Course Project | CS690

Detection of Anomalous Tissue Domains From Spatial Transcriptomic Data

Manavjeet Singh- 220616[a], Sachidanand- 220929[b] and Sikha Vamsi- 221053[c]

[a,b,c] *Department of Biological Sciences and Bio-Engineering*

Mentor: Dr. Hamim Zafar, Department of CSE and the Department of BSBE, IIT Kanpur

Contents

1. Problem Statement

Detecting and characterizing biologically heterogeneous anomalous tissue domains (ATDs) from tissue samples is of paramount importance in clinical diagnostics, targeted therapies and biomedical research. This procedure, which we refer to as Detection and Dissection of Anomalous Tissue Domains (DDATD), serves as the first and foremost step in a comprehensive analysis of tissues from affected individuals for revealing population-level and individual-specific factors (e.g., pathogenic cell types) associated with disease development. Spatial Transcriptomics (ST) provides an unprecedented opportunity to enhance DDATD by integrating spatial gene expression information across tissues. Unlike the spatial clustering task frequently encountered in ST, which focuses on clustering spatial spots into spatial domains, DDATD necessitates further isolation of anomalous clusters from normal ones.

2. Existing Methods

Existing methods developed for spatial clustering fail to identify anomalous clusters. *SpatialID* and *STANDS* are two methods capable of distinguishing anomalous spatial spots from normal ones.

2.1. STANDS [1]

STANDS is an advanced framework leveraging a suite of specialized *Generative Adversarial Networks (GANs)* to seamlessly integrate the three tasks of DDATD. The framework is composed of three components-

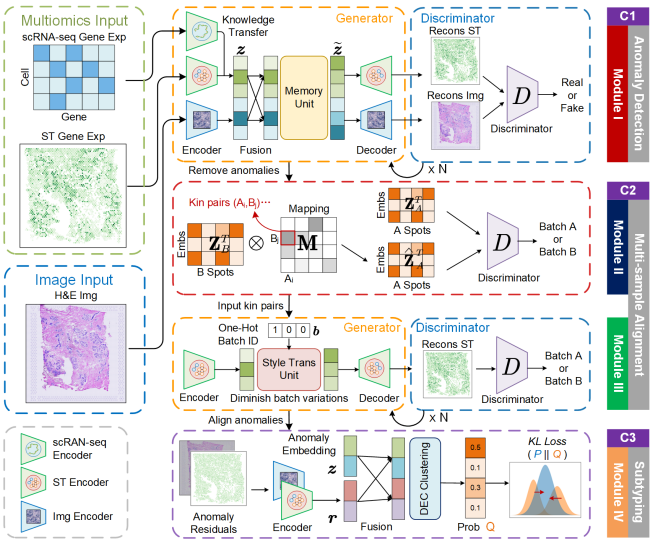


Figure 1. Framework of STANDS.

C1: Trains a GAN on reference data to reconstruct normal spots from multimodal features. Anomalous spots in target datasets are identified by high reconstruction deviations (anomaly scores).

C2: Aligns target datasets to the reference space, reducing batch effects, using cooperative GANs and style-transfer based on biologically similar spot pairs.

C3: Merges embeddings and reconstruction residuals of aligned anomalies for clustering them into distinct subtypes.

2.2. SpacialID [4]

Spatially resolved transcriptomics offers a powerful means to explore gene expression profiles alongside the spatial organization of cells in their native state. However, its utility is often limited by low transcript detection sensitivity or restricted gene throughput. Accurately annotating cell types in spatial transcriptomics data to unravel biological processes at the single-cell level remains a significant challenge. To address this, *Spatial-ID*, a supervision-based cell typing approach that integrates reference single-cell RNA-seq data with the spatial context provided by spatial transcriptomics. Through extensive benchmarking on publicly available datasets.

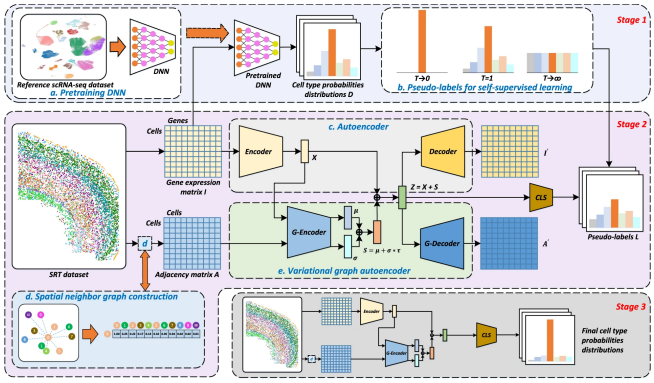


Figure 2. Overview of Spatial-ID.

2.3. Comparision between STANDS and SpacialID

While both methods utilize spatial transcriptomics to address challenges in tissue analysis, they differ in focus:

- STANDS emphasizes anomaly detection and biological heterogeneity characterization through unsupervised GAN-based techniques.
- Spatial-ID prioritizes cell-type annotation using supervised learning with reference datasets.

STANDS	Spatial-ID
Anomaly detection and characterization.	Cell-type annotation in spatial data.
Unsupervised GAN-based framework.	Supervised, leveraging reference RNA-seq data.
Detects anomalies, reduces batch effects, and clusters anomalies into subtypes.	Integrates spatial data with reference knowledge for precise annotations.
Handles heterogeneity, addresses batch effects, and provides a comprehensive pipeline.	Leverages prior knowledge, scalable to 3D tissues, and excels in benchmarks.
Requires computational resources and depends on reference diversity.	Relies on quality of reference data, focuses mainly on annotation.
Identifying anomalies and heterogeneous tissue regions.	Annotating cell types in spatial transcriptomics.

Table 1. Comparison of STANDS and Spatial-ID

3. Datasets and Benchmarking

3.1. Data Availability [5]

Since STANDS is the only method specifically designed for the Detection and Dissection of Anomalous Tissue Domains (DDATD), we used the dataset that STANDS employs for benchmarking and evaluation purposes. Download the raw STANDS dataset

3.2. Data Preprocessing and Filtering

View all the datasets generated and used

We followed the standard preprocessing pipeline, keeping the top 3000 highly variable genes (HVGs) for further analysis. This included steps like normalization, feature selection, and other necessary data transformations.

For anomaly detection, we focused on Cell-Cell Communication data and Pathway Activity Inference to identify anomalous tissue domains in Human Breast Cancerous Cell data.

Code to Process and Filter Raw Data

3.3. Data Generation: Cell-Cell Communication [6]

To generate cell-cell communication data, we utilized the *stlearn* package, which allows for the analysis and modeling of cell interactions within spatial transcriptomics datasets.

Code to Generate Cell Cell Communication Data

3.3.1. Overview of stLearn CCI analysis

The overall steps in the stLearn cell-cell interaction (CCI) analysis pipeline are:

- Load known ligand-receptor gene pairs. (here we are using connectomeDB2020_lit)
- Identify spots where significant interactions between these pairs occur.
- For each LR pair and each celltype-celltype combination, count the instances where neighbours of a signficant spot for that LR pair link two given cell types.
- Identify signficant interactions with p<.05 from cell type information permutation.
- Visualise the CCI results.

3.3.2. Visualization of Generated LRs Data

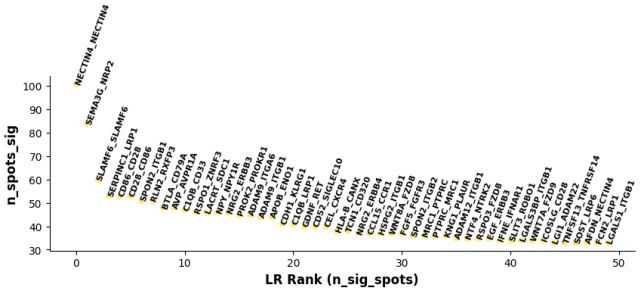


Figure 3. LRs by significant spots.

3.4. Data Generation: Pathway Activity Inference [3]

For generating data based on pathway activity, we used the *Decoupler* tool, which integrates with the Progeny database to provide pathway activity scores for the cells in our dataset. *PROGENy* (Pathway RespOnsive GENes), a comprehensive resource for pathway activity inference from gene expression data. PROGENy leverages a curated collection of pathways and their associated target genes, with weights assigned to each interaction based on experimental evidence.

Code to Generate Pathway Inference Data

3.4.1. Overview of the PROGENy Model

The PROGENy model uses weighted gene sets to infer pathway activities. For this study, we utilized the human pathway weights and focused on the top 500 responsive genes ranked by p-value. The model allows robust inference of pathway activities, enabling us to identify biological processes associated with anomalous tissue domains.

Below is a brief description of each pathway included in the PROGENy model:

- **Androgen**: Involved in the growth and development of male reproductive organs.
- **EGFR**: Regulates growth, survival, migration, apoptosis, proliferation, and differentiation in mammalian cells.
- **Estrogen**: Promotes the growth and development of female reproductive organs.
- **Hypoxia**: Promotes angiogenesis and metabolic reprogramming under low oxygen levels.
- **JAK-STAT**: Involved in immunity, cell division, cell death, and tumor formation.
- **MAPK**: Integrates external signals and promotes cell growth and proliferation.

- **NFkB**: Regulates immune response, cytokine production, and cell survival.
- **p53**: Regulates the cell cycle, apoptosis, DNA repair, and tumor suppression.
- **PI3K**: Promotes cell growth and proliferation.
- **TGFβ**: Involved in development, homeostasis, and tissue repair.
- **TNFα**: Mediates hematopoiesis, immune surveillance, tumor regression, and protection from infection.
- **Trail**: Induces apoptosis.
- **VEGF**: Mediates angiogenesis, vascular permeability, and cell migration.
- **WNT**: Regulates organ morphogenesis during development and tissue repair.

3.4.2. Implementation with Decoupler

To infer pathway activities, we employed the **decoupler** package, which provides an efficient framework for implementing the PROGENy model. Using the curated weights and responsive genes, we calculated pathway activity scores for each spatial transcriptomic sample. These scores serve as indicators of pathway activation, enabling the detection of anomalies within tissue domains.

For each cell in our dataset, it fits a linear model that predicts the observed gene expression based on all pathways' Pathway-Gene interactions weights. Once fitted, the obtained t-values of the slopes are the scores. If it is positive, we interpret that the pathway is active and if it is negative we interpret that it is inactive.

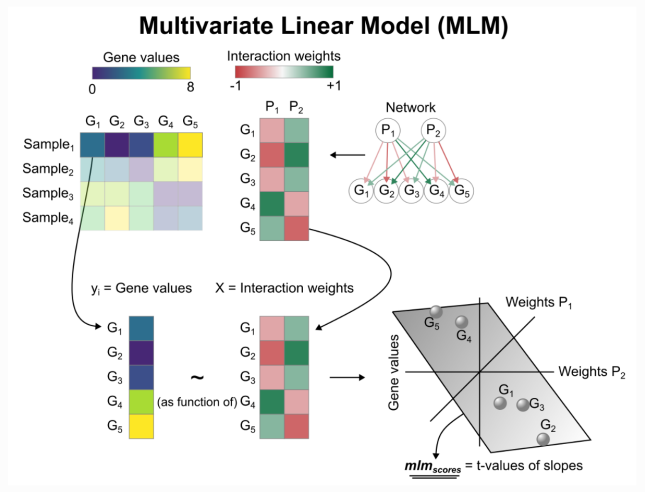
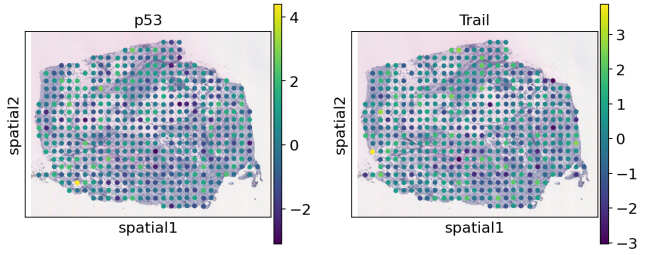


Figure 4. Overview of PROGENy Model.

3.4.3. Visualization of Generated Pathway Activity Data



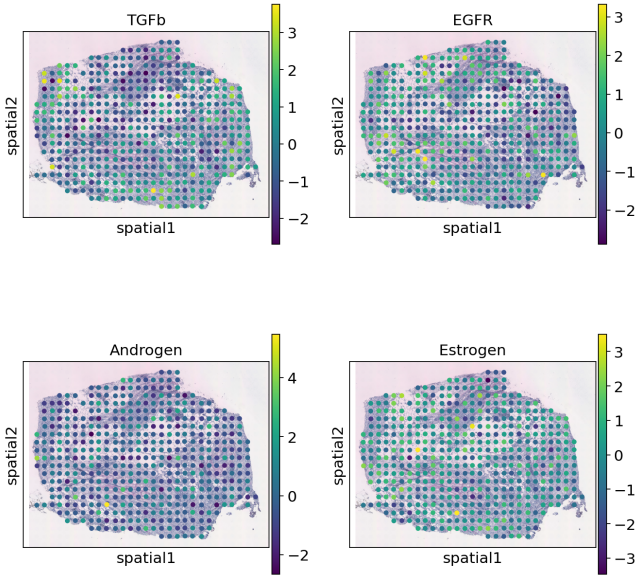


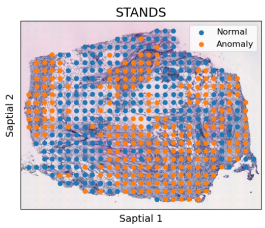
Figure 5. Spatial visualization of important pathways.

Some of the most important pathways involved in cancer biology are the ErbB family pathway, the p53-mediated apoptosis pathway, transforming growth factor (TGF)-beta family pathway, and the GSK3 signaling pathway. [2]

3.5. Benchmarking

We use STANDS as a benchmark to evaluate and compare our method's performance. The *Tumor Dataset* includes annotations distinguishing between normal and tumor spots, which we have utilized as ground truth for our analyses.

Figure 6. Ground Truth of STANDS Tumor Dataset. (Red- Tumor, Blue- Normal



Metric	Value
Accuracy	0.845824
F1	0.775000
SGD_degree	0.568125
SGD_cc	0.328555

Figure 7. STANDS Evaluation

4. Our Approaches

4.1. K-Mean Clustering with Pathway Activity Data

4.1.1. Introduction

This method applies K-Means clustering to detect anomalies in pathway activity data derived from cellular analysis. The goal is to identify deviations in tumor samples compared to normal cellular behavior by analyzing pathway activity features. Key pathways under investigation include *EGFR*, *NFkB*, *TGFb*, and *p53*. These pathways are central to cellular signaling and are known to exhibit dysregulated activity in diseases such as cancer.

Code for: K-Mean Clustering with Pathway Activity Data

4.1.2. Mathematical Background

Scaling the Data

The z-score normalization for each feature is given by:

$$x_{\text{scaled},i} = \frac{x_i - \mu}{\sigma}$$

Where:

- x_i: Original value of the i-th feature.
- μ: Mean of the feature in the normal dataset.
- σ: Standard deviation of the feature in the normal dataset.

K-Means Clustering

K-Means clustering minimizes the intra-cluster variance, represented as:

$$\text{Loss} = \sum_{i=1}^{N} ||x_i - c_{k_i}||^2$$

Where:

- x_i: Data point in the dataset.
- c_{k_i}: Centroid of the cluster k_i to which x_i belongs.
- $||\cdot||$: Euclidean distance norm.
- N: Total number of data points.

The centroid for cluster k is calculated as:

$$c_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} x_i$$

Where:

- C_k: Set of points belonging to cluster k.
- $|C_k|$: Number of points in cluster k.

Distance Calculation for Anomaly Detection

The distance from a data point x_i to its assigned cluster centroid c_{k_i} is computed using the Euclidean distance:

$$d(x_i, c_{k_i}) = \sqrt{\sum_{j=1}^{p} (x_{ij} - c_{k_i,j})^2}$$

Where:

- x_{ij}: j-th feature of the i-th data point.
- $c_{k_i,j}$: j-th feature of the centroid c_{k_i}.
- p: Total number of features.

Anomaly Threshold

The threshold for anomaly detection is determined using the 75th percentile of the distance distribution:

$$\text{Threshold} = \text{Percentile}_{75}(d(x_i, c_{k_i}))$$

Data points with distances exceeding this threshold are labeled as anomalies.

$$\text{Prediction}(x_i) = \begin{cases} \text{"normal"} & \text{if } d(x_i, c_{k_i}) \leq \text{Threshold} \\ \text{"anomaly"} & \text{if } d(x_i, c_{k_i}) > \text{Threshold} \end{cases}$$

4.1.3. Methodology

Training K-Means

1. Fit a K-Means model with 'nclusters=1' on the scaled normal data.
2. Compute the centroid of normal pathway activity data using the formula for ccc.

Testing Tumor Data

1. Scale tumor pathway activity data using the same mean and standard deviation as the normal data.
2. Compute Euclidean distances between tumor samples and the normal centroid.

Anomaly Detection

1. Identify a threshold (Q3) based on the distance distribution.
2. Assign predictions based on whether the distance exceeds the threshold.

4.1.4. Results

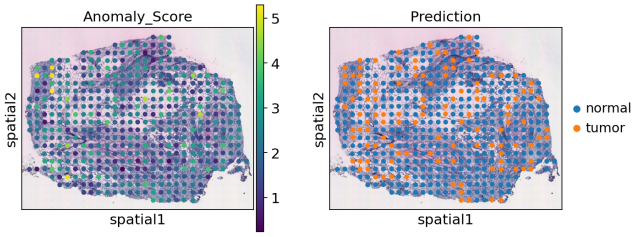


Figure 8. Pathway Activity K-mean Clustering Evaluation

Metric	Normal	Tumor	Macro Avg	Weighted Avg
Precision	0.65	0.32	0.49	0.54
Recall	0.74	0.24	0.49	0.57
F1-Score	0.69	0.27	0.48	0.55
Support	307	160	467	467
Accuracy			**0.57**	

Table 2. Classification Metrics and Report

4.1.5. Advantages

1. *Simple and Interpretable*: Straightforward clustering and distance-based anomaly scoring.
2. *Unsupervised*: No need for labeled training data.

4.1.6. Limitations

1. *Linear Assumptions*: Euclidean distance assumes linear separability, which may not hold in biological data.
2. *Single Cluster Representation*: Using one centroid may oversimplify the complex variability in normal data.
3. *Threshold Sensitivity*: Requires fine-tuning for optimal anomaly detection.

4.2. Isolation Forest with Pathway Activity Data

4.2.1. Methodology

It isolates anomalies by creating random splits in the data. The easier it is to isolate a point, the more likely it is an anomaly. The anomaly score s(x) reflects how isolated a point is in the dataset, and this score

is used to classify cells as normal or anomalous. The technique is particularly effective when anomalies are rare and distinct from the rest of the data, which makes it suitable for tasks like detecting tumor cells in a biological dataset.

The model constructs multiple Isolation Trees by randomly selecting features and split points. After constructing the trees, it calculates the path length for each data point, which determines how isolated the point is.

Anomaly Score Calculation: The anomaly score $s(x)$ is computed for each data point x based on its path length:

$$s(x) = 2^{-\frac{E(h(x))}{c(N)}}$$

Where:

$$E(h(x))$$

is the path length for the point x, and $c(N)$ is the expected path length for a random point.

To achieve this we have leveraged *sklearn's IsolationForest* by fitting the reference data and predicting for the diseased dataset.

4.2.2. Results

Metric	Value
Accuracy	0.60
Precision	0.44
Recall	0.60
F1-Score	0.50

Table 3. Isolation Forests with Pathway Inference Data- Performance Metrics

4.3. Auto-Encoder with Pathway Activity Data

4.3.1. Introduction

An autoencoder is a type of neural network that learns to encode the input data into a lower-dimensional space and then decode it back to the original space. The model is trained by minimizing the reconstruction error, which measures how accurately the model can recreate the input data. Anomalous data points, which differ significantly from normal points, will have higher reconstruction errors because the model has not seen similar data during training.

In this report, we train an autoencoder on "normal" pathway activity data and use the reconstruction error (i.e., the Mean Squared Error or MSE) to identify anomalies in "tumor" data.

Code for Auto-Encoder with Pathway Activity Inference

4.3.2. Mathematical Background

Autoencoder Loss Function (MSE)

The loss function used to train the autoencoder is the **Mean Squared Error (MSE)**. This loss function penalizes large reconstruction errors, helping the model to learn to encode the data into a compact latent space and decode it back as accurately as possible. The MSE is given by:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\|X_i - \hat{X}_i\|^2$$

Where:

- N is the number of data points,
- X_i is the i-th input data point,
- $\hat{X}_i$ is the reconstructed data point.

Anomaly Score

The **Anomaly Score** for each tumor data point is calculated as the reconstruction error (MSE). Data points with high MSE are considered anomalous because they deviate from the patterns learned from the normal data. The anomaly score is computed as:

$$s(x) = \frac{1}{D}\sum_{j=1}^{D}(X_{ij} - \hat{X}_{ij})^2$$

Where:

- D is the number of features,
- X_{ij} is the feature value for the i-th data point and j-th feature,
- $\hat{X}_{ij}$ is the reconstructed feature value for the same data point.

Thresholding for Anomalies

The threshold for anomaly detection is set based on the percentile of reconstruction errors (MSE). For instance, using the 50th percentile would flag points whose reconstruction error is higher than half of all the data points.

4.3.3. Implementation

1. *Data Preprocessing*: The normal pathway data is scaled using StandardScaler.
2. *Model Definition*: A simple autoencoder is defined with an encoder and decoder. The encoder reduces the input to a latent space of size 6, and the decoder reconstructs the original data.
3. *Training*: The autoencoder is trained on normal data to learn the underlying patterns. The model uses the Adam optimizer and the MSE loss function.
4. *Anomaly Detection*: The model is tested on tumor data, and the reconstruction error (MSE) is calculated. Points with high reconstruction error are flagged as anomalies (tumor).
5. *Results*: The predictions and anomaly scores are printed, indicating which data points are considered tumor cells based on their reconstruction error.

4.3.4. Results

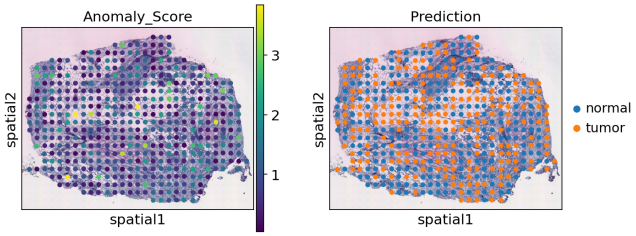


Figure 9. Pathway Activity Auto-Encoder, Evaluation

Metric	Normal	Tumor	Macro Avg	Weighted Avg
Precision	0.71	0.39	0.55	0.60
Recall	0.54	0.57	0.55	0.55
F1-Score	0.61	0.46	0.54	0.56
Support	307	160	467	467
Accuracy			**0.55**	

Table 4. Classification Metrics and Report, Auto-Encoder

4.4. GAN with Cell-Cell Communication Data

4.4.1. Introduction

In this study, we implement an anomaly detection model using Generative Adversarial Networks (GANs) to identify anomalous spots or cells in cell-cell communication datasets. These datasets consist of observations with spatial and communication data, which can be processed to detect outliers indicative of disease states (e.g., in pancreatic cells). The goal is to train a GAN to learn the normal patterns of healthy cells and then use the trained model to detect anomalous behavior in diseased cells.

4.4.2. Methodology

Data Preprocessing

We begin by loading and preprocessing three different healthy datasets that contain cell-cell communication information. Each dataset is represented in the form of **AnnData** objects containing communication data between cells, which includes information such as ligand-receptor (LR) scores and spatial coordinates of the cells (spots).

To ensure that the feature dimensions are consistent across all datasets, the LR scores for each dataset are padded or truncated to match the largest feature dimension among all datasets. This step ensures that the neural network can process the datasets uniformly.

Graph Construction

A graph-based approach is used to model the spatial relationships between cells. A k-nearest neighbors (k-NN) graph is constructed based on the spatial coordinates of the cells. The nodes of the graph represent the individual cells (or spots), and the edges represent their spatial proximity. The **LR scores** are used as the features associated with each node in the graph. This graph structure is crucial for capturing the spatial dependencies and relationships in cell-cell communication.

- Each dataset is represented as a **k-nearest neighbors (k-NN) graph** using spatial coordinates (`spatial`).
- The graph's nodes represent cells, and node features correspond to their LR scores.

Mathematically, for each cell i:

$$\text{Neighbors}(i) = \{j \mid \text{distance}(i, j) \text{ is among the smallest } k+1\}$$

Here, self-loops are added for computational stability.

Generative Adversarial Network (GAN)

A GAN consists of two primary components: the **Generator** and the **Discriminator**.

- **Generator**: The Generator takes a latent vector z (a random noise vector) and generates fake embeddings (fake node representations) that aim to resemble the real embeddings obtained from the healthy dataset's GNN model.
- **Discriminator**: The Discriminator evaluates both real embeddings (from the GNN) and fake embeddings (from the Generator). Its objective is to distinguish between real and fake embeddings. The Discriminator is trained to output a high value (close to 1) for real embeddings and a low value (close to 0) for fake embeddings.

GNN Embeddings The **Graph Neural Network (GNN)** processes the combined graph to produce cell embeddings.

- **Input Layer**: LR scores are node features, X.
- **Graph Convolutional Layers**:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(l)}W^{(l)}\right)$$

where:

  - $\tilde{A}$: adjacency matrix with self-loops.
  - $\tilde{D}$: diagonal degree matrix.
  - $W^{(l)}$: layer-specific learnable weights.
  - $H^{(l)}$: embeddings from layer l.
  - σ: activation function (ReLU).

Training

- **Discriminator Loss**:

$$\mathcal{L}_D = -\mathbb{E}_{x\sim\text{real}}[\log D(x)] - \mathbb{E}_{z\sim\mathcal{N}(0,I)}[\log(1 - D(G(z)))]$$

- **Generator Loss**:

$$\mathcal{L}_G = -\mathbb{E}_{z\sim\mathcal{N}(0,I)}[\log D(G(z))]$$

- **Optimization**: Uses Adam optimizer with gradient backpropagation.

Testing and Anomaly Detection

- Compute **embeddings** for diseased data using the trained GNN.
- Generate **fake embeddings** using the generator.
- Calculate **anomaly scores** as the L_2-norm (Euclidean distance) between real and generated embeddings:

$$\text{Anomaly Score}(i) = \|\text{Embedding}_{\text{real}}(i) - \text{Embedding}_{\text{fake}}(i)\|_2$$

- Define a **threshold** (e.g., 60th percentile of scores) to identify anomalies.

Results

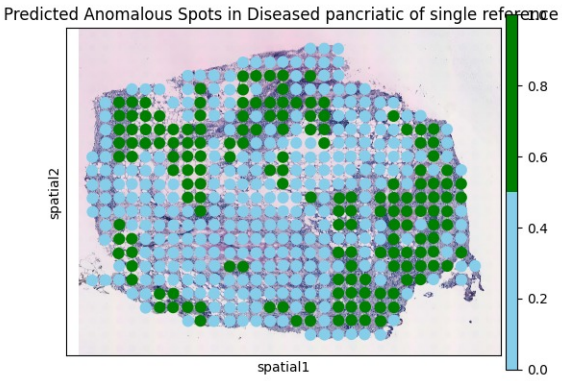


Figure 10. Cell-Cell Communication GAN (single ref), Evaluation

Metric	Value
Accuracy	0.561
Precision	0.39
Recall	0.41
F1-Score	0.40

Table 5. GAN Evaluation metrics with single reference

With some fine-tuning, parameter adjustment and using 3 reference datasets to train we obtained following results:

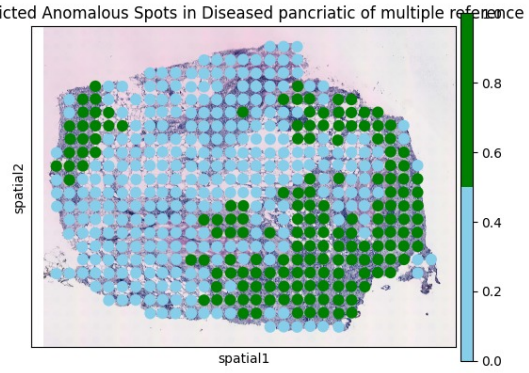


Figure 11. Cell-Cell Communication GAN, Evaluation

Conclusion

In this study, we successfully applied a **Generative Adversarial Network (GAN)** to detect anomalous spots in cell-cell communica-

Metric	Value
Accuracy	0.81
Precision	0.64
Recall	0.65
F1-Score	0.65

Table 6. GAN Evaluation metrics with multiple references

tion data. The **Generator** learned to produce realistic fake embeddings for healthy cells, while the **Discriminator** learned to distinguish between real and fake embeddings. After training, the **Generator** was used to create fake embeddings for diseased cells, and the **Euclidean distance** between the real and fake embeddings was used as an **anomaly score**. This method effectively identified anomalous spots or cells, potentially indicating disease states.

This approach demonstrates the power of GANs in learning complex data distributions and using that knowledge to detect outliers in high-dimensional data, such as cell-cell communication networks.

Code for GAN Cell-Cell Communication for Anomaly Detection

References

[1] K. Xu, Y. Lu, S. Hou, *et al.*, "Detecting anomalous anatomic regions in spatial transcriptomics with STANDS", *Nature Communications*, vol. 15, no. 1, p. 8223, 2024.

[2] *Cancer Biology Pathways, Thermofisher*. [Online]. Available: https://www.thermofisher.com/in/en/home/life-science/antibodies/antibodies-learning-center/antibodies-resource-library/cell-signaling-pathways/cancer-biology-pathways.html.

[3] *Generating Pathway Activity Inference with Decoupler*. [Online]. Available: https://decoupler-py.readthedocs.io/en/latest/notebooks/usage.html.

[4] *Shen, R., Liu, L., Wu, Z. et al. Spatial-ID: a cell typing method for spatially resolved transcriptomics via transfer learning and spatial embedding. Nat Commun 13, 7640 (2022).* [Online]. Available: https://rdcu.be/d1tP4.

[5] *STANDS Datasets*. [Online]. Available: https://catchxu.github.io/STANDS/start/#installation.

[6] *stLearn Cell-Cell Interaction Analysis*. [Online]. Available: https://stlearn.readthedocs.io/en/latest/tutorials/stLearn-CCI.html.