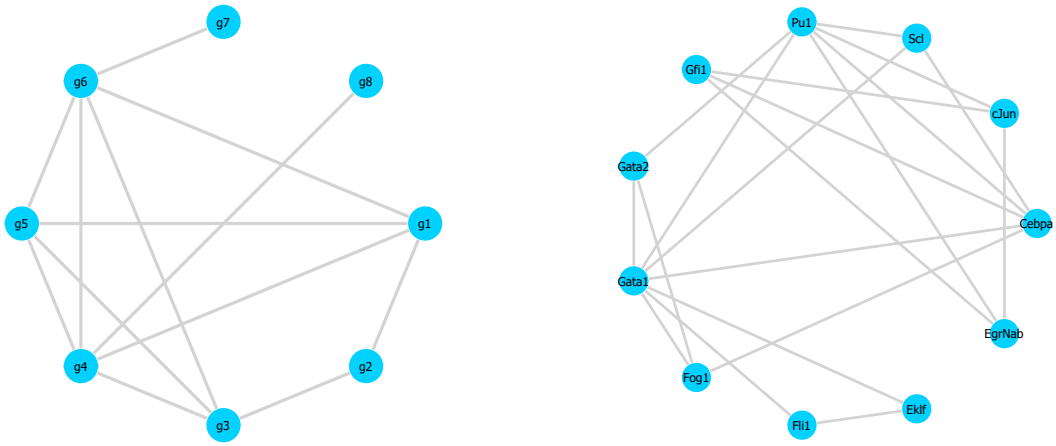
(a) Bayesian Blocks

(b) Correlation Preserving Discretization

(c) Interaction Preserving Discretization

(d) Uniform Width

Supplementary figure 5: Similar to supplementary figure 2 the histograms show the dependency of the Spearman Rank Corrleation ρ between genes in the original and the discretized data corresponding to the set by Pijuan-Sala et al. [28].



(a) TF network (b) HSC network

Supplementary figure 6: Synthetic data which was used in chapters 4 and 5 was sampled from trifurcating and hematopoetic stem cell differentiation networks [24]. All networks that belong to one type are isomorphic to the given representations.

p-values	Bayesian Blocks	Uniform Width	CPD	IPD
TF	0.15	0.06	0.91	0.66
HSC	<0.001	<0.001	0.51	0.61

Supplementary table 1: The p-values correspond to Wilcoxon signed-rank tests [31] of the null hypothesis that the median of the distribution given by the inferred average clustering coefficients is equal to the true value of $\langle C \rangle$.

p-values	Bayesian Blocks	Uniform Width	CPD	IPD
n=5	0.18	0.08	0.56	0.75
n=10	0.44	0.11	0.89	0.48
n=20	0.41	0.13	0.43	0.59

Supplementary table 2: In order to examine whether the distribution of the selected marker genes between the different clusters differs from the distribution in the original data, we perform Anderson–Darling tests [35]. The null hypothesis is that the critical marker genes originate from the same distribution as the marker genes in the dataset.

References

[1] LUSTGARTEN JONATHAN, GOPALAKRISHNAN VANATHA, GROVER HIMANSHU, VISWESWARN SHIYAM *Improving classification performance with discretization on biomedical datasets*, Proceedings of the Fall Symposium of the American Medical Informatics Association 2008, pages 445–449, 2008

[2] JEFFREY D. SCARGLE, JAY P. NORRIS, BRAD JACKSON, JAMES CHIANG *Studies in Astronomical Time Series Analysis VI. Bayesian Block Representations*, The Astrophysical Journal, 764(2):167, 2013

[3] SAMEEP MEHTA, SRINIVASAN PARTHASARATHY, HUI YANG *Toward Unsupervised Correlation Preserving Discretization*, IEEE transactions on knowledge and data engineering, 17(9):1174-1185, 2005.

[4] HOANG-VU NGUYEN, JILLES VREEKEN, EMMANUEL MÜLLER, KLEMENS BÖHM *Unsupervised interaction-preserving discretization of multivariate data*, Data Mining and Knowledge Discovery, 28(5-6):1366-1397, 2014

[5] FUCHOU TANG, CATALIN BARBACIORU, YANGZHOU WANG, ELLEN NORDMAN, CLARENCE LEE, NANLAN XU, XIAOHUI WANG, JOHN BODEAU, BRIAN B TUCH, ASIM SIDDIQI, KAIQIN LAO, AZIM SURANI *mRNA-Seq whole-transcriptome analysis of a single cell*, Nature methods, 6(5):377-382, 2009

[6] SILVIA GRIGOLON, SILVIO FRANZ, MATTEO MARSILI *Identifying relevant positions in proteins by Critical Variable Selection*, Molecular BioSystems, 12(7):2147-2158, 2016.

[7] THALIA E. CHAN, MICHALE P.H. STUMPF, ANN C. BABTIE *Gene regulatory network inference from single-cell data using multivariate information measures*, Cell Systems, 5(3):251-267, 2017

[8] JIANHUA LIN *Divergence Measures Based on the Shannon Entropy*, IEEE Transactions on Information Theory, 31(1):145-151, 1991

[9] HOANG-VU NGUYEN, JILLES VREEKEN *Non-parametric Jensen-Shannon Divergence*, Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 9285:173-189, 2015

[10] USAMA FAYYAD, KEKI IRANI *Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning*, Lecture Notes in Computer Science, 8140:155-169, 1993

[11] WILLIAM J. MCGILL *Multivariate Information Transmission*, Psychometrika, 19(2):97–116, 1954

[12] NICHOLAS TIMME, WESLEY ALFORD, BENJAMIN FLECKER, JOHN M. BEGGS *Synergy, redundancy, and multivariate information measures: an experimentalist's perspective*, Journal of Computational Neuroscience, 36(2):119-40, 2013

[13] PAUL L. WILLIAMS, RANDALL D. BEER *Nonnegative Decomposition of Multivariate Information*, Computing Research Repository, 1004(2515):1, 2010

[14] THOMAS M. COVER, JOY A. THOMAS *Elements of Information Theory*, John Wiley and Sons Publications, Second Edition, 2006

[15] IAN T. JOLLIFFE *Principal Component analysis*, Springer, Second edition, 2002

[16] MATTEO MARSILI, IACOPO MASTROMATTEO, YASSER ROUDI *On sampling and modeling complex systems*, Journal of Statistical Mechanics: Theory and Experiment, 2013(09):P09003, 2013

[17] DAVID N. RESHEF, YAKIR A. RESHEF, HILLARY K. FINUCANE, SHARON R. GROSSMAN, GILEAN MCVEAN, PETER J. TURNBAUGH, ERIC. S. LANDER, MICHAEL MITZENMACHER, PARDIS C. SABETI *Detecting Novel Associations in Large Data Sets*, Science, 334(6062):1518-1524, 2011

[18] HERBERT A. STURGES *The Choice of a Class Interval*, Journal of the American Statistical Association, 21(153):65-66, 2012

[19] DAVID P. DOANE *Aesthetic Frequency Classifications*, The American Statistician, 30(4):181-183, 1976

[20] MIGUEL A. ALVAREZ-CARMONA, JESUS A. CARRASCO-OCHOA *Combining Techniques to Find the Number of Bins for Discretization*, International Conference of the Chilean Computer Science Society, pages 54-57, 2013

[21] WEBSTER CASH *Parameter estimation in astronomy through application of the likelihood ratio*, Astrophysical Journal, 28(1):939-947, 1987

[22] CHRISTOPHER M. BISHOP *Pattern Recognition and Machine Learning*, Second edition, Springer, 2006

[23] JORMA J. RISSANEN *Modeling by shortest data description*, Automatica, 14(5):465-471, 1983

[24] ADITYA PRATAPA, AMOGH P. JALIHAL, JEFFREY N. LAW, ADITYA BHARADWAJ, T. M. MURALI *Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data*, Nature methods, 17(2):147-154,2020

[25] ANTONIO SCIALDONE, YOSUKE TANAKA, WAJIAD JAWAID, VICTORIA MOIGNARD, NICOLA K. WILSON, IAN C. MAMCAULAY, JOHN C. MARIONI, BERTHOLD GÖTTGENS *Resolving early mesoderm diversification through single-cell expression profiling*, Nature, 535(7611):289-293, 2016.

[26] SIMONE PICELLI, ASA BJÖRKUNG, OMID FARIDANI, SVEN SAGASSER, GÖSTA WINBERG, RICKARD SANDBERG *Smart-seq2 for sensitive full-length transcriptome profiling in single cellsg*, Nature methods, 10(11):1096-98, 2013.

[27] PHILIP BRENNECKE, SIMON ANDERS, JONG KYOUNG KIM, ALEKSANDRA KOLODZIEJCZYK, XIUWEI ZHANG, VALENTINA PROSERPIO, BIANKA BAYING, VLADIMIR BENES, SARAH TEICHMANN, JOHN MARIONO, MARCUS HEISLER *Accounting for technical noise in single-cell RNA-seq experiments*, Nature, 10(11):1093-1095, 2013

[28] BLANCA PIJUAN-SALA, JONATHAN GRIFFITHS, CAROLINA GUIBENIF, TOM HISCOCK, WAJID JAWAID, FERNANDO CALERO-NIETO, CARLA MULAS, XIMENA IBARRA-SORIA, RICHARD TYSER, DEBBIE LEE LIAN HOA, WOLF REIK, SHANKAR SRINIVAS, BENJAMIN SIMONS, JENNIFER NICHOLS, JOHN MARIONI, BERTHOLD GÖTTGENS *A single-cell molecular map of mouse gastrulation and early organogenesis*, Nature, 566(7745):490-495, 2019

[29] SHUANON CHEN, JESSICA C MAR *Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data*, BMC Bioinformatics, 19(1):232, 2018

[30] JAN KRUMSIEK, CARSTEN MARR, TIMM SCHROEDER, FABIAN THEIS *Hierarchical Differentiation of Myeloid Progenitors Is Encoded in the Transcription Factor Network*, PLoS One, 6(8):e22649, 2011

[31] FRANK WILCOXON *Individual Comparisons by Ranking Methods*, Biometrics Bulletin, 1(6):80-83, 1945

[32] SONIA NESTOROWA, FIONA HAMEY, BLANCA PIJUAN-SALA, EVANGELIA DIAMANTI, MAIRI SHEPHERD, ELISA LAURENTI, NICOLA WILSON, DAVID KENT, BERTHOLD GÖTTGENS *A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation*, Blood, 128(8):20–31, 2016

[33] DAMIAN SZKLARCZYK, ANNIKA GABLE, DAVID LYON, ALEXANDER JUNGE, STEFAN WYDER, JAMIE HUERTA-CEPAS, MILA SIMONOVIC, NADEZHADE DONCHEVA, JOHN MORRIS, PEER BORK, LARS JENSEN, CHRISTIAN VON MERIG *STRING v11: protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets*, Nucleic Acids Research, 47(D1):D607-D613, 2019 https://string-db.org/

[34] RONALD AYLMER FISHER *On the Interpretation of χ^2 from Contingency Tables, and the Calculation of P*, Journal of the Royal Statistical Society, 85(1):87-94, 1922

[35] THEODORE WILBUR ANDERSON, DONALD ALLAN DARLING *Asymptotic Theory of Certain "Goodness of Fit" Criteria Based on Stochastic Processes*, Annals of Mathematical Statistics, 23(2):193-212, 1952

[36] THALIA E. CHAN, 2017 https://github.com/Tchanders/Discretizers.jl

[37] NICLAS POPP, 2020 https://github.com/niclaspopp/MultivariateDiscretization.jl