KU LEUVEN

FACULTY OF BIOSCIENCE ENGINEERING
FACULTY OF ENGINEERING SCIENCE
FACULTY OF MEDICINE
FACULTY OF SCIENCES

Statistical Inference of AI-identified Subcellular RNA Localizations

Promoter:
Prof. Dr. Alejandro Sifrim
Department of Human Genetics
Faculty of Medicine

Dissertation presented in
fulfillment of the requirements
for the degree of Master of Science:
Bioinformatics

Nynke TILKEMA

September 2024

1 ACKNOWLEDGEMENTS

I cannot begin to express my thanks for my supervisors Prof. Dr. Alejandro Sifrim and David Wouters. Thank you for your profound belief in my abilities, your extensive knowledge and your guidance. Special thanks to all members of the Lab of Multi-Omics Integrative Bioinformatics for creating a great atmosphere where we can intrinsically discover science together, the 15:00 table tennis breaks and of course your valuable suggestions for my thesis. Many thanks to all my friends at the Master of Bioinformatics at KU Leuven, especially Anis Ismail, for your support, study sessions, and making the past two years both enjoyable and rewarding. I am also grateful to Alejandro Villaseñor Medina for your unwavering support through the past two years. Lastly, my success would not have been possible without the loving support and nurturing of my family.

Thank you all for contributing to the successful completion of my thesis dissertation.

2 ABSTRACT

The subcellular localization of RNA is crucial for processes such as cell polarization, division, and state, and is implicated in various diseases. Innovations in spatial transcriptomics now facilitate large scale and systematic studies of subcellular RNA localization. However, computational methods that characterize RNA localization are still developing. We enhanced an in-house convolutional autoencoder model that detects RNA localization patterns without manual feature engineering. Our primary goal was to develop a statistical framework to quantify the probability of RNA localization for individual genes across multiple cells. This framework was tested using simulated data and validated with an experimental MERFISH dataset of enterocyte apical-basal polarization in the small intestine.

Two approaches were employed: supervised classification using Random Forests and a latent space (LS)-based approach in which statistical inference of subcellular localization was used directly. For the LT-based approach we tested if two gene point clouds within the latent space significantly differed from each other with a permutation test comparing the Chamfer L1 distance. By aggregating model classifications across cells for each gene, we aimed to establish a robust method for determining gene localization probabilities. Both supervised classification and LS-based approaches effectively identified RNA localization patterns in simulated data with realistic pattern strengths and RNA counts. Pericellular patterns could be discerned from non-patterns and other localization patterns. Validation showed that simulated data results could be generalized to biological and experimental contexts. Both approaches demonstrated high sensitivity, especially with intermediate and strong pattern strengths. A comprehensive power analysis determined necessary sample sizes for varying pattern strengths and dynamic ranges for both approaches. The supervised approach generally outperformed the latent space approach, particularly when considering computational resources.

Our study presents a validated statistical framework for quantifying subcellular RNA localization probabilities across multiple cells, thereby enhancing sensitivity in detecting RNA localization patterns, even subtle ones. Future research should further refine this framework to ensure accuracy and applicability in biological contexts.

TABLE OF CONTENTS

3 List of Abbreviations and Symbols

ALS – Amyotrophic Lateral Sclerosis

AUC – Area Under the Curve

CVAE – Convolutional Variational AutoEncoder

FISH – Fluorescence In Situ Hybridization

FTD – Frontotemporal Dementia

KS – Kolmogorov-Smirnov

LS-based – Latent Space based

MOC – Microtubule Organizing Center

MERFISH – Multiplexed Error-Robust FISH

mRNA – messenger RNA

P-bodies – Processing bodies

RBP – RNA-binding Protein

RNP – Ribonucleoprotein

ROC – Receiver Operating Characteristic

smFISH – single molecule FISH

UMAP – Uniform Manifold Approximation and Projection

3'UTR – 3 prime Untranslated Region

4 List of Figures

5 CONTEXT AND AIMS

Spatial transcriptomics was deemed Nature's method of the year in 2020[1]. Because of these novel technologies, it is now possible to analyze subcellular RNA localization in a systematic and large scale manner. This will allow us to answer interesting fundamental biological questions in a variety of biological domains, in health and disease. However, computational methods to characterize subcellular RNA localization are still in their infancy. We therefore aim to tackle the following questions as part of this master thesis study:

How does one automatically classify whether a gene shows a subcellular localized expression pattern or not?

- Using supervised classification
    - Which classification algorithm is best suited? And how do we train it optimally
    - What is the performance of the optimal model?
    - Can we aggregate model classifications of a gene over every cell, and can we create a reliable statistical test to discern the probability that a gene localizes non-randomly?
    - If we can classify patterns from non-patterns, can we classify which specific pattern it is?
- Can we infer subcellular localization directly from the latent space embedding of an in-house developed neural network model, without training a classifier first?
- Do these results on simulated data generalize to real biological/experimental data?

6 INTRODUCTION

The subcellular localization of RNA is important for essential processes in the cell such as polarization, cell division and cell state[2], and has been implicated in many diseases[3]. It is estimated that the majority of RNA in the cell portrays some form of spatial expression pattern[3], however the evidence so far remains largely anecdotal. Recently, innovations in spatial transcriptomics techniques, allowing the detection of RNA species at the single-molecule level, have opened up the possibility for a systematic study of the subcellular localization. However, the computational techniques needed for this task are still developing, and currently no formal way exists to characterize these patterns. This section therefore will focus on the functions and mechanisms of RNA localization, and describes current methods of experimental and *in silico* detection of RNA localization at a single-cell level.

6.1 RNA SUBCELLULAR LOCALIZATION

Subcellular mRNA localization was first reported in 1983 by Jeffery et al[4] in ascidian eggs. He observed that *actin* mRNA was unevenly distributed in ascidian eggs, with a large proportion of the egg actin mRNA enriched in the myoplasm, which eventually develops into muscle fibers. We now know that subcellular RNA localization is an important form of post-transcriptional gene expression regulation that is conserved across all species (reviewed by Das et al.[2]). This has mainly been explored in mRNA, but emerging studies have demonstrated that it is relevant for the majority of RNA transcripts, including long non-coding RNAs[5,6]. High resolution imaging has shown that RNAs can localize in specific compartments (such as protrusions[7], dendrites[2,8], and the nuclear envelope[9]) and organelles (such as ER[2,10], nucleus[6] , and mitochondria[10]) (Figure 1). In addition, trans-cellular mRNA localization has been observed in *Drosophila Arc1* mRNA, which transfers across synapses between motor neurons and muscle cells[11] . Disruption of this trans-synaptic transport leads to disruptions in synaptic plasticity[3,11] , which could be clinically relevant as genetic mutations in the human Arc1 protein are linked to autism[12] and schizophrenia[13], which are both associated with abnormal neuroplasticity. These studies imply that RNA localization plays a role in post-transcriptional fine-tuning of gene expression and regulating fundamental biological processes like cell movement, polarization and differentiation.

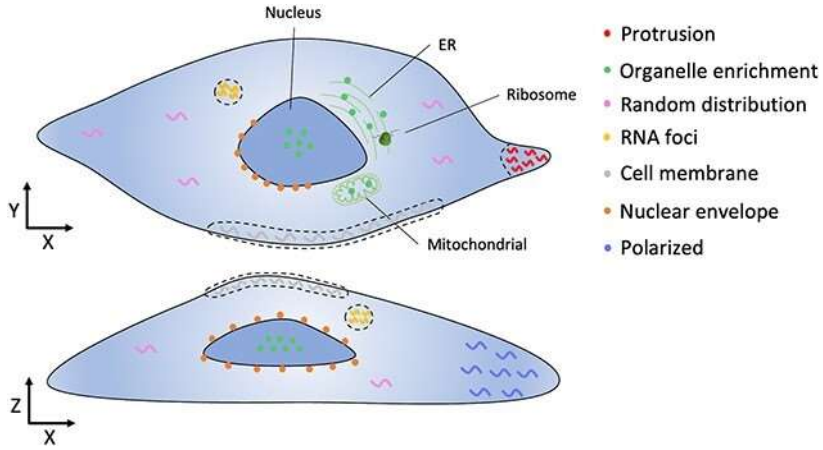

Figure 1: Schematic representations of different RNA localizations. Adapted from [14].

6.2 Why Does RNA Localize?

The functional advantages of subcellular RNA localization can be distinguished between unicellular and multicellular/tissue levels. At the unicellular level, RNA localization controls cell migration, polarity, and cell division through rapid responses to intra- and extracellular cues[2]. Conversely, in multicellular organisms or tissues, RNA localization plays crucial roles in maintaining homeostasis, promoting differentiation and orchestrating development. Interestingly, impairments in mRNA localization seldom lead to lethality or growth impairments in unicellular organisms. Dysregulation in RNA localization in multicellular organisms has been linked to various pathologies, such as in cancer progression[15], neurodevelopmental[16,17] and neurodegenerative disorders[18,19]. Moreover, disrupting RNA localization during *Drosophila* development results in severe developmental defects[20]. The following section details the functional and clinical relevance of subcellular RNA localization.

Subcellular RNA localization enhances protein production efficiency and reduces the risk of protein malfunction. Translating mRNA locally multiple times is more cost-effective than transporting individual proteins, as a single mRNA can generate tens to hundreds of proteins[21]. This is particularly important when molecular diffusion processes become prohibitive for function over large distances, such as in motor neuron axons, which can span in length over one meter. Moreover, mRNA colocalization and local protein synthesis increase the likelihood of successful protein complex formation due to the decreased physical distance between protein subunits[22]. The colocalization of subunit mRNAs could also minimize the risk of protein subunits unintentionally interacting with other proteins during transport to their action site, preventing the formation of nonsensical complexes[3]. Lastly, proteins undergo various modifications throughout their lifetime, including normal changes such as phosphorylation, acetylation and ubiquitination, as well as pathological changes like oxidative damage and aggregation. Producing newly-formed proteins at distal sites helps avoid damage that might occur during transport from the soma, ensuring optimal protein functionality[23].

Subcellular RNA localization enables cells to respond swiftly to environmental stimuli by facilitating local protein synthesis, bypassing the need for long-distance transport of pre-synthesized proteins[23]. This rapid response is particularly critical in neurons, which need to quickly respond to excitatory and inhibitory inputs which each require different protein effectors. Formicola et al. (2019)[24] reviewed how the material properties of neuronal ribonucleoprotein (RNP) granules can change substantially due to synaptic activation on both a transient and sustained timescales. For instance, in the dendrites of cultured neurons, RNP granules degranulate in response to chemically induced long term potentiation. As a result, the granules release *beta-actin* mRNA and ribosomes, which allows for active translation at the activated synapses[8].

In a pathological context, the behavior of proteins like TDP-43 and FUS in amyotrophic lateral sclerosis (ALS) and frontotemporal dementia (FTD) illustrates the importance of this mechanism. These proteins influence the liquidity of transport granules, which enables granules to swiftly exchange RNA and RNA-binding proteins (RBPs) with the cytosol in response to stimuli. In ALS and FTD, mutant versions of FUS and TDP-43 shift the equilibrium of proteins in these granules from a liquid to a more condensed state in response to oxidative stress[18,25]. This shift seems to either exclude RNA from RNP granules or binds the RNA too tightly to be released upon environmental stimuli[23,26], thereby impairing the rapid and localized protein synthesis necessary for optimal cellular function.

Localized mRNA can also perform regulatory roles beyond their coding functions. For instance, the mRNA of the RBP Oskar acts as a scaffold during early oogenesis of *Drosophila melanogaster*, independent of the Oskar protein[27]. Similarly, the untranslated *Tp53inp2* mRNA enhances NGF-TrkA signaling to regulate axon growth in sympathetic neurons by influencing the endocytosis and signaling of the TrkA receptor[28]. Notably, protein and RNA localization does not always coincide. The RNA of *RAB13* localizes to the protrusions of migrating mesenchymal cells, whereas RAB13 proteins are expressed perinuclearly[29]. Research by Moissoglu et al.[29] demonstrated that specific prevention of *RAB13* RNA localization does not affect the protein's localization, but does impair efficient cell migration and the activation of GTPase, highlighting the regulatory function of *RAB13* transcripts. In the context of cancer, *RAB13* RNA is located at the invasive front of leader cells. Given that *Rab13* activity induces the formation of protrusions, its localization facilitates cancer cell metastasis[30]. These examples underscore the diverse and critical roles that localized mRNA can play in various biological processes.

6.3 How Does RNA Localize?

Within the eukaryotic cell, one of the most common mechanisms of RNA localization is through active transport along the cytoskeleton. RNA is most commonly trafficked with active transport within a RNP complex[2], which consists of proteins such as RBPs and their target coding or non-

coding RNA. Trans-acting RBPs recognize and bind to *cis*-elements within the 3'UTR of the RNA[3,31]. These *cis*-elements act as a zip code, marking the destination of the RNA[3]. An RNP complex can transport one or multiple RNAs at the same time, such as *CaMK11alpha*, *Neurogranin* and *Arc* which are transported to dendrites within the same complex[32]. The active transport can be both long-range and short-range, which use different machinery and routes. Long-range travel occurs over microtubules by dyneins and kinesins, whereas short-range travel occurs over actin filaments by myosins[2] . However, active transport costs energy, so for longer distances - particularly traversing the axon[14] - mRNA can hitchhike along with organelles like lysosomes[33], endosomes[34] or mitochondria[35] for long-distance travel through the axon.

A second way RNA localizes is through local entrapment in RNP granules. RNP complexes can be contained within a RNP granule, which is a group of membrane-less organelles found in the nucleus and various cytosolic compartments[36]. Examples include stress granules, processing bodies (P-bodies), and neuronal granules. These RNP granules serve various functions, including mRNA trafficking[32], RNA processing[2] and the temporary storage of mRNAs in a translationally repressed state[36,37]. In migrating mouse fibroblasts, the APC complex anchors RNA granules at the tips of protrusions, resulting in RNA localization[7] . Upon disruption or loss of function of the APC complex, RNAs cease to localize in protrusions[7], underlining local entrapment in RNP granules as a localization mechanism.

Cells utilize RNA localization in stress granules and P-bodies as an adaptive response to stressors such as heat-shock or oxidative stress[38]. RNP granules form through interactions among individual RNPs via protein-protein, protein-RNA or RNA-RNA interactions, and partly through RNA self-assembly[39]. During cellular stress, actively translated mRNAs are released from ribosomes and compacted up to 200-fold[37], leading to a rapid influx of these mRNAs into stress granules[40]. Once the stress stimulus is removed, stress granules rapidly disassemble, and the mRNAs previously trapped in stress granules and P-bodies are translated and degraded at rates similar to their cytosolic counterparts[2,41]. The entrapment of RNA in RNP granules thus provides a dynamic mechanism for subcellular localization, allowing cells to regulate protein synthesis in response to cellular conditions.

Lastly, RNA can be localized through selective degradation or protection. The first two hours of embryogenesis in *Drosophila melanogaster* are programmed by maternally synthesized mRNA's, after which the majority of these mRNAs are degraded[42]. After 2,5-3 hours more than 96% of the maternal *Hsp83* mRNA has been degraded[42]. The remaining *Hsp83* transcripts are locally protected in the germ plasm at the posterior pole, resulting in subcellular localization[42]. This local protection is facilitated via a sequence in the 3'UTR which can be deleted or exchanged by the RBP SMAUG[43]. The *Nos* mRNA is selectively protected through a similar mechanism. SMAUG binds to cytoplasmic *Nos* transcripts, halting their translation and prompting recruitment of the CCR4-NOT complex for mRNA degradation via deadenylation[44]. However, in the germ plasm at the posterior pole the

RBP Oskar displaces SMAUG from *Nos* mRNA's, shielding them from degradation and lifting the translation block[2,45]. This targeted protection results in the subcellular localization of *Nos* at the posterior pole.

Cells often use a combination of the aforementioned localization methods to regulate the transcription and translation of its transcriptome. For example, in budding yeast *ASH1* transcripts localize in the distal bud tip during the anaphase of the cell cycle[46] through active transport within a large RNP[47]. After localized translation, the Ash1 protein acts as a transcriptional repressor of the *HO* gene in the daughter-cell nucleus[48], achieving RNA localization through asymmetric repression. The *HO* gene encodes an endonuclease that initiates mating-type switching, which means mother cells can switch their mating type whereas daughter cells cannot[46]. When any of the proteins responsible for transporting *ASH1* within the RNP are depleted, *ASH1* transcripts are translated prematurely, preventing proper localization to the bud tip[47,48]. This disruption nullifies *HO* silencing, leading to both mother and daughter cells possessing the same mating type[47].

6.4 The Experimental Study of RNA Localization

Hybridization-based techniques, such as Fluorescence In Situ Hybridization (FISH) techniques, provide direct visual evidence of RNA localization within cells, allowing the mapping of their precise location. Single molecule FISH (smFISH) uses fluorescent probes that bind to specific RNA transcripts with high specificity and sensitivity[49], allowing the determination of the subcellular locations of these RNAs. However, smFISH is limited to detecting only a few genes at a time due to the need for fluorescent dyes with distinct spectra.

Multiplexed error-robust FISH (MERFISH) builds upon smFISH by enabling the simultaneous measurement of a much larger number of transcript species with an 80% detection efficiency and a 4% misidentification rate[50]. In MERFISH, each gene in the panel is assigned a unique *N*-bit barcode, which corresponds to a complementary probe with fluorophore labels[51]. The imaging process involves *N* sequential rounds of smFISH, with each round reading out one bit of the binary barcode (Figure 2). A fluorescent signal in the image represents a '1' in the corresponding bit position, while the absence of a signal represents a '0'. Theoretically, *N* rounds of hybridization could allow the detection of up to 2^N - 1 different RNA species. For example, 15 hybridization rounds could theoretically probe over 32,000 genes, enough to cover all human nuclear protein-coding genes[52].

However, with each round of hybridization the RNA slightly degrades and the risk of misidentification increases[51]. To mitigate misidentification rates, MERFISH uses a subset of the total available binary barcodes based on a modified version of the Hamming Distance. MERFISH uses a Hamming distance 4 and prioritizes correcting missed hybridization events over misidentification of

background spots as RNA. This approach provides tolerance for a certain number of misreads before an RNA species is incorrectly identified.

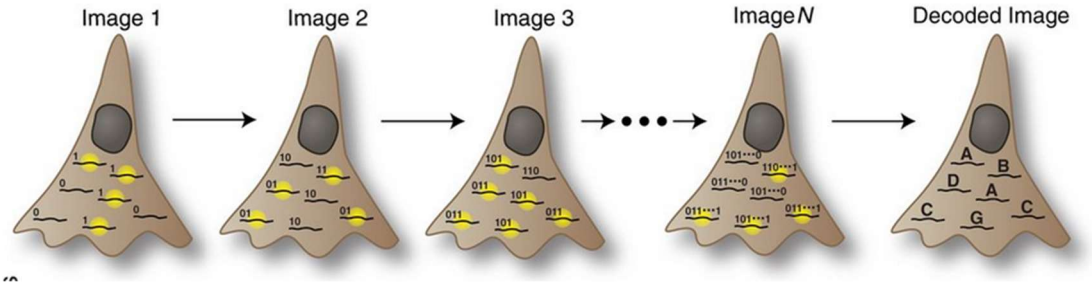

Figure 2: Schematic representation of multiple hybridization rounds of MERFISH. Adapted from [51].

Given that a MERFISH experiment can include up to thousands of genes, assayed in tens to hundreds of thousands of cells, it can be difficult and laborious to identify genes exhibiting subcellular localization by hand. It would therefore be advantageous to automatically detect whether a gene portrays subcellular localization, and if so, what particular type of subcellular localization. Automating this process not only saves time but would also ensure a more objective and reproducible analysis. The knowledge of which genes in a MERFISH panel exhibit subcellular RNA localization enables a more targeted investigation into the functional roles of these localized RNA's. This transition from manual to automated analysis paves the way for high-throughput studies and deeper insights into the spatial organization of gene expression.

6.5 State-of-the-art of RNA Localization Detection

In silico approaches to predict subcellular RNA localization can be broadly classified into sequence-based and image-based methods. Sequence-based approaches utilize RNA sequences and secondary structures to predict localization, leveraging known localizations of similar sequences to forecast new sequences[53]. This method is grounded in the biological principle that RBPs recognize specific binding motifs and secondary structures[3]. Sequence-based models can be classified into two categories: those predicting a single localization for an RNA sequence[54–57] and those predicting multiple possible localizations[53,58–61]. By identifying RNAs with predicted localizations, one can prioritize targets for experimental validation and focus their efforts on the most promising candidates.

Image-based approaches often use the precise RNA locations within cells provided by FISH images to automatically detect RNA localization. Currently there are seven studies proposing methods to automatically detect subcellular RNA localization based on FISH experiments. Four of these studies originate from the two groups that co-developed FISH-quant[62], a commonly used toolbox which

includes simFISH to generate simulated FISH images with different localization patterns. Unsurprisingly, the majority of these 7 studies used simFISH to generate simulated datasets, as labeled/ground-truth experimental data is hard to come by.

6.5.1 Methods using manually curated features

Five out of seven studies used manually curated features to detect RNA localization detection. Three of these studies - Bento[63], Samacoits et al.[64], and Chouaib et al.[65] - employed random forest machine learning classifiers to detect subcellular RNA localization based on manually curated feature vectors. Each study turned (simulated) smFISH data into feature vectors of varying lengths: Bento used 13 features, Samacoits et al. used 23 features, and Chouaib et al. used 15 features. These features commonly included metrics such as polarization, point density throughout the cell, and the distance of each RNA transcripts to various cellular landmarks. While Samacoits trained a multilabel RF classifier with five patterns (foci, protrusion, intracellular, nuclear envelope, random), Bento Toolbox and Chouaib et al. used five binary RF classifiers to be able to assign multiple labels per observation.

A closer examination of the evaluation strategies and dataset annotations employed by Bento and Chouaib et al. indicated potential gaps in the transparency and robustness of their reported results. Although Bento's methods mentioned the manual annotation of RNA localization from 165 seqFISH+ and 238 MERFISH samples for validation of the RF classifier, the results section did not provide performance details on this ground-truth dataset. Chouaib et al. used an imbalanced training set with a one-versus-all strategy to train their binary classifiers. However, it was unclear whether a train-test split was performed or if the genes shown in the results were part of the training set.

Two studies - pointFISH[66] and DypFISH[67] - incorporated manual features alongside experimental data in their overall model. PointFISH[66] utilizes both manual features and RNA point clouds as inputs for their attention-based artificial neural network model. The method employed five features: the occurrence of foci, and the distance and position of each RNA relative to the cellular and nuclear membranes. In order to classify subcellular RNA localizations, a support vector classifier was trained on the attention-based model embeddings. This classifier was compared to the Chouaib et al.[65] classifier, achieving similar performance for each pattern. PointFISH achieved an F1-score of 0.95 on the simulated test dataset and 0.82 on the ground-truth smFISH test dataset. The difference in F1 scores could be partly attributed to the model's lack of exposure to protrusion localizations during training. Additionally, pointFISH did not support multi-label predictions, whereas the ground-truth smFISH data included genes with multiple localizations within the same cell, such as foci near the nuclear envelope.

Tools	Model	Training Data	RNA count per training sample	Pattern Strength	Training samples per pattern	Input Features	Localizations	1 gene multiple cells tested?	Multi-label prediction?	Last github commit	Most recent interacted Github issue
Bento (Mah et al. 2022)	5 binary RF on input features	Simulated with simFISH	5-300	10, 50, and 90% pattern strength	2000	13 input features	Nuclear, cytoplasmic, nuclear edge, cell edge, no pattern	No	Yes	Apr/24	Apr/24
FISHFactor (Walter et al. 2023)	Factor analysis adapted for spatial transcriptomics data	Simulated with in house model	NA	5 intensities, ranging from low till strong	10	#factors	Dataset dependent, validated on nucleus, cytoplasm and protrusion	Yes	No	May/23	Oct/23
PointFISH (Imbert et al. 2023)	Support vector classifier on attention-based network embeddings	Simulated with simFISH	50-900	60-100% pattern	20,000	5 input features	random, foci, intranuclear, nuclear edge, perinuclear, protrusion	No	No	Aug/22	NA
Rfclassifier (Samacoits et al. 2018)	Multiclass RF on input features	Simulated with simFISH	100-200	Moderate and strong	?	23 features	nuclear envelope, intranuclear, protrusion, foci, random	Yes	No	Jul/22	Oct/23
CNN workflow (Dubois et al. 2019)	CNN: SqueezeNet	Simulated with simFISH	100-1000	Varied pattern strengths	28,500	#RNA per pixel Nucleus&cell staining	No pattern, protrusion, nuclear edge, intranuclear, foci, polarized, cell edge	No	No	NA	NA
Rfclassifier (Chouaib et al. 2020)	5 binary RF on input features	labeled smFISH	30+	NA	320-750	15 features	Foci, protrusion, perinuclear, nuclear edge, intranuclear	Yes	Yes	Jul/22	Oct/23
DypFISH (Savulescu et al. 2021)	Various tools, incl. per-quadrant statistics and colocalization	smFISH	?	NA	?	cell+nucleus boundaries, position of MOC	Foci, protrusion, de novo	Yes	No	Jun/23	Nov/21

Tabel 1: Overview of RNA localization detection studies. NA indicates not applicable, a question mark denotes we were not able to find the information.

The DypFISH[67] method used manually annotated landmarks and smFISH pictures of standardized cell shapes to analyze the spatial distribution of mRNA and proteins. The cell shapes were positioned in a uniform manner with the use of micropatterning, ensuring that each cell maintains the same shape. As input, DypFISH required RNA and protein images from smFISH and immunofluorescence, with cell stainings for cell boundaries and nuclei, and manual annotation of the microtubule organizing center (MOC). The MOC was used to overlap the uniform cells and average them per time point. The cells were divided into quadrants using isolines radiating from the MOC to calculate mRNA and protein concentrations in each quadrant. Colocalization scores for RNA and protein were then calculated using Ripley's K function. Additionally, the average distance of cytoplasmic mRNA from the nuclear envelope was measured to indicate the extent of cytoplasmic spread. However, manually annotating the MOC is labor-intensive, and the method is less effective for cells that do not conform to standardized shapes achieved through micropatterning, so it is unlikely to perform well *in vivo* for tissues other than muscle tissue.

6.5.2 Methods without manually curated features

A study by Dubois et al. (2019)[68] introduced a deep learning approach to identify mRNA localization patterns, demonstrating a proof-of-principle for the use of deep neural networks to identify subcellular localization patterns without relying on handcrafted features. The study focused on seven localization patterns: no pattern, protrusions, nuclear edge, intranuclear, foci, polarized, and cell edge. The data preprocessing involved decomposing foci into individual RNA, counting the RNA per pixel, and a nucleus and cell boundary staining. The authors utilized the SqueezeNet architecture[69], a fully convolutional neural network which produces 512 feature maps. These feature maps were then processed through a final 1x1 convolutional layer to classify the seven localization patterns, and a softmax activation function was applied to produce the probability for each localization pattern. The model achieved an overall accuracy of 91% on an independent simulated test set, with the lowest F1 score being 0.77 for distinguishing non-patterned versus patterned observations.

FISHfactor[70] adapts factor analysis for spatial transcriptomics data by modeling RNA molecule coordinates as a spatial Poisson point process. This process is represented as expression intensity, where a higher intensity indicates a higher likelihood of a molecule being present at that location[71]. The expression intensity is then factorized into non-negative factors and weights to achieve a biologically meaningful interpretation. While weights remain consistent across cells, factors are cell-specific. FISHfactor does not predefine the localization patterns, allowing for the discovery of *de novo* patterns. However, the number of factors must be defined in advance, making it not entirely hypothesis-free as the expected number of different patterns must be predicted. The model was tested on three relatively distinct patterns, without including a non-pattern control. One limitation is that the model was validated on a test set with 60 genes in 20 cells and does not seem to scale well

with an increasing number of cells, genes and factors. It is therefore unclear whether this model would be suitable for MERFISH datasets.

6.5.3 Gene localization detection across cells

Rather than observing RNA localization at a single-cell level, it would be interesting to determine whether a gene consistently exhibits a pattern across various cells, rather than in isolated observations. Four studies have been developed to detect gene localization across cells, each with distinct approaches. FISHfactor utilizes a factor model, where the shared weights form a gene-by-factor matrix, allowing for the readout of gene patterns across all cells[70]. DypFISH[67] aligned and averaged their micropatterned cells to analyze mRNA/protein colocalization across cells. Samacoits et al.[64] used heatmaps to depict the majority voting results of the RF classifier for each gene across all cells. They furthermore employed Gini impurity on the average posterior probabilities from the RF at both the single-cell and gene levels to assess mRNA localization heterogeneity. Building upon Samacoits et al.'s[64] heatmap visualizations, Chouaib et al.[65] introduced two key modifications: allowing observations to display multiple patterns simultaneously and incorporating statistical testing. The authors used the RF posterior probability scores with a threshold of 0.5 to label observations as patterned, and applied Fisher's exact test to compare the frequency of pattern labeling between test genes and non-patterned control genes. Unlike Samacoits et al.[64], Chouaib et al.[65] did not use the Gini impurity test.

6.5.4 Current gaps

There are several notable gaps in the field of subcellular RNA localization detection. Most models rely on numerous features[67], which necessitate manually intensive work. Additionally, many models make unrealistic assumptions for biological data, such as using RNA counts that are excessively high. For instance, in the host-lab's experience genes generally express 1-2 RNA molecules per cell, and genes with RNA counts exceeding 100 are rare. However, Bento[63] was the only study that included RNA counts lower than 30, while the majority had average RNA counts of 150 or higher (see Table 1), misrepresenting a biologically realistic setting. Moreover, most models address RNA localization as a single-label multi-class task, ignoring the fact that RNA can localize to multiple localizations simultaneously[7]. Only two models currently account for this complexity[63,65].

Another limitation is the focus on global trends. It would be useful if models would assess the probability of a gene exhibiting specific patterns across multiple cells. Although four models perform gene localization across cells[64,65,67,70], most rely heavily on handcrafted features. Moreover, FISHfactor is unsuitable for larger datasets, dypFISH is mainly applicable for micropatterned cells, Samacoits et al. only did an exploratory analysis, and Chouaib et al. used binary classification of RF posterior probability which reduces its power.

In order to address these gaps in the field, the host research group has built a convolutional variational autoencoder (CVAE) that automatically detects RNA localization at a single-cell level without requiring manual feature input. This thesis aims to enhance the CVAE model by creating a statistical framework that quantifies the probability of RNA localization for individual genes across multiple cells. To achieve this, we will use simulated data that better reflects biological reality and validate the findings with a ground-truth experimental MERFISH dataset. For the statistical framework we will consider two approaches: supervised classification and using the latent space embedding directly. By aggregating the model classifications across all cells for each gene, we aim to establish a test to determine the probability of gene localization. Additionally, this thesis will explore whether we can differentiate specific patterns from non-patterns and accurately classify them. Lastly, we will determine if results obtained from simulated data can be generalized to biological and experimental contexts.

7 METHODS

All analyses were conducted using Python 3.9.13. The packages utilized at each step are documented in a YAML file available on the GitHub repository, which contains all the software code used in this thesis (see section 10). Unless otherwise specified, all figures were created using the Seaborn, scikit learn, and Matplotlib packages.

7.1 DATA OVERVIEW

7.1.1 Simulated dataset

The simulated dataset was generated using simFISH within the FISH-quant tool[62]. Each observation was defined as a two dimensional point cloud, representing the set of observed RNA transcripts of a gene within a specific cell. Observations were generated for nine different patterns: foci (RNA aggregates, e.g. granules), intranuclear (inside the nucleus), extranuclear (in the cytoplasm), nuclear edge (near the nuclear membrane), cell edge (at the cellular membrane), perinuclear (on one side of the nuclear membrane), pericellular (polarized to one side of the cell), protrusion (in cell extensions), and non-patterned. The non-patterned localization was modeled as a Poisson point process, where all points occur independently[72].

For each observation, the number of RNAs varied between 1 and 150 to simulate differences in expression levels. For the ease of visualizations, we binned the RNA counts into 5 bins: 0 - 10, 10 - 30, 30 - 60, 60 - 100, and above 100. Furthermore, different pattern strengths were simulated: low, moderate and strong levels (Figure 3). Pattern strength was defined as the percentage of RNA transcripts in an observation that localize in their pattern, with the remainder of transcripts randomly distributed through the cell. The strength levels were implemented as defined in the simFISH v2 tutorial: low, moderate and strong pattern strength levels, with moderate pattern strength assumed to reflect biological samples. Briefly, all patterns except protrusion and non-pattern had pattern strengths of 10%, 50%, and 90% of spots within pattern, whereas protrusion had pattern strengths of 5%, 25%, and 45%. As for non-pattern localizations, given that all of its RNA transcripts are randomly distributed rather than a percentage of transcripts, the concept of pattern strength does not apply and is therefore not assigned.

In total, approximately 114,000 samples were simulated for protrusions and 189,100 samples for all other patterns, resulting in 1,626,800 training samples. The simFISH software generates observations from 317 smFISH images of HeLa cells[62]. To prevent our models from learning these specific cell shapes, an 80-20% train-test split was created based on cellular identities. This means that observations from 254 cell shapes were used for the training set, while observations from the remaining 63 cell shapes were used for the test set.

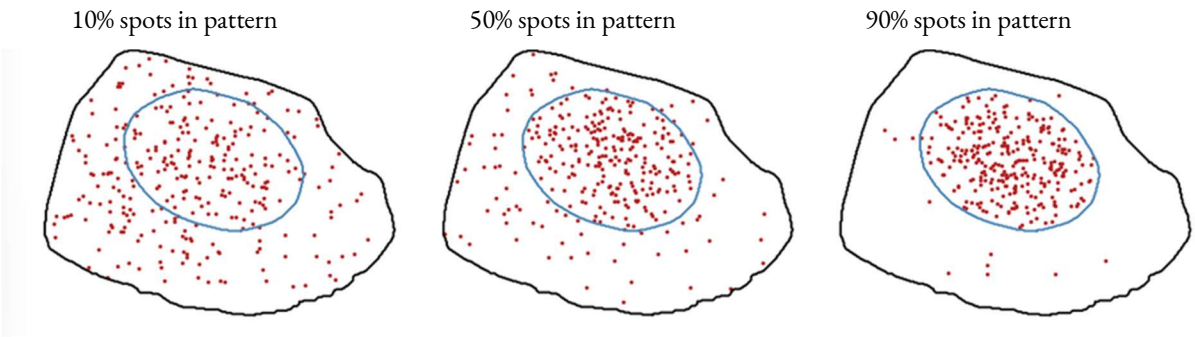

Figure 3: Examples of simFISH simulations of low, intermediate and strong pattern strengths. Adapted from [62].

7.1.2 Simulated genes

To train a framework to assess RNA localization of a gene across cells, annotated ground-truth genes were required. However, the simulated dataset only provided predefined cellular identities, not distinct genes, which is why genes were created from the simulated dataset. The overall approach involved using the entire Anndata object, which stored the simulated dataset, and filtering it based on the desired localization patterns, pattern strength, RNA counts. Pandas.sample() was utilized to obtain a random sample from the filtered Anndata object, with the sample size corresponding to the desired number of cells for the simulated gene. Sampling was done using a random seed unless specified otherwise, and the sampled Anndata object was then returned as the simulated gene.

Four types of genes were generated: specific-pattern genes, mixed-pattern genes, other-pattern genes, and non-patterned genes. These gene types differed in two of the four filtering criteria: localization pattern and pattern strength. The RNA count and number of cells were consistent across all gene types. The pattern strength varied as follows: genes with a pattern (specific-pattern, mixed-pattern, and other-pattern genes) were filtered based on the inputted pattern strength, while non-patterned genes were not filtered by pattern strength, as it did not apply to them. For localization patterns, the specific-pattern and non-patterned genes only included one specific pattern, while the mixed-pattern and other-pattern genes included multiple localization patterns.

- **Specific-pattern gene:** Created by filtering the dataset for one specific localization pattern (e.g., pericellular)
- **Non-patterned gene:** Created by filtering the dataset to include only non-patterned observations.
- **Mixed-pattern gene:** Created by excluding non-patterned observations, resulting in a gene with a mix of all patterned localizations.
- **Other-pattern gene:** Created by filtering out non-patterned observations and one specific localization pattern of choice.

7.1.3 MERFISH dataset

To validate our methods using experimental data, a MERFISH dataset provided by the Laboratory for Systems Physiology at ETH Zurich[73] was employed. This dataset examined 500 genes across 408 cells for subcellular RNA localization patterns in the small intestine. Six of these genes—*ApoB*, *CDH1*, and *CDKN1A* (apically localized) and *CYB5R3*, *PIGR*, and *NET1* (basally localized)—are well-known for their subcellular localization in enterocytes, making them suitable as ground-truth genes for our analysis. Additionally, we chose two genes, *CLCA4a* and *SLC39A14*, which showed no apparent localization patterns upon visual inspection. These genes served as exploratory controls to assess whether our methods would indiscriminately flag most genes as patterned or demonstrate a higher level of selectivity.

7.2 CVAE Architecture

The host research group developed a CVAE for the automatic detection of subcellular RNA localization. The model's inputs were single-cell images generated from the simulated dataset, which included a nuclear boundary and where the image boundaries corresponded to the cellular boundary. Initially, each image was represented as a 100x100 array with RNAs marked by a value of 1. These images were subsequently processed with a gaussian blur (sigma = 1.5), normalized, and augmented with rotations to ensure rotational invariance in the encodings. A train-test split was performed based on cell identities, using all images from 80% of the unique cell shapes for training and the remaining 20% for testing.

The CVAE's architecture comprises an encoder with four convolutional layers followed by a fully connected layer that maps to a 15 dimensional latent space represented as a distribution. Decoding involves sampling from this distribution, then passing through another fully connected layer and four convolutional layers. Additionally, a single-layer linear classifier evaluated the 15-dimensional embeddings to determine pattern presence, encouraging the model to prioritize learning this feature. The loss function combines the Kullback-Leibler loss, binary classifier loss, and reconstruction loss. The CVAE was trained on the complete training set using the ADAM optimizer[74], with a batch size of 256, over a maximum of 100 epochs. All data described in this thesis were generated by embedding the original datasets using this trained model.

7.3 Quantifying Pattern Presence of a Gene

To determine the pattern presence probability of a gene for subcellular RNA localization, a two-round approach was adopted. In the first round, the presence of a localization pattern was assessed for each gene. If a pattern was detected, the second round would identify the specific subcellular localization pattern. This two-round method offered distinct advantages over a single-round approach, where specific patterns were directly compared to a non-patterned control, as

implemented by Chouaib et al.[65] Firstly, the first round could be used as a proof-of-principle, demonstrating that latent space embedding could effectively determine pattern presence. Secondly, conducting both analyses in a single round would restrict detection to the specific patterns included in the statistical framework, potentially missing *de novo* patterns. Even if these *de novo* patterns were not classified into predefined categories in the second round, one could manually examine these observations to identify new types of localizations.

7.3.1 Supervised classification

The scikit-learn package[75] was utilized to train, create and optimize supervised classifiers. Two methods were considered: Random Forest (RF) and K-nearest Neighbors (KNN). All models were trained and validated on observations with strong pattern strength. Hyperparameter optimization was conducted for both classifiers in the first round. The classifier with better performance during round one was further explored in the second round to identify specific localization patterns. For the second round classifiers, non-patterned observations were excluded, and separate binary classifiers were trained for each specific pattern to determine the probability of an observation belonging to that pattern versus any other pattern.

For hyperparameter selection and model comparison, the area under the curve (AUC) and F1 metrics were considered. The F1 score, the harmonic mean between precision and recall, prioritizes true positives over true negatives but requires a specific threshold (typically 50%) for classification. This threshold can cause information loss by enforcing a binary decision. In contrast, AUC measures model performance across all possible thresholds, assessing the tradeoff between the true positive rate and false positive rate, preserving more information. Additionally, AUC accounts for both positive and negative cases, providing a balanced assessment of the model's performance. Given these advantages, AUC was used to train and evaluate our supervised classifiers, ensuring a robust performance assessment.

For the Random Forest, hyperparameters considered included the maximum size of a random subset of features (i.e. latent space dimensions) when splitting a node, and the number of trees. The optimal number of features was determined using GridSearchCV with 100 trees (2, 3, 4, 5 and 6), followed by optimization of the number of trees (50 to 500 in increments of 50). The grid for the features was chosen using the square root of the latent dimensions with 2 values above and below, resulting in a grid of 2, 3, 4, 5 and 6 features. For the KNN, feature scaling of the latent dimensions was performed using min-max scaling, given their different variations (see Figure 5), and optimized the choice of k. Odd values of k were used to avoid classification ties, centered around the square root of the total data points (419), with five values above and below this point at intervals of 50. Sklearn caching[75] was used to efficiently perform the KNN grid search with cross validation.

In the first round, classifiers were trained and hyperparameters were optimized for pattern presence testing. These optimized settings were applied to classifiers in the second round. The effect of dataset balancing on training performance was compared to that of the unbalanced dataset. For balancing, the dataset was down sampled to equalize the sizes of pattern and non-pattern classes, maintaining the ratios between different patterns, pattern strengths, and different RNA counts within the pattern class.

To quantify pattern presence of a gene across cells, each observation was assigned a posterior probability using the supervised classifier. Probability density functions were then created for the test gene and its non-patterned control gene, comparing these probability density functions using a two-sample Kolmogorov-Smirnov (KS) test. This non-parametric test evaluates whether two samples originate from the same probability distribution by comparing their cumulative distribution functions. The maximum absolute difference between these curves is compared to the expected distance if the samples were from the same underlying unknown distribution. After the first classification round, the specific pattern can be determined during the second round, for which a separate binary classifier was created for each specific pattern.

7.3.2 Latent space based

For the latent space approach, from now on referred to as the LS-based approach, the pattern presence of a gene was tested directly from the latent space. Given that the observations reside within a 15 dimensional latent space, test and control genes could be considered as point clouds within this space. This means the similarity between these two points could be computed to assess whether the test gene is significantly different from the control gene. Two components were needed to test whether two genes significantly differ from each other: a statistical test and a similarity measure. We focused on the first round of classification in which pattern presence was determined, to serve as a proof-of-principle of this LS-based approach.

For the statistical test, the permutation test was chosen: a non-parametric test where the null hypothesis (H0) states that the test gene and the non-patterned control gene originate from the same underlying distribution, while the alternative hypothesis (H1) states they do not. The cell count of the two genes was matched, as the permutation distribution is sensitive to unbalanced sample sizes[76]. Our permutation method was based on the permutation test from the SciPy stats package[77], adapted for our 15-dimensional latent space. Briefly, the point clouds of the two input genes were concatenated and randomly reassigned observations to either the test or control gene, computing the similarity metric each time. This process was repeated 9999 times to generate a distribution of the similarity under the null hypothesis. The p-value was then calculated as the proportion of samples with a distance equal to or greater than the observed distance, where the observed statistic was always included in the null distribution[78,79]. For genes where the number of permutations (9999) exceeds the binomial coefficient of (n choose k), an exact test was performed instead.

For the similarity measure, the Chamfer distance was chosen over the Earth Mover's Distance, both of which are commonly used to train point cloud generators[80]. The Chamfer distance can be understood as the mean L1 distance from each point in point cloud 1 to its nearest neighbor in point cloud 2, and vice versa. The sum of these two mean distances is the Chamfer distance. Conversely, the Earth Mover's Distance computes the least expensive way to transform one point cloud into another, but is substantially more computationally intensive[81]. Therefore, we selected the Chamfer distance as our similarity metric.

7.3.3 Power analysis

The power analysis was conducted following the methodology of Baumgartner & John Kalassa[82]. Specifically, cell counts (sample sizes) were evaluated for 14 rounded counts on a log10 scale ranging from 5 to 7000. For RNA counts, the binned RNA counts described in section 7.1.1 were used, and three levels of pattern strength were considered: low, intermediate and strong. For each combination of cell count, RNA count and pattern strength, 1000 random test and control genes were obtained using the simulated gene function without a random seed, ensuring each gene was unique. For each of the 1000 test-control pairs, test statistics were determined. The two-sample KS test was used for supervised classification, and a permutation test was used for the LS-based approach. These results were compared against critical values of 0.05 and 0.00001 (Bonferroni correction for 5000 tests). The number of times the null hypothesis was rejected out of the 1000 samples was counted and then divided by 1000 to determine the power for each combination.

8 RESULTS

In this study we aimed to develop a statistical framework to determine the probability that a gene exhibits significant subcellular RNA localization. To achieve this, we used the FISH-quant simulation framework to generate ground-truth data[62] (methods section 7.1.1). This data was then fed into the CVAE, and we developed our statistical framework within the latent space embedding. First, we explored the feature distribution within the latent space (Section 8.1). Following this, we developed two methods for automatically classifying the pattern presence of a gene (Section 8.2). Finally, we assessed whether our statistical framework is applicable to experimental MERFISH data using ground-truth genes (Section 8.3).

8.1 LATENT SPACE EXPLORATION

In order to visualize the feature distribution of the CVAE latent space, we projected the 15-dimensional embedding onto two dimensions with a UMAP projection[83] (Figure 4), where a single point represents the subcellular pattern of 1 gene in 1 cell, from now on referred to as an observation. Observations with the same localization pattern clustered in the same region within the embedded feature space, whereas observations with different patterns are located in different regions (Figure 4A). The observations do not appear to cluster based on the cellular identities (i.e. different cell shapes) used to generate the simulated data (Figure 4B), suggesting cellular identity was not learned as a feature by the embedding. On the other hand, the absolute count of RNA molecules of an observation did cluster in different regions of the embedded feature space (Figure 4D). Observations with lower counts were located in the bottom right with the highest counts located at the top of the UMAP.

As for the strength of the pattern, given that non-patterned observation do not have a pattern strength (methods), these observations were excluded from Figure 4C for clarity. Observations with the same pattern strength clustered together (Figure 4C), with a gradient from low pattern strength in the center of the plot to a strong pattern in the extremities on the left and right. Interestingly, observations with a low pattern strength (≤10% of RNAs are in pattern) (Figure 4C) colocalized with observations without a pattern (Figure 4A), which suggests the CVAE assigns similar features to both.

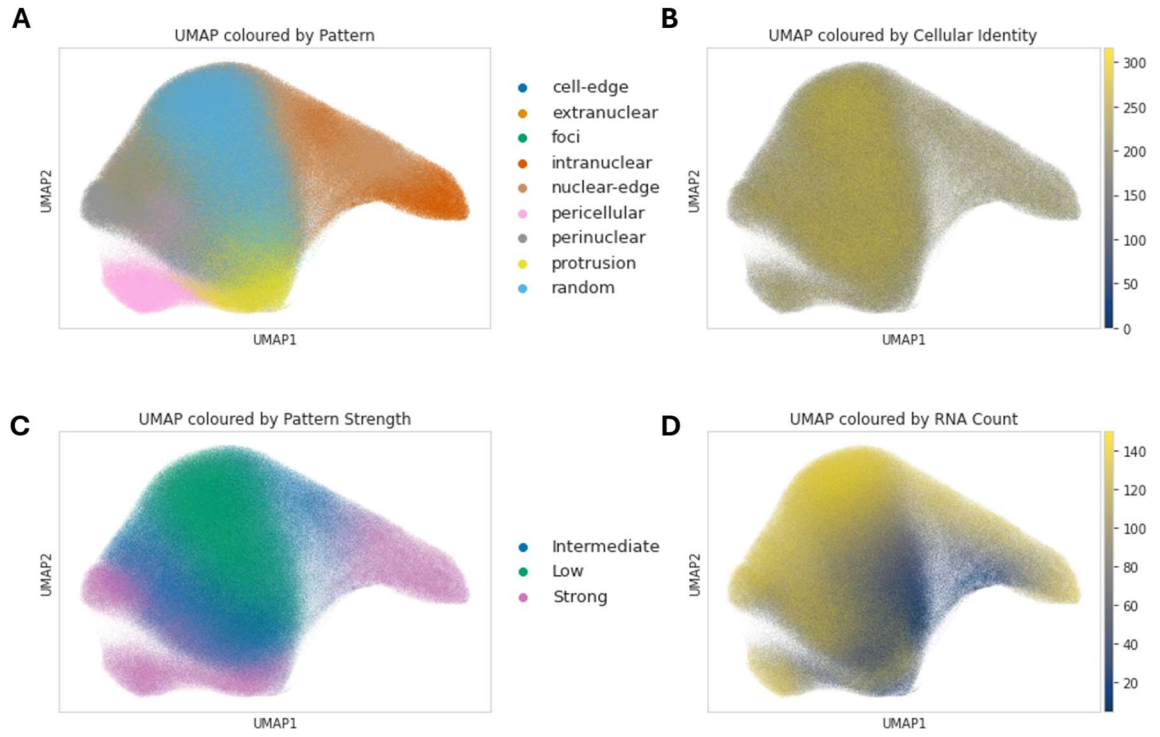

Figure 4. UMAP dimensionality reduction plots of the CVAE latent space: Every dot represents a 2-dimensional representation of a gene expression pattern in a given cell colored by (A) localization pattern type, (B) cellular identity, (C) pattern strength, (D) RNA count.

Regarding the latent dimensions, in theory, they should all have the same range of values due to the regularization imposed by the Gaussian priors on the latent space. However, in practice, we observed some variation between the dimensions (Figure 5). For example, dimension 5 and 6 had different mean values, while dimensions 6 and 14 exhibited different variances. To assess whether we should use non-parametric or parametric statistical tests, we tested whether the dimensions were multivariate normally distributed using Henze-Zirkler's multivariate normality test in the Pingouin package. The test rejected the null hypothesis ($W = 3.30$, $p = <0.001$).

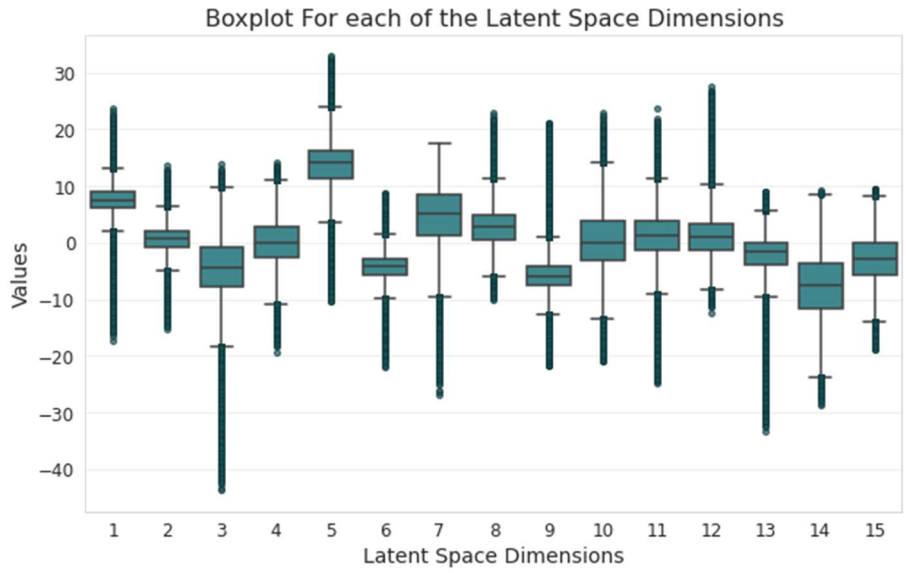

Figure 5. Distributions of CVAE latent space dimensions: Each boxplot on the X-axis represents the summary statistics for one of the CVAE latent dimensions. The horizontal line represents the median value, the box represents the Q1-Q3 interquartile range, upper and lower whiskers represent the 95% confidence interval and points represent outlier observations.

8.2 Quantifying the Pattern Presence of a Gene

We explored two different approaches to classify patterns from non-patterns: using the classification scores of a supervised learner and comparing distribution similarities directly within the latent space. We tested both approaches with experimental data to determine their effectiveness in detecting patterns in actual MERFISH experiments, specifically focusing on genes known to exhibit apical-basal in enterocyte RNA localization patterns.

8.2.1 Using supervised classification

8.2.1.1 Choice of classification algorithm

Given the challenges posed by correlated and non-normally distributed latent dimensions, we selected RF and KNN as suitable classifiers for our analysis. The hyperparameters for both RF and KNN were tuned using an exhaustive grid-search with 5-fold cross validation implemented in the sci-kit learn package[75]. The KNN model required feature scaling (min-max normalization, see methods) due to the non-normal latent dimensions, as discussed in section 7.3.1. Optimal hyperparameters for the RF included 3 maximum features for splitting nodes and 200 trees in the forest. For the KNN, the optimal number of neighbors was set to 69.

Both models achieved great results on the test set with AUC scores (Figure 6), suggesting that the classification results were reproducible across different model types. We chose to continue our analysis with the RF model because random forests are more robust against noisy data with outliers compared to KNN.

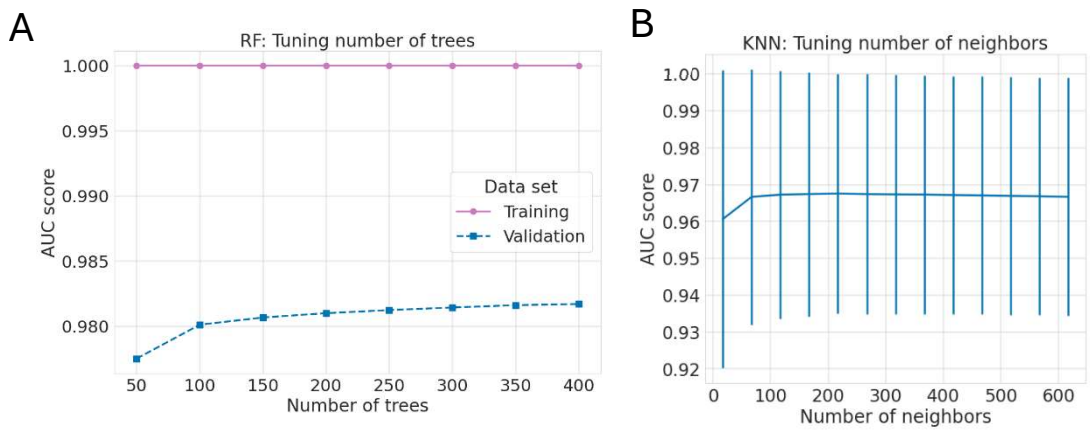

Figure 6. Hyperparameter tuning for the Random Forest and KNN classifiers: (A) Line plot representing the relation between the AUC scores (Y-axis) and the number of trees in the Random Forest ensemble (X-axis), (B) Line plot representing the AUC scores (Y-axis) in relation with the number of neighbors in the KNN.

8.2.1.2 Factors affecting classifier training

8.2.1.2.1 Impact of balancing the training data

We investigated the impact of balancing the training data on the performance of our classification models in the context of a substantially imbalanced simulated dataset. The dataset exhibited a notable 8:1 ratio between the patterned and non-patterned classes. Given that most classification algorithms aim to minimize the error rate, this imbalance could lead the classifier to disproportionately label observations as patterned, the more frequent class. To address this, we tested the influence of balanced versus unbalanced training data on the model's performance.

When we compare the ROC curves (Figure 7A), the AUC was identical for both balanced and unbalanced models. However, the confusion matrices (Figure 7B-C) revealed notable differences. The balanced RF predicted fewer observations as 'patterned' than the unbalanced RF, regardless of the true label. Specifically, the balanced RF had twice as many false negatives and 3.4 times fewer false positives. Based on these results, we chose to train the RF on the balanced data, prioritizing a more conservative approach in labeling observations as patterned to reduce the incidence of false positives.

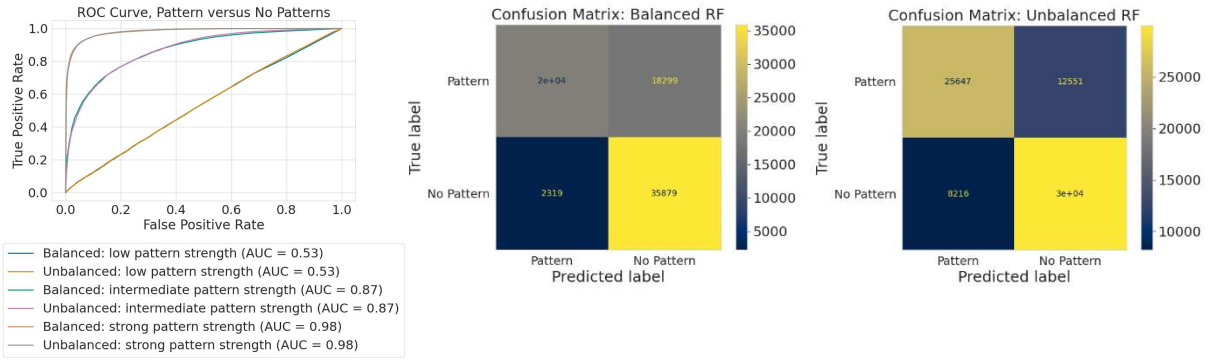

Figure 7. Classification performance for balanced and unbalanced RF classifiers for pattern presence: (A) Receiver operator characteristic (ROC) curves performances for balanced versus unbalanced training sets. (B) Confusion matrix for a RF classifier using a balanced training set, (C) Confusion matrix for a RF classifier using an unbalanced training set. Both (B) and (C) used a balanced testing set with all pattern strengths.

8.2.1.2.2 Impact of cellular identity on training the data

As 317 images of cells were used as a template to generate thousands of simulated genes (methods), we explored the impact of this cellular stratification on the extent of overfitting of the RF classifier and the embedding itself. Firstly we examined the impact of cellular identity on the Random Forest, by performing two different train-test splits. In one split, all observations with a particular cellular identity were assigned either to the train or the test set, ensuring that no cellular identity appeared in both sets. In the other split, cellular identity was not considered while dividing the dataset. The models trained and tested based on split per cellular identity showed a slightly worse AUC score, but overall, the difference between the two splits was negligible indicating that the classifier did not overfit on cellular identity. To enhance the external validity of our model for other datasets, we decided to continue using train-test splits based on the cellular identity.

Next, we investigated the impact of cellular identity stratification on the embedding by training two Random Forests. The first RF was trained on balanced data from an embedding that was trained on all 317 cellular identities (referred to here as the Mixed cell-id embedding), whereas the other RF was trained with balanced data from an embedding that was trained on 80% of the cellular identities (see methods, referred to here as the Split cell-id embedding). The remaining 20% of cellular identities were then projected onto the embedding with the pretrained encoder and served as the test dataset. The test set of the Mixed cell-ID embedding model used the same 80% of cellular identities to select the train set for this model, with the remaining 20% of cellular identities forming the test set. Training the embedding on all cellular identities or 80% of them influenced the RF probability scores assigned to the test sets. Both models performed well on the test data obtained from the same embedding (Figure 8), showing the models did not learn to identify pattern presence based on cellular identity.

As a negative control, we were interested in what the two embeddings would do with a completely random input. Given that both embeddings have completely different feature distributions, giving the test set of the split cell-ID embedding to the Mixed cell-ID embedding model (and vice versa) should therefore be completely random data for the RF. As shown in Figure 8, the RF scores the

patterned and non-patterned observations from this 'random' test dataset on the decision boundary (i.e. scores around 0.5), suggesting both models exert caution in labeling unknown data. Given that the split cell-ID embedding is theoretically more sound and performs well on cellular identities that neither the model nor the embedding had encountered before, we proceeded with this embedding. This approach should ensure robustness and generalizability.

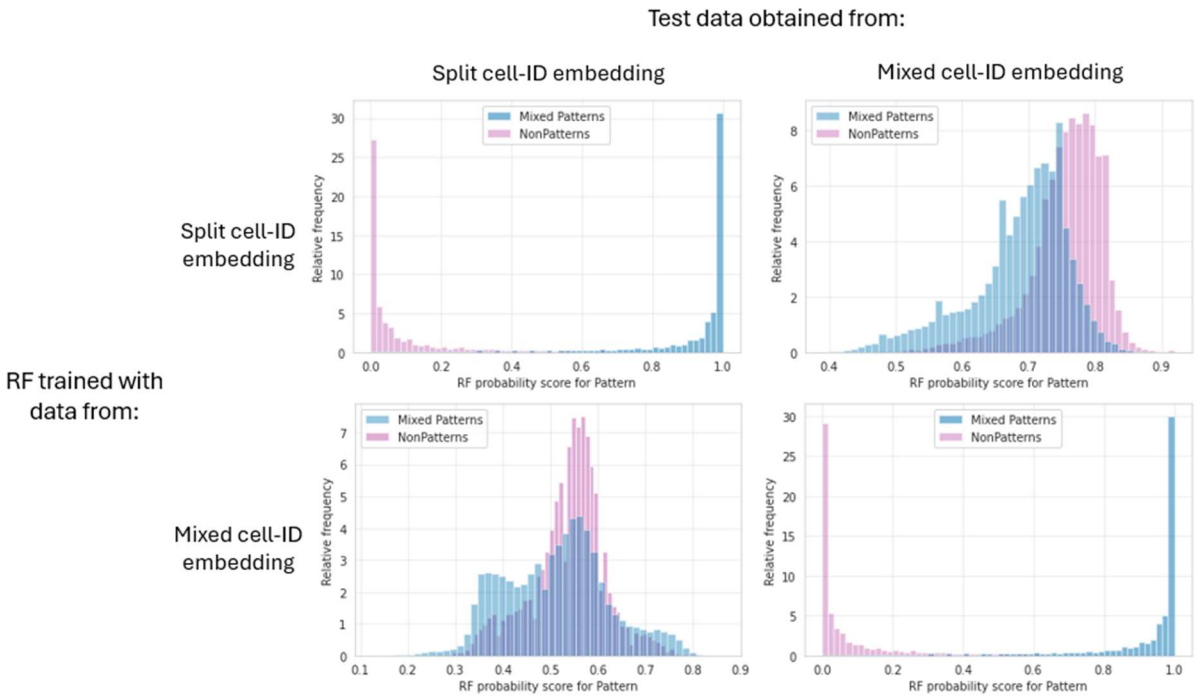

Figure 8. Random Forest prediction score distributions under different CVAE training scenarios: Histograms representing Random Forest prediction scores for different training/testing scenarios.

8.2.1.3 The optimal model's performance

The final model was trained with 3 features considered for each split and 200 trees. The receiver operating characteristic (ROC) curves, presented for subsets of the test dataset categorized by pattern strength and RNA count (Figure 9), highlight the model's efficacy. The RF's AUC scores improved incrementally with increasing RNA counts for all pattern strengths. This trend could be attributed to the model having a higher false positive rate for non-pattern observations with lower RNA counts than those with higher counts. The model performed almost perfectly for observations with a strong pattern strength and an RNA count of 30 or higher, with an AUC exceeding 0.98. Even for RNA counts below 30, the model maintained an AUC of at least 0.83. This indicates that the RF model consistently classifies observations with strong pattern strength effectively. For observations with intermediate pattern strength and RNA counts of 30 or higher, the model achieves AUC scores between 0.85 and 0.92. However, the model's performance drops for low pattern strength observations, yielding AUCs ranging from 0.52 to 0.54, which is only marginally better than random guessing. These results demonstrate that the RF model performed sufficiently well for individual observations with intermediate and strong pattern strengths.

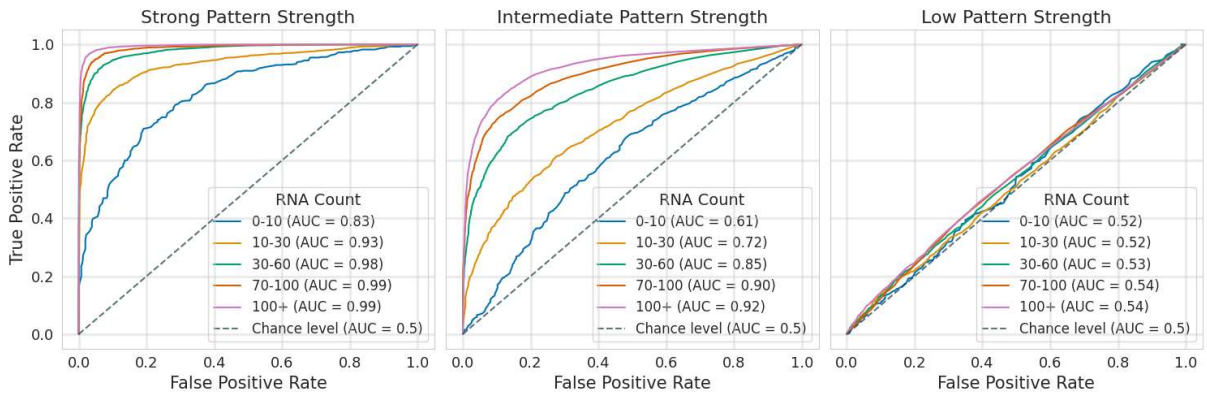

Figure 9. Classification performance to detect pattern presence: ROC curves for different RNA count values for strong (left), intermediate (middle) and low (right) pattern strengths.

8.2.1.4 Gene localization detection across cells

Aggregating the RF results over multiple cells is likely to enhance its overall performance, as theoretically, the ensemble's predictive power should improve with aggregation. Moreover, this approach would enable the determination of whether a gene consistently exhibits a pattern across various cells, rather than in isolated observations. We therefore created simulated genes (methods) and tested whether the RF score probability density curve of the test (patterned) gene significantly differed from that of a non-patterned control gene. A illustrates this comparison for a mixed-pattern gene with 600 observations with a strong pattern strength and an RNA count of 0-10. The test gene significantly differed from the non-patterned control gene ($D(600,600)=0.48$, $p = 5.92 \cdot 10^{-63}$), and would therefore be classified as a patterned gene by our model/framework.

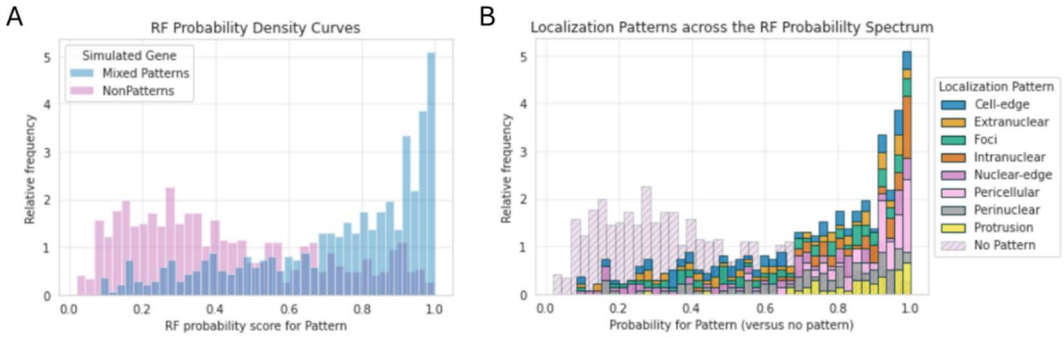

Figure 10. RF score distributions for simulated genes showing either patterns or no patterns: (A) Histograms represent RF score distributions for 600 gene simulations with RNA counts between 0 and 10 and either a mixed pattern type with strong pattern strength (blue), or no pattern (pink). (B) The same simulations colored by pattern type.

We then investigated whether our model classifies certain patterns better than others. Figure 10B shows the distribution of patterns across the probability spectrum seemed fairly even, indicating that the random forest classifier does not exhibit a significant bias towards any particular pattern. However, the model classified certain patterns, such as pericellular and intranuclear, with higher probabilities, suggesting better performance for these patterns. In conclusion, the even distribution

of different patterns in the RF probability spectrum reinforces the model's robustness and generalizability in pattern classification. Moreover, the mixed-pattern gene appears to be a good approximation of the average RF score across various patterns.

While the observations so far provide a strong proof of concept, it remains to be seen whether this statistical framework works for more biologically realistic cases, where patterns may have lower strength, or involve fewer cells. For instance, a simulated gene with intermediate pattern strength but the same RNA count and cell count also significantly differed from a matched non-pattern control ($D(600,600)=0.18$, $p = 1.36 \cdot 10^{-08}$). The method is also sensitive enough to detect differences in rare cell types with few cells available. For example, a gene with intermediate pattern strength, an RNA count of 10-30 and a cell count of 50 would be classified as patterned ($D(50,50)=0.42$, $p = 0.0002$), albeit this would not pass multiple testing corrections commonly performed in MERFISH analyses.

To evaluate the proposed statistical framework, a power analysis was conducted. The objective was to determine the minimum number of cells required to reliably reject the null hypothesis when there is a true difference in RF classification scores between a patterned gene and a non-patterned control gene. The power analysis was performed with an alpha level of 0.05 as a commonly accepted benchmark in statistical analyses (Figure 11).

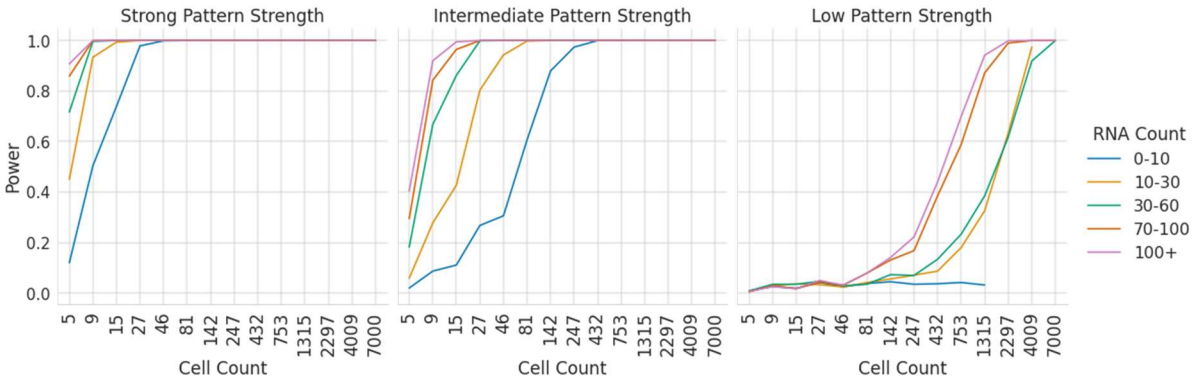

Figure 11. Power analysis for pattern presence detection across cells using the RF approach: Line plots represent the relationship between sample size and statistical power for detecting patterns across cells for strong (left), intermediate (middle) and low (right) pattern strength. Successful test statistics were considered at a false discovery rate (α) of 0.05.

For genes exhibiting strong pattern strength, the analysis demonstrated a near 100% power with as few as 30 cells. To achieve a power of 0.8, 27 cells were sufficient for genes with at least 10 detected RNA molecules. In contrast, 100 cells were necessary for genes with RNA counts below 10. Although low pattern strength only marginally outperformed random chance for individual patterns (Figure 9), the power analysis revealed that it is possible to detect low pattern strength patterns when a sufficient number of cells are aggregated. Specifically, 1315 cells were required for RNA counts above 70, and 4000 cells for RNA counts between 10 and 70. This indicates that even genes with a pattern strength that is too low to detect by the human eye can be detected through aggregations, underscoring the effectiveness of this approach in distinguishing patterned genes.

To simulate correcting for false discovery rates for 500 genes across 10 cell types, the power analysis was adjusted using a Bonferroni correction for 5000 tests. Strong and intermediate patterns remained robust despite this correction (Figure 12). Differences could be reliably detected even with low RNA counts starting from a cell count of 400. For low pattern strength, detection was generally unfeasible unless RNA counts exceeded 70. Using Cohen's (1998)[84] recommendation of a maximum Type II error probability of 20%, genes with low pattern strength but high RNA counts (70 or above) still met the power threshold, provided a sufficient number of cells were included. However, genes with such high RNA counts are atypical in the host-lab's experience.

Given the current power analysis and the size of the simulated dataset, it remains unclear whether genes with low RNA counts and low pattern strength could achieve sufficient power. As MERFISH experiments can include 100,000 cells for non-rare cell types, it could be that these sample sizes would be high enough to reach sufficient power. Further investigation is required to draw definitive conclusions for these categories.

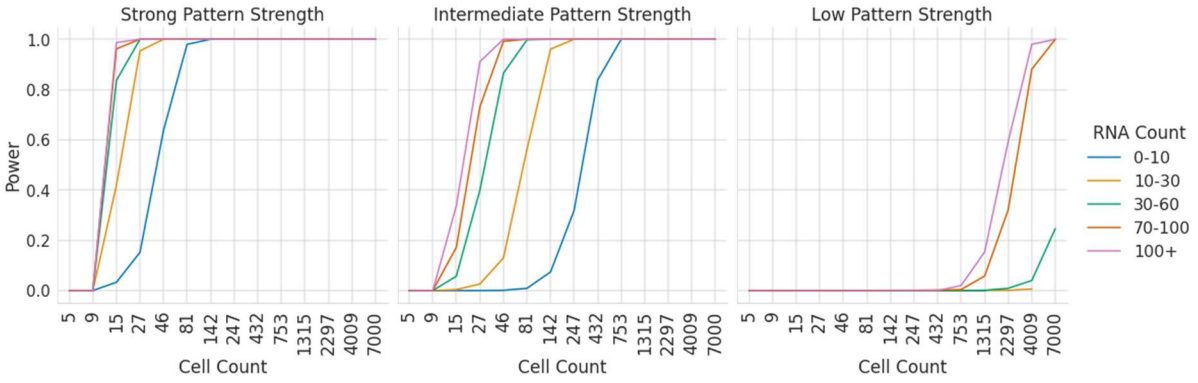

Figure 12. Power analysis for pattern presence detection across cells after Bonferroni multiple testing correction using the RF approach: Line plots represent the relationship between sample size and statistical power for detecting patterns across cells for strong (left), intermediate (middle) and low (right) pattern strengths. Successful test statistics were considered at a Bonferroni corrected false discovery rate (α) of 0.05/5000.

8.2.1.5 Assessment of the specific localization pattern

To accurately classify specific pattern categories that a gene might exhibit, we opted for a separate binary RF classifier for each pattern rather than a single multi-class classifier. This approach allows genes to portray multiple localizations within the same cell (i.e. pericellular and foci). To ensure this approach did not compromise accuracy, we conducted a sensitivity analysis comparing the AUC scores of each pattern between the multi-class classifier and the eight binary classifiers. Both classification methods achieved comparable AUC scores (see Table 2), validating our choice for the binary RF classifiers to assign probabilities for genes to portray specific subcellular localization patterns. Going forward, we limited the scope of our analysis to one specific pattern for brevity. We opted for pericellular localization, which was one of the better performing patterns (see Table 2 and Figure 10B).

Pattern	Multiclass AUC	8 Binary AUC
Cell-Edge	96.06%	96.14%
Extranuclear	93.38%	93.97%
Foci	95.34%	94.92%
Intranuclear	99.31%	99.23%
Nuclear-Edge	99.12%	99.18%
Pericellular	99.56%	99.49%
Perinuclear	97.85%	97.79%
Protrusion	99.47%	99.28%

Table 2: Comparison between the AUC scores of the multiclass RF classifier and 8 binary RF classifiers. AUCs were calculated on the strong pattern test dataset. Multiclass AUCs were calculated with a one versus rest approach.

We examined the effect of different variations of training datasets on the performance of the pericellular RF classifier. Firstly the impact of balancing the dataset was investigated. Unlike the general pattern/nonpattern classifier, this analysis revealed a difference between balanced and unbalanced datasets at the intermediate pattern strength level (Figure 13A). Based on these findings, all subsequent classifiers were trained with balanced pattern type data. Furthermore, we examined the effect of including non-patterned training data in the RF classifier. This inclusion aimed to enhance the model's performance by providing a reference for non-patterned observations, which could prove useful if a gene deemed patterned in the first round of classification (i.e. pattern vs non-pattern) has some non-patterned cells amongst its observations. The ROC curves show that both models perform similarly (Figure 13B), but the model including non-patterned data slightly outperformed the other for the intermediate pattern strength. Thus, this model was adopted for further analysis.

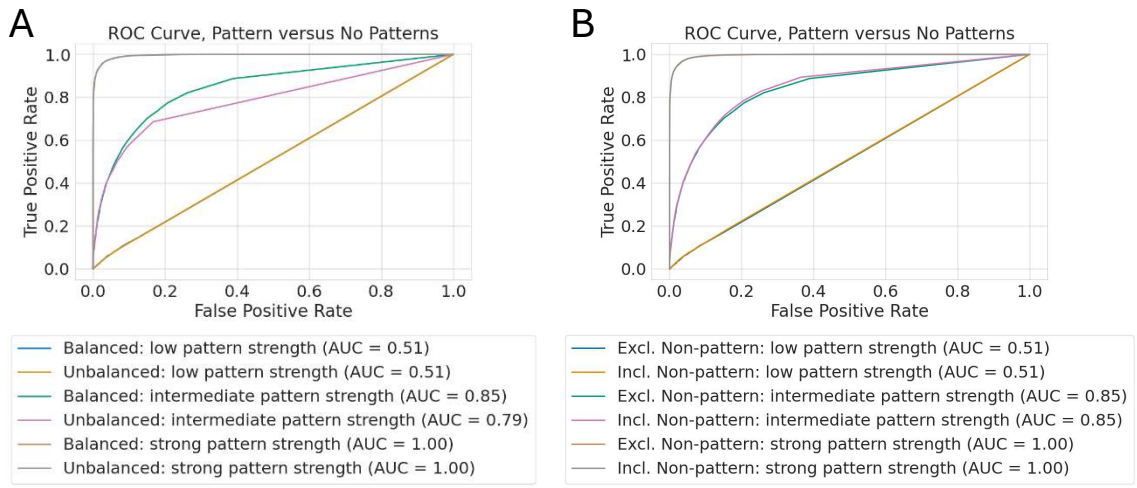

Figure 13. Classification performance for pericellular patterns: ROC curves with different degrees of pattern strength for (A) balanced and unbalanced training with respect to pattern type, pattern strength and RNA count, evaluated with a balanced test set, and (B) Including or excluding non-pattern data while training, with a balanced test set including non-patterned observations.

The performance of the optimized pericellular versus other patterns model is detailed in Figure 14. The model achieved perfect predictions for observations with strong pattern strengths and RNA counts higher than 30, with an AUC of at least 0.91 for lower RNA counts. For intermediate pattern strength, the model's performance ranged from 0.78 to 0.88, improving with higher RNA counts. For low pattern strength, the model's performance is equal to or slightly better than random chance.

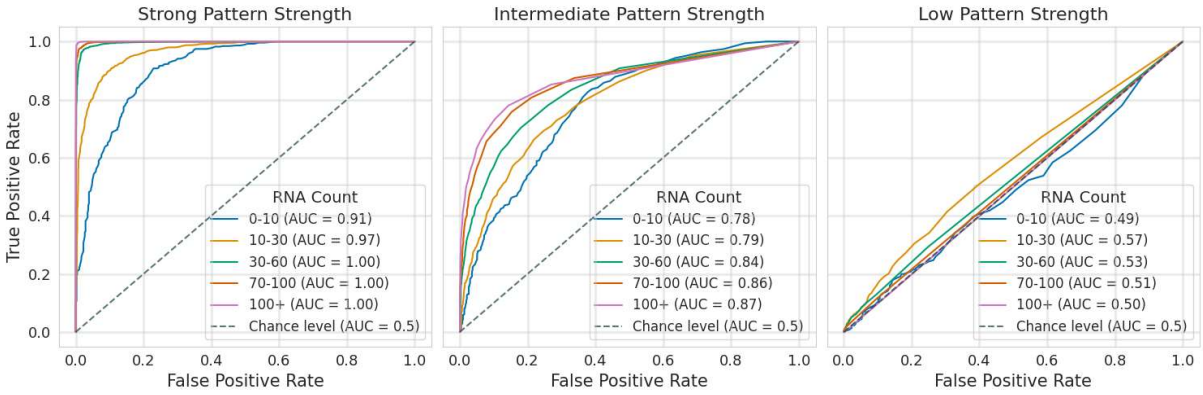

Figure 14. Classification performance to detect pericellular pattern presence: ROC curves for pericellular classification under strong (left), intermediate (middle) and low (right) pattern strengths, colored by varying levels of RNA counts.

To determine whether we could detect a gene that consistently exhibits a pericellular localization pattern across various cells, we compared three simulated genes: a pericellular gene, a non-patterned gene, and a gene composed of a mix of all non-pericellular localization patterns. Each gene included 300 cells with RNA counts between 0 and 10, and the two patterned genes had an intermediate pattern strength. This comparison was performed for models including (Figure 15A) and excluding random patterns (Figure 15B). The pericellular gene exhibited a wide probability distribution, with many observations falling below a 0.5 probability threshold, which would not classify them as

pericellular individually. However, both the simulated non-patterned gene and the gene composed of other patterns were assigned significantly lower probabilities, distinguishing the pericellular gene from other patterns (D: 0.50, p = $1.89 \cdot 10^{-34}$) and from the non-patterned gene (D: 0.56, p = $4.08 \cdot 10^{-43}$). Interestingly, even the RF classifier that had not been trained on non-patterned observations during training classified these instances as non-pericellular, supporting the robustness of the pericellular RF classifiers. Overall, we could statistically distinguish pericellular localizations, both compared to other patterns and to genes displaying no patterns. Our framework complement the statistical framework of Chouaib et al[65] where the frequency of RF classifications of specific patterns were directly compared to those of a non-patterned control.

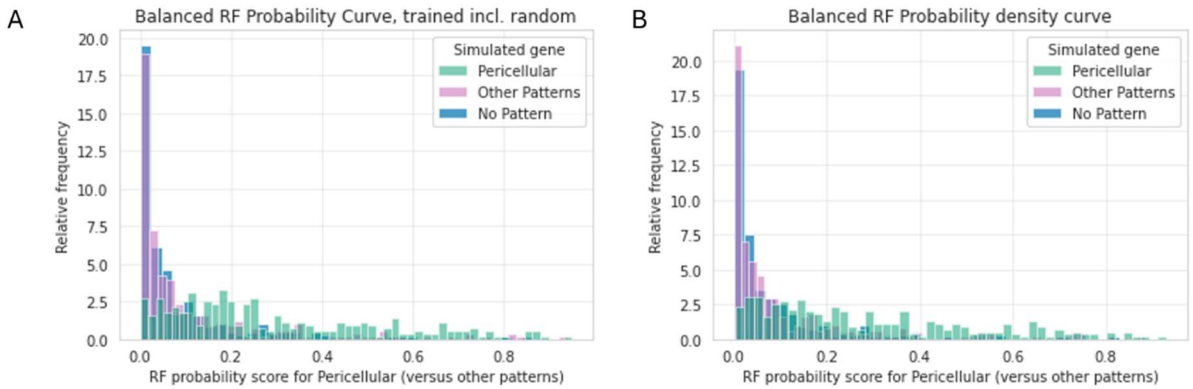

Figure 15. Random Forest score distributions of a simulated pericellular patterned gene: Histograms showing the RF scores for a simulation of 300 cells showing a gene with RNA counts between 0 and 10 exhibiting a pericellular pattern with intermediate pattern strength (turquoise), a mixture of non-pericellular patterns with intermediate pattern strength (pink) or no pattern localizations (blue) either including (A) or excluding (B) random patterns while training the pattern type classifier.

8.2.2 Using the latent space

As an alternative approach to supervised classification for determining the presence of a pattern for a particular gene, we could also directly compare point clouds in the latent space embedding. The advantage of directly comparing point clouds within the latent space is that there is no dependence on predefined patterns. This could be especially of interest when determining which pattern a gene might portray, given that one could in theory detect *de novo* patterns, which would be undetectable through supervised classification of a set of predefined patterns. We therefore compared the similarity of the point clouds of test genes with non-patterned control genes using a Chamfer distance metric (methods). A permutation test was used to test if the observed distance was significantly larger than a hypothetical null distribution (see methods). Figure 16 illustrates this comparison for a mixed pattern gene with 600 observations with an intermediate pattern strength and an RNA count of 0-10. The observed distance between the test gene and non-patterned control gene (pink dotted line) was significantly larger than expected if the two genes would have been drawn from the same underlying distribution (p = 0.0001). The test gene would therefore be classified as a patterned gene by our model/framework.

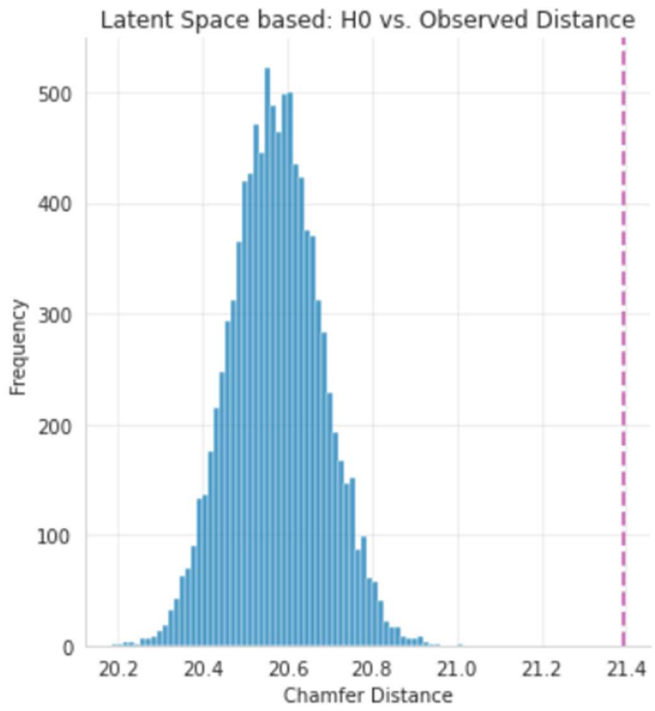

Figure 16. Empirical distribution of Chamfer distance: Histogram representing the distribution of Chamfer distance estimates for the null distribution of permutations (blue) versus the observed Chamfer distance (pink) for a patterned gene with 600 observations, intermediate pattern strength and RNA counts between 0 and 10 for each observation.

We conducted a power analysis to thoroughly assess the robustness of the proposed LS-based statistical test. To compare results with the supervised-based framework, the power analysis was performed with an alpha level of 0.05 (Figure 17). Due to time constraints, the power analysis was only computed for the sample sizes up to 1315 cells. For genes exhibiting strong pattern strength, the analysis demonstrated nearly 100% power with as few as 46 cells. For genes with intermediate pattern strength, 81 cells were sufficient to achieve a power of at least 0.8 for cells with at least 10 RNA. In contrast, 247 cells were necessary for genes with RNA counts below 10.

Generally for both the supervised classifier and this LS-based approach the power increased with increasing counts, however the opposite occurred for observations with a low pattern strength. For these observations, the power went down with increasing RNA counts. The LS-based method achieved a power of 0.99 at 432 cells, whereas higher RNA counts had an average power of 0.28. Nevertheless, all RNA counts were able to reach a power of at least 0.8. Similarly to the supervised-based model, a Bonferroni correction of 5000 was performed, however all observations remained underpowered. This was to be expected, given that 9999 permutations only a minimum p-value of 0.0001 can be achieved, whereas the Bonferroni would require $p < 0.00001$ for a successful detection. Future studies should perform a power analysis with more permutations to ascertain whether the permutation test could surpass a Bonferroni correction of 5000.

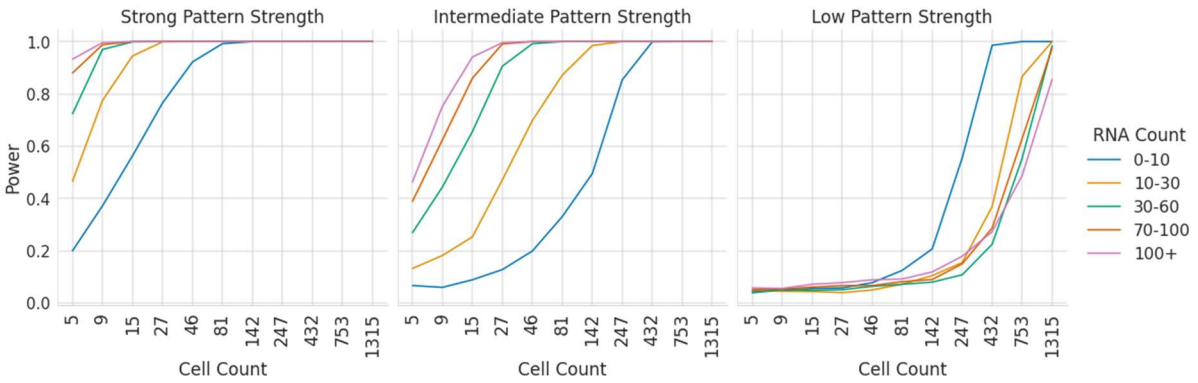

Figure 17: Power analysis for pattern presence detection using the latent space permutation test: Line plots represent the relationship between sample size and statistical power for detecting patterns across cells for strong (left), intermediate (middle) and low (right) pattern strength. Successful test statistics were considered at a false discovery rate (α) of 0.05.

8.3 Validation on a Biological Dataset

Having established that both tests work for simulated data, we aimed to determine if these results would generalize to biological and experimental contexts. To this end, we used an experimental MERFISH dataset from the Laboratory for Systems Physiology, ETH Zurich[73] **,** which focused on subcellular RNA localization patterns in the small intestine. This dataset included six genes which are generally accepted to exhibit subcellular localization in enterocytes: *ApoB*, *CDH1*, and *CDKN1A* (apically localized) and *CYB5R3*, *PIGR*, and *NET1* (basally localized). Our goal was to evaluate whether these ground-truth genes were also classified as patterned by the supervised and LS-based approaches.

For each of the six genes, we simulated a non-patterned control with the same cell count and matched RNA count. The RNA count for the simulated control was matched by taking the mean RNA per cell of the test gene, plus or minus the standard deviation, and rounding to the nearest tenfold. We then compared each gene with its matched control gene using our two statistical tests. The descriptive statistics and results can be found in Table 3. All genes showed significant differences from their matched controls. While the p-values varied in the supervised approach, the LS-based p-values were uniformly 0.0001. To further investigate whether the two approaches might classify any biological gene as patterned, we selected two genes (*CLCA4a* and *SLC39A14*) that visually did not exhibit any localization pattern. Both of these genes also showed significant differences from their non-patterned controls, with RF p-values surviving a Bonferroni correction of 5000 (*CLCA4a*: LS-based: $p = 0.0001$, RF: $p = 4.21 \cdot 10^{-27}$; *SLC39A14*: LS-based: $p = 0.0001$, RF: $p = 8.03 \cdot 10^{-26}$).

Description of the Genes					*Simulated control*	*Statistical results*		
Gene	Apical/ Basal	Cell count	Mean (std) RNA count	Min - max RNA count	RNA count used for simulated control	P-value: LS-based	RF: KS stat	P-value: RF
ApoB	Apical	406	26.5 (15.2)	[1-92]	0-100	0.0001	0.90	$2.91 \cdot 10^{-175}$
CDH1	Apical	358	3.55 (2.53)	[1 – 15]	0-10	0.0001	0.68	$1.61 \cdot 10^{-79}$
CDKN1A	Apical	376	4.2 (3.28)	[1 – 21]	0-10	0.0001	0.59	$5.74 \cdot 10^{-60}$
CYB5R3	Basal	406	18.3 (10.5)	[1 – 63]	10-30	0.0001	0.67	$3.42 \cdot 10^{-88}$
NET1	Basal	155	1.4 (0.75)	[1 – 4]	0-10	0.0001	0.80	$9.31 \cdot 10^{-50}$
PIGR	Basal	408	68.1 (41.5)	[4 - 299]	20-100	0.0001	0.91	$8.22 \cdot 10^{-182}$
CLCA4a	Seems random	125	4.86 (7.94)	[1-39]	0-20	0.0001	0.67	$4.21 \cdot 10^{-27}$
SLC39A14	Seems random	389	5.49 (3.76)	[1-23]	0-10	0.0001	0.38	$8.03 \cdot 10^{-26}$

Table 3: Description of the biological validation dataset and results of the statistics of the LS-based and RF-based approaches.

We hypothesized that extreme RNA counts might skew the p-values. Therefore, we filtered the dataset based on RNA counts that could be visually recognized as pattern presenting during the training of the CVAE model. Specifically, RNA counts below 10 were excluded (as it was difficult to say if e.g. 4 RNA spots formed a pattern or not), as well as counts above 100 (where molecular crowding made it difficult to distinguish patterns). After applying these filters, *NET1* was excluded due to insufficient cell counts. New matching non-patterned controls were created for the remaining genes (see Table 4). All the genes remained significant, although the p-values were less extreme and seemed more realistic. Notably, *CLCA4a* barely reached significance (LS-based: $p = 0.022$, RF: $p = 0.045$), and four genes (*CDH1*, *CDKN1A*, *CLCA4a*, and *SLC39A14*) would no longer be significant after a Bonferroni correction of 5000. It remains unclear whether the reduced extremity of p-values compared to the unfiltered dataset was due to lower cell counts (and thus lower test power), or if low RNA count observations might have overestimated pattern presence.

Description of the Genes					*Simulated control*	*Statistical results*		
Gene	Apical/ Basal	Cell count	Mean (std) RNA count	Min - max RNA count	RNA count used for simulated control	P-value: LS-based	RF: KS stat	P-value: RF
ApoB	Apical	353	29.5 (14.0)	[11-92]	10-50	0.0001	0.89	$2.47 \cdot 10^{-148}$
CDH1	Apical	9	12 (1.2)	[11 – 15]	10-20	0.0067	0.67	0.034
CDKN1A	Apical	22	13.5 (2.9)	[11 – 21]	10-20	0.0001	0.64	0.00017
CYB5R3	Basal	297	22.6 (8.9)	[11 – 63]	10-30	0.0001	0.63	$7.06 \cdot 10^{-55}$
NET1	Basal	313	53.7 (23.5)	[11,100]	30-70	0.0001	0.81	$9.61 \cdot 10^{-104}$
PIGR	Basal	17	22.6 (8.8)	[12-39]	10-40	0.022	0.47	0.045
CLCA4a	Seems random	40	13.7 (2.9)	[11-23]	10-20	0.0001	0.4	0.0030
SLC39A14	Seems random	353	29.5 (14.0)	[11-92]	10-50	0.0001	0.89	$2.47 \cdot 10^{-148}$

Table 4: Description of the filtered biological validation dataset and results of the statistics of the LS-based and RF-based approaches. RNA counts below 10 and above 100 were excluded from the biological validation dataset.

9 DISCUSSION

In this study, we built upon an in-house convolutional autoencoder model which automatically detects RNA localization patterns without relying on manual feature inputs. Our primary goal was to create a statistical framework capable of quantifying the probability of RNA localization for individual genes across multiple cells. This framework was tested using simulated data that closely mimics biological reality and was further validated with an experimental MERFISH dataset of enterocyte apical-basal polarization. An extensive power analysis also reveals the necessary sample sizes at varying pattern strengths and dynamic ranges. We employed two distinct approaches: supervised classification and localization detection within the latent space embedding. By aggregating model classifications across all cells for each gene, we aimed to establish a robust method for determining gene localization probabilities. Furthermore, we explored our model's ability to distinguish pericellular patterns from non-patterns and from other localization patterns. Finally, we validated that the results from simulated data could be generalized to biological and experimental contexts.

Even on individual observations, the random forest alone showed promising sensitivity with intermediate and strong pattern strengths. Given that random forests, as an ensemble method, have more power to accurately classify data than a single decision tree, we hypothesized that aggregating moderately accurate predictions across many cells for each gene would enhance sensitivity in detecting RNA localization patterns. This hypothesis was confirmed. Both the supervised classification and LS-based statistical framework successfully identified RNA localization patterns across populations of cells, achieving satisfactory results with simulated data that had realistic pattern strengths and RNA counts. Notably, we included a more diverse set of localization patterns (nine in total) compared to the current standard in the field, which typically include five (see Table 1). These findings were further validated with a ground-truth experimental dataset, demonstrating the robustness of our approach.

Our statistical framework demonstrated sensitivity to low pattern strengths, whereas random forests alone performed only marginally better than chance. Moreover, both approaches detected significant pattern presence in all genes from our experimental dataset, including those that did not visibly exhibit any localization pattern. This underscores the conceptual intrigue of observations with low pattern strength. Patterns with only 10% of their RNA localized in specific regions can appear indistinguishable from non-patterned genes to the human eye. Our framework can detect these subtle patterns, suggesting that some genes labeled as non-patterned in biological datasets might actually exhibit low-strength patterns. This capability is significant given the nascent state of the spatial transcriptomics field and the reliance on visual inspection. It remains unclear whether low-strength patterns are biologically relevant. Our framework could help further investigate the potential biological relevance of these low-strength patterns.

The supervised classification approach generally outperformed the LS-based approach, particularly when considering resource requirements. The LS-based approach is resource-intensive because it calculates the mean L1 distance from each point in point cloud 1 to its nearest neighbor in point cloud 2 and vice versa. This calculation necessitates a distance matrix with dimensions proportional to the cell count, causing the computational intensity to scale exponentially with the number of included cells. Furthermore, this process must be repeated for each of the 9,999 permutations, allowing for a minimum p-value of 0.0001. If a lower p-value is required to correct for multiple testing errors, even more resources would be needed. In contrast, the RF-based approach is less computationally demanding. The random forest obtains a posterior probability by averaging the predictions of all its decision trees, resulting in a substantially lower complexity than the LS-based approach. Therefore, even though both approaches achieved satisfactory results, the RF-based approach is preferred over the LS-based approach.

Unexpectedly, the permutation test in low pattern strengths performed better for genes with low RNA counts compared to higher counts (Figure 17). This phenomenon could potentially be an artifact of molecular crowding. Alternatively, it could be explained by the overlap of low pattern strength observations with non-patterned observations within the latent space, in relation to the RNA counts. As shown in Figure 4D, lower RNA counts occupy a distinct space in the center right, whereas higher counts (i.e. above 100) cluster closer to non-patterned observations (Figure 4A). Consequently, genes with low pattern strength but high RNA counts are nearer to non-patterned observations in the latent space than genes with low counts. This results in the observed Chamfer distance and the hypothesized null-distribution for the two genes being closer, making it less likely to reject the null hypothesis for genes with high RNA counts than those with low RNA counts which are further from non-patterned observations.

9.1 LIMITATIONS

As previously mentioned, our framework can detect pattern presence in observations with 10% pattern strength. However, as it is unknown whether these pattern strengths are biologically relevant, it is possible that detecting 10% patterns might be an artifact of training on simulated data, and therefore overclassification on real data. Indeed, the two genes from the biological validation set that visually did not exhibit any localization pattern, were detected as pattern presenting by both the RF- and LS-based approaches (see Table 3 and Table 4). This could indicate that the algorithm might be overly sensitive. Even if these low strength patterns would be biologically meaningful, the statistical framework currently does not allow users to merely detect strong patterns while excluding all genes with subtle patterns. Future studies could explore whether the KS test statistic would be reliable enough to treat as an effect size one could filter on. A KS lower than 0.5 has been acknowledged as not distinct enough as a general rule of thumb. In our biological validation test set, at least one of the

two visually non-patterned genes had a KS value lower than 0.5 in both the unfiltered and filtered genes (see table Table 3 and Table 4

Description of the Genes					*Simulated control*	*Statistical results*		
Gene	Apical/ Basal	Cell count	Mean (std) RNA count	Min - max RNA count	RNA count used for simulated control	P-value: LS-based	RF: KS stat	P-value: RF
ApoB	Apical	353	29.5 (14.0)	[11-92]	10-50	0.0001	0.89	2.47•10-148
CDH1	Apical	9	12 (1.2)	[11 – 15]	10-20	0.0067	0.67	0.034
CDKN1A	Apical	22	13.5 (2.9)	[11 – 21]	10-20	0.0001	0.64	0.00017
CYB5R3	Basal	297	22.6 (8.9)	[11 – 63]	10-30	0.0001	0.63	7.06•10-55
NET1	Basal	313	53.7 (23.5)	[11,100]	30-70	0.0001	0.81	9.61•10-104
PIGR	Basal	17	22.6 (8.8)	[12-39]	10-40	0.022	0.47	0.045
CLCA4a	Seems random	40	13.7 (2.9)	[11-23]	10-20	0.0001	0.4	0.0030
SLC39A14	Seems random	353	29.5 (14.0)	[11-92]	10-50	0.0001	0.89	2.47•10-148

Table 4), suggesting this could be interesting to further explore.

A ground-negative random distribution for subcellular RNA localization in a biological setting has not yet been described in literature. Although one study labeled 375 genes as non-patterned[65], this was conducted in HeLa cells, raising questions about its generalizability to other cell lines or *in vivo* tissue samples. Additionally, given the classification was based on visual inspection, some of these non-patterned genes might actually exhibit 10% pattern presence, which may or may not be biologically relevant. The assumption that simulated non-patterned localizations reflect biological reality poses a significant limitation. Both the CVAE model architecture and the proposed statistical framework were trained on this simulated dataset. If our definition of a non-patterned localization is not biologically accurate, comparing a test gene to this simulated non-patterned control would not be relevant. To overcome this limitation, our research group plans to conduct a MERFISH experiment, visualizing genes APEX-seq has flagged as non-patterned in their transcriptome-wide subcellular RNA atlas[10], to hopefully identify ground-truth non-patterned genes. If experimentally validated randomly distributed genes are found, future MERFISH experiments could include these genes as a ground-negative control. The statistical framework can then be adjusted using these new ground-negative control genes as a new null distribution.

Another limitation of our simulated dataset is the uncertainty regarding the actual strength of biological patterns compared to the simulated ones. SimFISH v1[64] (using MATLAB) addressed this by defining low, moderate, and strong pattern strengths, with moderate corresponding to biological data. However, when the software was translated to Python with SimFISH v2[62] (which we used for our simulations), different parameters were used to express pattern strength, specifically the percentage of RNAs showing patterned localization. The authors of SimFISH v2 did not compare these new parameters to real biological data for context. Similar to SimFISH v1, SimFISH v2 also displayed three different pattern strengths in their tutorial. We assumed these correspond to the original three levels, making the 50% (moderate) pattern strength a reasonable approximation of biological data. Future studies should validate this assumption to ensure accuracy.

The simulated dataset had a limited size, particularly for the non-patterned observations, which restricted the sample sizes available for the power analysis. Since we needed to match the cell count between the test and control genes[76], our sample size was constrained by the non-patterned observations (see Table 5). Consequently, we could only conduct the power analysis up to a cell count of 7000 per gene for genes with RNA counts above 30. We could analyze up to 1315 and 5000 cells for genes with RNA counts between 0-10 and 10-30 respectively. Sampling beyond these limits would require using nearly identical non-patterned genes or sampling with replacement, which

RNA count	Patterns per pattern strength			Non-patterned
	Low	Moderate	Strong	
0-10	3900	4206	3802	1442
10-30	13264	12987	13590	5305
30-60	19896	20502	19812	7836
70-100	20247	19760	19450	7402
100+	39734	39604	40397	16213

Table 5: Overview of the size of the test dataset, grouped per RNA counts, and patterned versus non-patterned data. The patterned observations were grouped based on pattern strength.

would have compromised the reliability and validity of the power analysis. In a MERFISH experiment hundreds of thousands of cells can be assayed, so our analysis does not reflect the potential for achieving sufficient power for genes with low RNA counts and low pattern strengths (Figure 12). Future studies could ensure a more comprehensive power analysis by using the current simulated data points to estimate a probability density curve, therefore allowing infinite sampling from the continuous curve. However, this approach has downsides: the probability density curve is an approximation of the data, introducing an error term, and it assumes that the distributions will remain consistent with higher samples. This consistency may not be the case as effect sizes tend to increase with larger samples, potentially introducing bias.

Matching the control to the experimental test gene proved to be critical for accurate statistics, particularly for smaller cell counts. Figure 17 illustrates this with the filtered *SLC39A14* gene (n = 40) from the ground-truth dataset, compared to a non-patterned control. The only difference between the two non-patterned genes in these two figures is the random seed used to sample the data. Therefore, *SLC39A14* did not significantly differ from the control in figure 17A (Chamfer: p = 0.5, RF: p = 0.54), while the difference was significant in figure 17B (Chamfer: p = 0.0001, RF: p = 0.003).

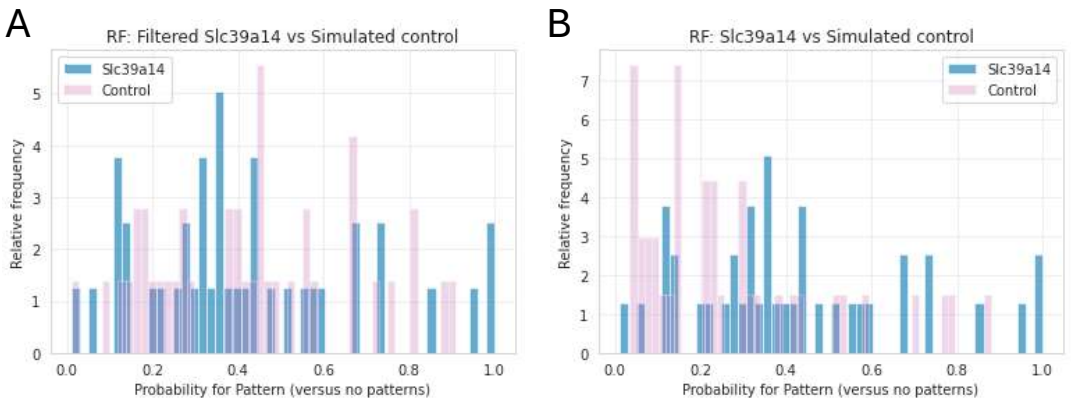

Figure 18. The influence of random seed choice on the RF score distributions: Histograms showing the RF scores for the filtered Slc39a14 gene with a simulated non-patterned control matched on RNA counts. (A) used random seed = 101 whereas (b) used random seed = 42.

Although the choice of random seed is predetermined and difficult to control, the matching of the non-patterned control can be improved by more precisely aligning the RNA counts of the experimental test gene with the control gene. Future studies should aim to exactly match the RNA counts of the control gene with those of the test gene, using the list of RNA counts from the test gene rather than a range based on the mean ± standard deviation. This is particularly important for the LS-based approach, as many distance measures, including the Chamfer distance, are sensitive to outliers within point clouds due to their reliance on the nearest neighbor in the other point cloud. The RNA count was clearly a feature learned by the CVAE and a driving force in creating the latent space. Consequently, observations with different RNA counts localized in different parts of the latent space (Figure 4D), meaning that slight variations in RNA counts could inadvertently affect the proximity of points in the latent space and thus influence the Chamfer distance. By ensuring identical RNA counts between the test and control genes, we can eliminate a potential source of variation and error. Thus, future studies should match the RNA spot counts between the test and control genes to ensure accurate results.

9.2 Future Directions

Building on the current findings, several avenues for further research and development can enhance the robustness and applicability of our framework. For example, the second classification round to identify the specific subcellular localization pattern can be further expanded. For the supervised approach, random forests can be trained for all non-pericellular patterns. For the LS-based approach, the identification of specific subcellular localization patterns needs to be implemented from the start. Currently, the plan would be to compare a gene point cloud with each pattern in the latent space and determine which one it does not significantly differ from. However, this approach is counterintuitive to statistical testing principles, as failing to reject the null hypothesis does not provide evidence that it is true.

Another area to explore is how the framework would classify a localization pattern that it has not encountered before. The most straightforward method is to assess the performance during validation when the CVAE embedding and the random forest have not been trained on a specific localization pattern. However, this requires creating many different models, which does not truly test whether the framework would perform well with *de novo* patterns. Additionally, if there are multiple distinct *de novo* localization patterns, it is important to investigate whether the CVAE would assign them similar embedding features or place them in different regions of the latent space.

The current framework has primarily been tested with simulated genes that either exhibit one specific localization pattern or a mix of all non-random localization patterns. Previous studies have shown that the same gene can portray two or three different localization patterns[65]. Therefore, it would be valuable to explore the heterogeneity of patterns within a gene across cells in future studies. From a biological perspective, it would be intriguing to see if a certain gene exhibits high heterogeneity in cell type A but low heterogeneity in cell type B. Therefore, future studies should create simulated genes that exhibit a specific localization pattern in e.g. 60% of cells and a non-patterned localization in 40%, so we can test whether the statistical framework would still detect this heterogeneous gene as pattern-presenting. Similarly, for a gene with 60% of cells showing localization pattern A and 40% showing localization pattern B, it would be interesting to see if these patterns are identified in the second classification round. Implementing the Gini impurity index, as done by Samacoits et al.[64], could be a relatively easy way to test for heterogeneity between cells in a gene.

The CVAE embedding currently struggles with foci patterns, as the algorithm blurs the MERFISH input image, making it difficult to detect foci blobs. Following the approach of Chouaib et al.[65] and FISH-quant v2[62], using DBSCAN[85] to automatically detect and/or count the number of foci in a cell could be beneficial. This metadata could then be fed into the CVAE.

For the permutation test, expanding upon the Chamfer distance is a promising direction. Recent developments have introduced modifications to the Chamfer distance that account for the density of the point clouds[80], considering potential differences between their density distributions.

However, this method was implemented using Euclidean distance, which is effective in 3D spaces but falls short in higher dimensions due to the curse of dimensionality. Given that the L1 (Manhattan distance) version of the Chamfer distance is widely used in training point cloud generators[80] , we decided to implement this as a proof of concept for the LS-based approach. Future studies could adapt the density-aware Chamfer Distance to use the L1 distance, enhancing its applicability in higher-dimensional spaces.

In this master thesis study, we developed and validated a statistical framework to quantify subcellular RNA localization probabilities for individual genes across multiple cells. Using simulated and experimental MERFISH datasets, we employed supervised classification and LS-based approach to aggregate model classifications and determine gene localization probabilities. Our findings showed that the aggregating across cells significantly enhanced sensitivity in detecting RNA localization patterns, including subtle low-strength patterns that are visually indistinguishable for individual observations. The supervised approach generally outperformed the LS-based approach, especially when considering Bonferroni corrections. Future research should validate our assumptions and refine the framework to ensure its accuracy and applicability in biological contexts. We are confident that the proposed statistical framework will help shift the focus in the field of subcellular RNA localization from global trends to gene localization across cells. This contribution will enable deeper insights into the molecular biology of subcellular RNA localization, thereby advancing the broader field of spatial transcriptomics.

10 CODE AVAILABILITY

All code used is available at: https://github.com/nynkekatinka/subcellular_RNA_localization

11 REFERENCES

1. Marx, V. Method of the Year: spatially resolved transcriptomics. *Nat. Methods* **18**, 9–14 (2021).
2. Das, S., Vera, M., Gandin, V., Singer, R. H. & Tutucci, E. Intracellular mRNA transport and localized translation. *Nat. Rev. Mol. Cell Biol.* **22**, 483–504 (2021).
3. Engel, K. L., Arora, A., Goering, R., Lo, H. G. & Taliaferro, J. M. Mechanisms and consequences of subcellular RNA localization across diverse cell types. *Traffic* **21**, 404–418 (2020).
4. Jeffery, W. R., Tomlinson, C. R. & Brodeur, R. D. Localization of actin messenger RNA during early ascidian development. *Dev. Biol.* **99**, 408–417 (1983).
5. Cabili, M. N. *et al.* Localization and abundance analysis of human lncRNAs at single-cell and single-molecule resolution. *Genome Biol.* **16**, 20 (2015).
6. Clark, M. B. *et al.* Genome-wide analysis of long noncoding RNA stability. *Genome Res.* **22**, 885–898 (2012).
7. Mili, S., Moissoglu, K. & Macara, I. G. Genome-wide screen reveals APC-associated RNAs enriched in cell protrusions. *Nature* **453**, 115–119 (2008).
8. Buxbaum, A. R., Wu, B. & Singer, R. H. Single β-actin mRNA detection in neurons reveals a mechanism for regulating its translatability. *Science* **343**, 419–422 (2014).
9. Cornelison, G. L., Levy, S. A., Jenson, T. & Frost, B. Tau-induced nuclear envelope invagination causes a toxic accumulation of mRNA in Drosophila. *Aging Cell* **18**, e12847 (2019).
10. Fazal, F. M. *et al.* Atlas of Subcellular RNA Localization Revealed by APEX-Seq. *Cell* **178**, 473-490.e26 (2019).
11. Ashley, J. *et al.* Retrovirus-like Gag Protein Arc1 Binds RNA and Traffics across Synaptic Boutons. *Cell* **172**, 262-274.e11 (2018).

12. Alhowikan, A. M. Activity-Regulated Cytoskeleton-Associated Protein Dysfunction May Contribute to Memory Disorder and Earlier Detection of Autism Spectrum Disorders. *Med. Princ. Pract. Int. J. Kuwait Univ. Health Sci. Cent.* **25**, 350–354 (2016).

13. Fromer, M. *et al.* De novo mutations in schizophrenia implicate synaptic networks. *Nature* **506**, 179–184 (2014).

14. Wang, J., Horlacher, M., Cheng, L. & Winther, O. RNA trafficking and subcellular localization-a review of mechanisms, experimental and predictive methodologies. *Brief. Bioinform.* **24**, bbad249 (2023).

15. Ioannou, M. S. & McPherson, P. S. Regulation of Cancer Cell Behavior by the Small GTPase Rab13. *J. Biol. Chem.* **291**, 9929–9937 (2016).

16. Nousiainen, H. O. *et al.* Mutations in mRNA export mediator GLE1 result in a fetal motoneuron disease. *Nat. Genet.* **40**, 155–157 (2008).

17. Lessel, D. *et al.* De Novo Missense Mutations in DHX30 Impair Global Translation and Cause a Neurodevelopmental Disorder. *Am. J. Hum. Genet.* **101**, 716–724 (2017).

18. Patel, A. *et al.* A Liquid-to-Solid Phase Transition of the ALS Protein FUS Accelerated by Disease Mutation. *Cell* **162**, 1066–1077 (2015).

19. Lester, E. *et al.* Tau aggregates are RNA-protein assemblies that mislocalize multiple nuclear speckle components. *Neuron* **109**, 1675-1691.e9 (2021).

20. Johnstone, O. & Lasko, P. Translational regulation and RNA localization in Drosophila oocytes and embryos. *Annu. Rev. Genet.* **35**, 365–406 (2001).

21. Wang, C., Han, B., Zhou, R. & Zhuang, X. Real-Time Imaging of Translation on Single mRNA Transcripts in Live Cells. *Cell* **165**, 990–1001 (2016).

22. Batada, N. N., Shepp, L. A. & Siegmund, D. O. Stochastic model of protein–protein interaction: Why signaling proteins need to be colocalized. *Proc. Natl. Acad. Sci.* **101**, 6445–6449 (2004).

23. Fernandopulle, M. S., Lippincott-Schwartz, J. & Ward, M. E. RNA transport and local translation in neurodevelopmental and neurodegenerative disease. *Nat. Neurosci.* **24**, 622–632 (2021).

24. Formicola, N., Vijayakumar, J. & Besse, F. Neuronal ribonucleoprotein granules: Dynamic sensors of localized signals. *Traffic Cph. Den.* **20**, 639–649 (2019).

25. Gasset-Rosa, F. *et al.* Cytoplasmic TDP-43 De-mixing Independent of Stress Granules Drives Inhibition of Nuclear Import, Loss of Nuclear TDP-43, and Cell Death. *Neuron* **102**, 339-357.e7 (2019).

26. Yasuda, K., Clatterbuck-Soper, S. F., Jackrel, M. E., Shorter, J. & Mili, S. FUS inclusions disrupt RNA localization by sequestering kinesin-1 and inhibiting microtubule detyrosination. *J. Cell Biol.* **216**, 1015–1034 (2017).

27. Jenny, A. *et al.* A translation-independent role of *oskar* RNA in early *Drosophila* oogenesis. *Development* **133**, 2827–2833 (2006).

28. Crerar, H. *et al.* Regulation of NGF Signaling by an Axonal Untranslated mRNA. *Neuron* **102**, 553-563.e8 (2019).

29. Moissoglu, K. *et al.* RNA localization and co-translational interactions control RAB13 GTPase function and cell migration. *EMBO J.* **39**, e104958 (2020).

30. Ioannou, M. S. *et al.* DENND2B activates Rab13 at the leading edge of migrating cells and promotes metastatic behavior. *J. Cell Biol.* **208**, 629–648 (2015).

31. Buxbaum, A. R., Haimovich, G. & Singer, R. H. In the right place at the right time: visualizing and understanding mRNA localization. *Nat. Rev. Mol. Cell Biol.* **16**, 95–109 (2015).

32. Gao, Y., Tatavarty, V., Korza, G., Levin, M. K. & Carson, J. H. Multiplexed Dendritic Targeting of α Calcium Calmodulin-dependent Protein Kinase II, Neurogranin, and Activity-regulated Cytoskeleton-associated Protein RNAs by the A2 Pathway. *Mol. Biol. Cell* **19**, 2311–2327 (2008).

33. Liao, Y.-C. *et al.* RNA Granules Hitchhike on Lysosomes for Long-Distance Transport, Using Annexin A11 as a Molecular Tether. *Cell* **179**, 147-164.e20 (2019).

34. Baumann, S., König, J., Koepke, J. & Feldbrügge, M. Endosomal transport of septin mRNA and protein indicates local translation on endosomes and is required for correct septin filamentation. *EMBO Rep.* **15**, 94–102 (2014).

35. Cohen, B. *et al.* Co-transport of the nuclear-encoded *Cox7c* mRNA with mitochondria along axons occurs through a coding-region-dependent mechanism. *J. Cell Sci.* **135**, jcs259436 (2022).

36. Krichevsky, A. M. & Kosik, K. S. Neuronal RNA granules: a link between RNA localization and stimulation-dependent translation. *Neuron* **32**, 683–696 (2001).

37. Adivarahan, S. *et al.* Spatial Organization of Single mRNPs at Different Stages of the Gene Expression Pathway. *Mol. Cell* **72**, 727-738.e5 (2018).

38. Liu, B. & Qian, S.-B. Translational reprogramming in cellular stress response. *Wiley Interdiscip. Rev. RNA* **5**, 301–315 (2014).

39. Tauber, D., Tauber, G. & Parker, R. Mechanisms and Regulation of RNA Condensation in RNP Granule Formation. *Trends Biochem. Sci.* **45**, 764–778 (2020).

40. Roden, C. & Gladfelter, A. S. RNA contributions to the form and function of biomolecular condensates. *Nat. Rev. Mol. Cell Biol.* **22**, 183–195 (2021).

41. Wilbertz, J. H. *et al.* Single-Molecule Imaging of mRNA Localization and Regulation during the Integrated Stress Response. *Mol. Cell* **73**, 946-958.e7 (2019).

42. Bashirullah, A. *et al.* Joint action of two RNA degradation pathways controls the timing of maternal transcript elimination at the midblastula transition in Drosophila melanogaster. *EMBO J.* **18**, 2610–2620 (1999).

43. Semotok, J. L. *et al.* Drosophila maternal Hsp83 mRNA destabilization is directed by multiple SMAUG recognition elements in the open reading frame. *Mol. Cell. Biol.* **28**, 6757–6772 (2008).

44. Semotok, J. L. *et al.* Smaug recruits the CCR4/POP2/NOT deadenylase complex to trigger maternal transcript localization in the early Drosophila embryo. *Curr. Biol. CB* **15**, 284–294 (2005).

45. Zaessinger, S., Busseau, I. & Simonelig, M. Oskar allows nanos mRNA translation in Drosophila embryos by preventing its deadenylation by Smaug/CCR4. *Dev. Camb. Engl.* **133**, 4573–4583 (2006).

46. Long, R. M. *et al.* Mating Type Switching in Yeast Controlled by Asymmetric Localization of *ASH1* mRNA. *Science* **277**, 383–387 (1997).

47. Gu, W., Deng, Y., Zenklusen, D. & Singer, R. H. A new yeast PUF family protein, Puf6p, represses ASH1 mRNA translation and is required for its localization. *Genes Dev.* **18**, 1452–1465 (2004).

48. Irie, K. The Khd1 protein, which has three KH RNA-binding motifs, is required for proper localization of ASH1 mRNA in yeast. *EMBO J.* **21**, 1158–1167 (2002).

49. Femino, A. M., Fay, F. S., Fogarty, K. & Singer, R. H. Visualization of single RNA transcripts in situ. *Science* **280**, 585–590 (1998).

50. Xia, C., Fan, J., Emanuel, G., Hao, J. & Zhuang, X. Spatial transcriptome profiling by MERFISH reveals subcellular RNA compartmentalization and cell cycle-dependent gene expression. *Proc. Natl. Acad. Sci. U. S. A.* **116**, 19490–19499 (2019).

51. Chen, K. H., Boettiger, A. N., Moffitt, J. R., Wang, S. & Zhuang, X. RNA imaging. Spatially resolved, highly multiplexed RNA profiling in single cells. *Science* **348**, aaa6090 (2015).

52. Piovesan, A. *et al.* Human protein-coding genes and gene feature statistics in 2019. *BMC Res. Notes* **12**, 315 (2019).

53. Wang, J., Horlacher, M., Cheng, L. & Winther, O. DeepLocRNA: an interpretable deep learning model for predicting RNA subcellular localization with domain-specific transfer-learning. *Bioinformatics* **40**, btae065 (2024).

54. Yan, Z., Lécuyer, E. & Blanchette, M. Prediction of mRNA subcellular localization using deep recurrent neural networks. *Bioinforma. Oxf. Engl.* **35**, i333–i342 (2019).

55. Zhang, Z.-Y. *et al.* Design powerful predictor for mRNA subcellular location prediction in Homo sapiens. *Brief. Bioinform.* **22**, 526–535 (2021).

56. Garg, A., Singhal, N., Kumar, R. & Kumar, M. mRNALoc: a novel machine-learning based in-silico tool to predict mRNA subcellular localization. *Nucleic Acids Res.* **48**, W239–W243 (2020).

57. Li, J., Zhang, L., He, S., Guo, F. & Zou, Q. SubLocEP: a novel ensemble predictor of subcellular localization of eukaryotic mRNA based on machine learning. *Brief. Bioinform.* **22**, bbaa401 (2021).

58. Wang, D. *et al.* DM3Loc: multi-label mRNA subcellular localization prediction and analysis based on multi-head self-attention mechanism. *Nucleic Acids Res.* **49**, e46 (2021).

59. Bi, Y. *et al.* Clarion is a multi-label problem transformation method for identifying mRNA subcellular localizations. *Brief. Bioinform.* **23**, bbac467 (2022).

60. Chen, Y. *et al.* mRNA-CLA: An interpretable deep learning approach for predicting mRNA subcellular localization. *Methods* **227**, 17–26 (2024).

61. Li, F. *et al.* Advancing mRNA subcellular localization prediction with graph neural network and RNA structure. Preprint at https://doi.org/10.1101/2023.12.14.571762 (2023).

62. Imbert, A. *et al.* FISH-quant v2: a scalable and modular tool for smFISH image analysis. *RNA N. Y. N* **28**, 786–795 (2022).

63. Mah, C. K. *et al.* Bento: a toolkit for subcellular analysis of spatial transcriptomics data. *Genome Biol.* **25**, 82 (2024).

64. Samacoits, A. *et al.* A computational framework to study sub-cellular RNA localization. *Nat. Commun.* **9**, 4584 (2018).

65. Chouaib, R. *et al.* A Dual Protein-mRNA Localization Screen Reveals Compartmentalized Translation and Widespread Co-translational RNA Targeting. *Dev. Cell* **54**, 773-791.e5 (2020).

66. Imbert, A., Mueller, F. & Walter, T. PointFISH -- learning point cloud representations for RNA localization patterns. Preprint at https://doi.org/10.48550/ARXIV.2302.10923 (2023).

67. Savulescu, A. F. *et al.* Interrogating RNA and protein spatial subcellular distribution in smFISH data with DypFISH. *Cell Rep. Methods* **1**, 100068 (2021).

68. Dubois, R. *et al.* A Deep Learning Approach To Identify MRNA Localization Patterns. in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)* 1386–1390 (IEEE, Venice, Italy, 2019). doi:10.1109/ISBI.2019.8759235.

69. Iandola, F. N. *et al.* SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. Preprint at https://doi.org/10.48550/ARXIV.1602.07360 (2016).

70. Walter, F. C., Stegle, O. & Velten, B. FISHFactor: a probabilistic factor model for spatial transcriptomics data with subcellular resolution. *Bioinforma. Oxf. Engl.* **39**, btad183 (2023).

71. Stoyan, D., Chiu, S. N., Kendall, W. S. & Mecke, J. *Stochastic Geometry and Its Applications*. (John Wiley & Sons Inc, Chichester, West Sussex, United Kingdom, 2013).

72. Illian, J., Penttinen, A., Stoyan, H. & Stoyan, D. *Statistical Analysis and Modelling of Spatial Point Patterns*. (Wiley, 2007). doi:10.1002/9780470725160.

73. Laboratory for Systems Physiology | ETH Zurich. https://bsse.ethz.ch/lsp (2024).

74. Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. Preprint at https://doi.org/10.48550/ARXIV.1412.6980 (2014).

75. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. (2012) doi:10.48550/ARXIV.1201.0490.

76. Christensen, W. F. & Zabriskie, B. N. When Your Permutation Test is Doomed to Fail. *Am. Stat.* **76**, 53–63 (2022).

77. Virtanen, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020).

78. Phipson, B. & Smyth, G. K. Permutation P-values should never be zero: calculating exact P-values when permutations are randomly drawn. *Stat. Appl. Genet. Mol. Biol.* **9**, Article39 (2010).

79. Ernst, M. D. Permutation Methods: A Basis for Exact Inference. *Stat. Sci.* **19**, (2004).

80. Wu, T. *et al.* Density-aware Chamfer Distance as a Comprehensive Metric for Point Cloud Completion. Preprint at https://doi.org/10.48550/ARXIV.2111.12702 (2021).

81. Liu, M., Sheng, L., Yang, S., Shao, J. & Hu, S.-M. Morphing and Sampling Network for Dense Point Cloud Completion. *Proc. AAAI Conf. Artif. Intell.* **34**, 11596–11603 (2020).

82. Baumgartner, D. & Kolassa, J. Power considerations for Kolmogorov–Smirnov and Anderson–Darling two-sample tests. *Commun. Stat. - Simul. Comput.* **52**, 3137–3145 (2023).

83. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. Preprint at https://doi.org/10.48550/ARXIV.1802.03426 (2018).

84. *Statistical Power Analysis for the Behavioral Sciences*. (L. Erlbaum Associates, Hillsdale, N.J, 1988).

85. Sander, J., Ester, M., Kriegel, H.-P. & Xu, X. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Min. Knowl. Discov.* **2**, 169–194 (1998).

12 APPENDICES

12.1 USE OF GENERATIVE AI ASSISTANCE

Use of Generative Artificial Intelligence (GenAI) – Form to be completed

Student name: Nynke Tilkema
Student number: r0927201

Please indicate with "X" whether it relates to a course assignment, to the BIG-project or to the master's thesis:

X This form is related to my master's thesis.
Title master's thesis: Statistical Inference of AI-identified Subcellular RNA Localizations
Promoter: Alejandro Sifrim

O This form is related to a BIG-project.
Title BIG-project: …
Promoter: …

O This form is related to a course assignment.
Course name: …
Course code: …

Please indicate with "X":

- o I did not use GenAI tools.
- X I did use GenAI tools. In this case specify which one (e.g. ChatGPT/GPT4/...): GPT4 and Copilot

Please indicate with "X" (possibly multiple times) in which way you were using it:

- o **X As a language assistant for reviewing or improving texts you wrote yourself, provided that the model does not add new content.** In this case, the use of GenAI is similar to the spelling and grammar check tools we already have today, so you do not need to explicitly mention using GenAI for this).
- o **As a search engine to get initial information on a topic or to make an initial search for existing research on the topic.** (This way of gathering information is similar to using an ordinary search engine when working on anassignment. As a student, you are responsible for checking and verifying the absence and correctness of references. Therefore, after this initial search, look for scientific sources and conduct your own analysis of the source documents. Interpret, analyse and process the information you obtained; don't just copy-paste it. If you then write your own text based on this information, you do not have to mention you used GenAI.)
- o **To generate text blocks.** (If you do copy-paste text blocks of GenAI output, you have to cite your GenAI sources and quote them, i.e. you clearly state that the item was created via GenAI by citation/reference.)
- o **To generate graphs or figures.** (If you do copy-paste graphs/figures of GenAI output, you have to cite the GenAI sources and quote them, i.e. you clearly state that the item was created via GenAI by citation/reference.)

- **To generate some code as part of a larger assignment.** (Watch out, this can only be done if the teacher/promotor explicitly allows it.)
- **X Other** (Contact the teacher of the course or the supervisor of the thesis or BIG project. Explain how you comply with article 84 of the examination regulations. Explain the usefulness or added value of using GenAI.): Used to debug my already written code.

Further important guidelines and remarks:

The faculty follows the KU Leuven policy regarding responsible use of GenAI. This form is an aid towards transparency about the use of GenAI by the student which is essential. Irresponsible and non-transparent use of GenAI can be considered an irregularity and can be sanctioned. Students who consider to use GenAI should inform themselves through the university website concerning the additional guidelines (How to correctly quote and refer to GenAI? What is (not) allowed? Tips and points of attention for responsible use):
https://www.kuleuven.be/english/education/student/educational-tools/generative-artificial-intelligence