GRAPH PSEUDOTIME ANALYSIS AND NEURAL STOCHASTIC DIFFERENTIAL EQUATIONS FOR ANALYZING RETINAL DEGENERATION DYNAMICS AND BEYOND

Dai Shi*, **Kuan Yan***, **Lequan Lin***, **Yue Zeng**, **Ting Zhang**, **Jialing Zhang**

Dmytro Matsypura, **Mark C. Gillies**, **Ling Zhu**†, **Junbin Gao**†

The University of Sydney, Australia

ABSTRACT

Understanding the progression of disease at the molecular level usually requires capturing both the structural dependencies between pathways and the temporal dynamics of how diseases evolve. In this work, we resolve the former challenge by developing a biologically informed graph-forming method to efficiently construct pathway graphs for subjects from our newly curated transcriptomic dataset of JR5558 mice that spontaneously develop neovascularization beneath their retinas. We then developed Graph-level Pseudotime Analysis (GPA) to infer graph-level trajectories that reveal how the disease progresses at the population level rather than in individual subjects. Based on the trajectories estimated by GPA, we identify the most sensitive pathways that drive transitions between disease stages. In addition, we measure changes in pathway features using neural stochastic differential equations (SDEs), which enable us to formally define and compute pathway stability and disease bifurcation points (points of no return)—two fundamental problems in research on disease progression. We have extended our theory to allow pathways to interact with each other, enabling a more comprehensive and multi-faceted characterization of disease phenotypes. Comprehensive experimental results demonstrate the effectiveness of our framework in reconstructing pathway dynamics, identifying critical transitions, and providing novel insights into the mechanistic understanding of the evolution of disease.

1 INTRODUCTION

Understanding disease progression through the lens of molecular pathways is crucial for identifying critical transitions and potential intervention targets (Liu and Chance, 2013). Although recent advances in time-series forecasting (Tripto et al., 2020), differential equations (Xie et al., 2024), and dynamical simulations (Xu et al., 2024; Satorras et al., 2021) have significantly improved our ability to quantify disease evolution, their applicability remains constrained when relevant biological signals cannot be consistently measured within the same individual. For example, single-cell transcriptomic profiling provides crucial insights into disease states, but requires cell lysis, making longitudinal observations within the same subject infeasible (Chen et al., 2022). This limitation presents a fundamental challenge in modeling the path-level disease dynamics and tasks such as identifying the key path-sensitive pathways (SP), the stability of the pathway (Khasminskii, 2012), and the estimation of the bifurcation of the disease (Flores-Garza et al., 2023). In this work, we solve these challenges by developing Graph Pseudotime Analysis (GPA), which extends the traditional pseudotime analysis (Qiu et al., 2017) to the object's pathway graph profiles formed by our biology-informed graph-forming algorithm. The GPA estimated trajectories among graphs offer us insights

*Equal Contribution. To QuanQuan 🐱, a gift in solemn tribute, for thy love unyielding and thy company unfaltering; in memory's hallowed halls, thou shalt dwell forevermore.

†Corresponding to {ling.zhu, junbin.gao}@sydney.edu.au.

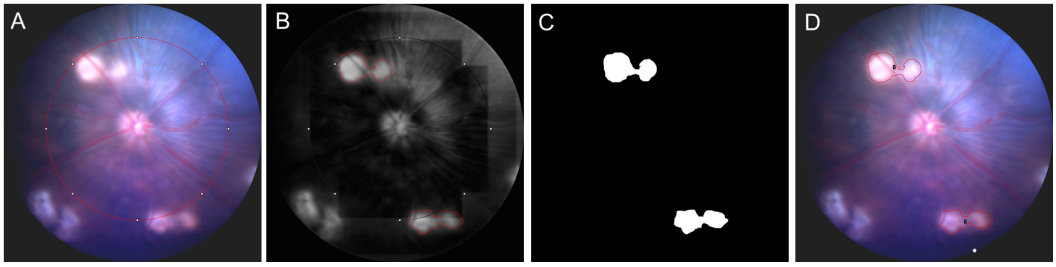


Figure 1: Illustration of the key steps of the quantification method. (A) A circular region of interest (ROI) with a $283\mu m$ radius centered on the optic nerve head was defined. (B) The image was converted to 8-bit grayscale, the background was removed using a $200\mu m$ rolling ball algorithm, and lesions within the ROI were manually outlined using the freehand selection tool. (C) A threshold was applied to accurately segment the lesion areas. (D) The selected lesions were verified in the original color fundus image.

into identifying the SPs that are likely to induce the changes of disease stages. Furthermore, we model these trajectories using neural partial stochastic differential equations (SDEs) (Li et al., 2020), which provide a principled approach for analyzing pathway stability and detecting disease bifurcation points. Our comprehensive analysis and experimental results establish a novel paradigm for studying the genomic dynamics of pathway profiles, offering new insights into disease progression and potential therapeutic interventions. We summarize our contributions as follows.

- We provide a novel mouse dataset (namely JR5558) that integrates transcriptomics data for studying pathway-level disease dynamics. Furthermore, we form our research objects molecular pathway graphs based in a biology-informed manner. We then identify the SPs from our data by the graph regression techniques.
- We develop GPA to uncover the orders between graphs, and we apply temporal GNN models to show the SPs that drive the stage change of the disease.
- We further analyze the estimated pathway trajectories using SDEs, rigorously defining pathway-level dynamic stability and establishing conditions for disease bifurcation. By leveraging our learned SDEs, we precisely identify the timing of disease bifurcations and assess the stability of pathways. Furthermore, we extend our SDE framework to incorporate interactions of pathway, enabling a more comprehensive and multi-faceted characterization of disease phenotypes (Appendix C.2).
- We conduct comprehensive numerical experiments based on our developed approaches above and provide biological interpretations of our findings and suggestions for future studies.

2 Preliminaries

Notations We leave the summary of the related works in Appendix A but introduce the necessary notation as follows. Throughout this paper, we let $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{W})$ be the weighted graph with $\mathcal{V}$ and $\mathcal{E}$ be the sets of nodes and edges, and $\mathbf{W} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ be the matrix contains all the edge weights with $w_{i,j} > 0$ if $i \sim j$. We also denote $\mathbf{A}$ and $\mathbf{L} \in \mathbb{R}^{N\times N}$ be the adjacency and Laplacian matrices of the graph, respectively, where N is the number of nodes. Furthermore, we let $\{(\lambda_i, \mathbf{u}_i)\}_{i=1}^N$ be the set of eigen-pairs of $\mathbf{L}$ where $\mathbf{u}_i$ are the row vectors of $\mathbf{U}$ obtained from the eigendecomposition of $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$. Lastly, the feature matrix of the graph is denoted as $\mathbf{X} \in \mathbb{R}^{N\times d}$, where d standards for the feature dimension.

JR5558 Dataset We present a concise overview of the JR5558 dataset to establish a clear foundation for the analyses that follow. Specifically, the JR5558 mouse model, obtained from Jackson Laboratory (JAX stock #005558), is a well-established system for studying subretinal neovascular which is a feature of neovascular age-related macular degeneration (nAMD). This study utilized bulk RNA-seq data collected from the retinas of 23 eight-week-old male JR5558 mice. RNA extraction was

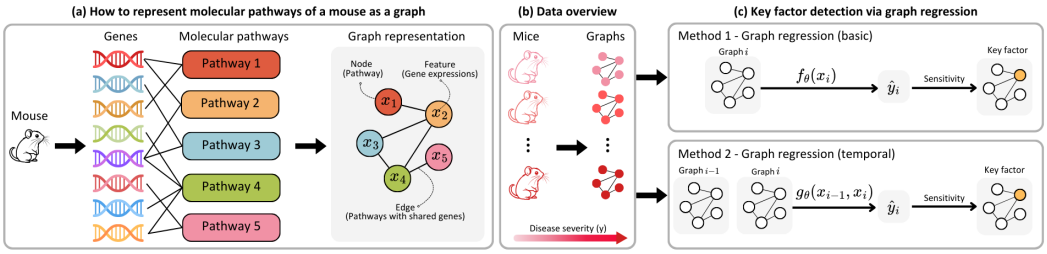


Figure 2: Illustration on how the pathway graphs are formed and how SPs are identified through two types of graph regressions.

performed using the GenEluteTM Single Cell RNA Purification Kit (Sigma Aldrich, RNB300), with library preparation, quality control, and sequencing conducted by Novogene.

The RNA-seq dataset contains expression levels for 56,748 genes, normalized as Fragments Per Kilobase of transcript per Million mapped reads (FPKM), of which 24,888 genes with corresponding Entrez IDs were used for pathway analysis. In addition to the RNA-seq data, fundus photographs were taken to quantify subretinal lesion severity. The lesion area was calculated as a percentage of the total retinal area using an automated image analysis pipeline. The key steps include delineating the optic nerve region, adjusting the image threshold, and validating the selected lesion area as illustrated in Figure 1. This methodology ensures consistent and accurate quantification of lesion severity.

3 Uncover The Disease Sensitive Genetic Pathways

In this section, we show in detail how our graph-based methods can efficiently extract the key sensitive pathways (SP) via JR5558. This involves identifying the most sensitive pathways when we treat the disease development as a whole, i.e., there is no temporal (sequential) relationship between objects. On the other hand, we also aim to uncover those SPs that drive the stage changes of the disease by building a pseudotime among objects' graph profiles. Figure. 2 provides an illustration of how our model works.

3.1 Biology-informed Graph Forming

At the initial stage, we aim to form graphs that represent the pathway profile of each mouse. Specifically, we group genes (total 24,888) into 343 canonical molecular pathways by developing a web scraper extract detailed biological pathway information related to Mus musculus from the KEGG database[1]. We note that there are 9703 unique genes (out of 24,888) selected for forming genetic pathways, suggesting a large reduction in the complexity of the subsequent computations. The resulting genetic pathways are with different gene lengths, ranging from 2 to 1168. For each mouse in our dataset, we form a pathway graph with 343 nodes, i.e., genetic pathways as nodes[2]. As the number of genes is different between pathways, and there are 9703 unique genes involved in forming pathways, for the sake of the feature propagation in the subsequent GNN models, we initially set the dimension of the feature as 9703. Node features are directly obtained from the FPKM (see the introduction of JR5558 dataset in Section 2) with the rest part (i.e., the part does not match 9703 unique genes) as zeros. Furthermore, we build edges between nodes if gene overlap(s) exists between two nodes, and the edge weight is computed by the Euclidean distance between node features. Accordingly, for each mouse, the node feature matrix is $\mathbf{X} \in \mathbb{R}^{343\times 9703}$, and the graph connectivity is the same for all mice but with different edge weights. Lastly, we formed 23 graphs for these mice based on the aforementioned paradigm. We finally note that it is commonly seen in the recent literature that edge weights are computed via the distance between node features such as protein graphs (Xu et al., 2024; Wang et al., 2023) and electrical networks (Black et al., 2023).

[1] https://www.genome.jp/kegg-bin/show_organism?menu_type=pathway_maps&org=mmu

[2] In the sequel, we will use pathways and nodes interchangeably.

3.2 Graph Regression and Detection of Disease Sensitive Pathways

Graph-level regression leverages graph adjacency and node feature information to approximate the target graph labels usually denoted as a continuous variable as $\mathbf{y} \in \mathbb{R}^m$, where m is the number of graphs. The process can be formulated as

$$f_{\boldsymbol{\theta}}(\mathbf{A}(i), \mathbf{X}(i)) \approx \mathbf{y}_i, \quad i \in [1, m], \tag{1}$$

where we let $f_{\boldsymbol{\theta}}$ be the learned graph regression model, usually containing GNN models with additional pooling and regression layers. Please check the details of the model implementation via Section 5 and Appendix C. We note that the learned parameters (i.e., $\boldsymbol{\theta}$) are shared over all graphs. After the model is trained, we compute the model's sensitivity to all nodes. Specifically, for each node $\mathbf{x}_i$, we compute is sensitivity score as

$$\text{Sensitivity}(\mathbf{x}_s) = \sum_{i=1}^{m} \left\| \frac{\partial \widehat{\mathbf{y}}_i}{\partial \mathbf{x}_s} \right\|, \tag{2}$$

where we denote $\widehat{\mathbf{y}}_i$ as the model prediction based on the input graph i with $\mathbf{A}(i)$ and $\mathbf{X}(i)$. As there are m graphs with different adjacency weights, one shall have m number of the Jacobian norms (i.e., $\|\frac{\partial \widehat{\mathbf{y}}_i}{\partial \mathbf{x}_s}\|$) and we denote the sum of these Jacobian norms of node s be the model's sensitivity on s. We remark that the node Jacobian norm has been widely applied to various GNN research, especially for analyzing the so-called over-squashing problem, in which nodes gradually lose their dependencies after a long-range propagation via GNNs, even though the graph is connected (Shi et al., 2023). Furthermore, the sensitivity scores denoted in Equation (2) can be easily obtained via the Pytorch loss backward gradient once the model is trained, demonstrating the effectiveness of leveraging such definition on finding the SPs of the disease.

3.3 Discovering Order Amidst Chaos: Graph Pseudotime Analysis

While our initial graph regression approach efficiently identifies key SPs, practical applications often demand the establishment of graph orders to enable graph-level time series forecasting. More importantly, constructing an ordered sequence of graphs allows for more meaningful clustering analyses, uncovering a biologically significant question:

What are the pathways driving transitions between distinct stages of the disease?

To resolve this problem, we novelly developed graph-level pseudotime analysis (GPA) adopted from the Monocle model and its variants developed in the recent work in (Haghverdi et al., 2016; Qiu et al., 2017). We note that our principal purpose is to reveal the temporal orders of pathway graphs rather than estimating trajectories (dynamics) of the single mRNA, which serves as the main objective of the pseudotime methods (Prabhakaran et al., 2016; Gut et al., 2015). Thus, we leave the discussion between different pseudotime methods and their potential extension to graphs as future works.

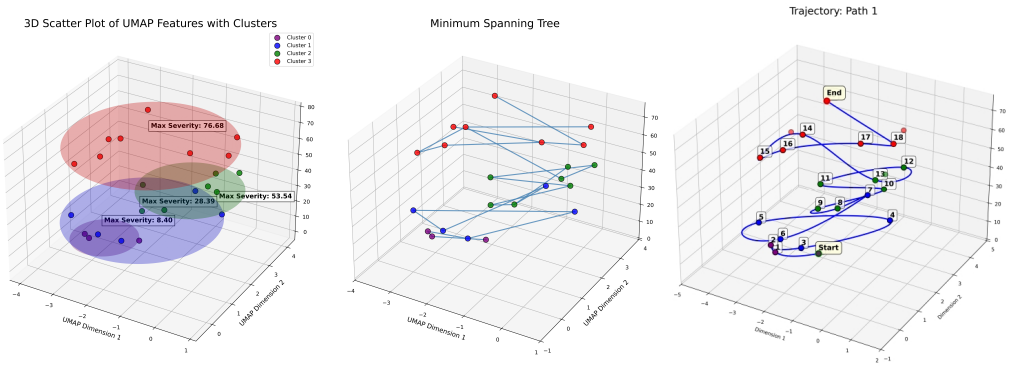
Graph Dimensionality Reduction via Positional Encoding Our GPA starts with the dimensionality reduction of the graphs. First, we conduct the positional encoding to enrich the graph feature information with their spectral characteristics. That is for each node feature matrix $\mathbf{X}(i)$ from the graph $\mathbf{A}(i)$, we have

$$\widehat{\mathbf{X}}(i) = \text{PE}(\mathbf{X}(i), \mathbf{A}(i)) = [\mathbf{X}(i) \mid\mid \mathbf{U}(i)], \tag{3}$$

where $[\cdot||\cdot]$ denotes the concatenation of the vectors, and $\mathbf{U}(i)$ is the eigenvector matrix of graph i. We then conduct mean and maximum pooling operations to ensure the final output vector (denoted as $\widehat{\mathbf{x}}(i)$) is sufficiently representative in terms of both node feature and graph topology information. That is

$$\widehat{\mathbf{x}}(i) = [\text{MaxPool}(\widehat{\mathbf{X}}(i)) \mid\mid \text{MeanPool}(\widehat{\mathbf{X}}(i))]. \tag{4}$$

One can check that for each graph i, its $\widehat{\mathbf{x}}(i)$ is with the size $\mathbb{R}^{N+d}$. Then, their low-dimensional representations are obtained through UMAP (McInnes et al., 2018), that is $\widehat{\mathbf{z}}(i) = \text{UMAP}(\widehat{\mathbf{x}}(i))$. In practice, we usually let the dimension of $\widehat{\mathbf{z}}$ to be 2, and to sufficiently leverage the disease severity



(a) (b) (c)

Figure 3: Process of generating graph trajectory via GPA. (a): Graph dimensionality reduction and clustering; (b): Minimum spanning tree; (c): Trajectory estimation using shortest distance method.

information (Shi et al., 2024), we further concatenate each $\widehat{\mathbf{z}}(i)$ with its $\mathbf{y}_i$, i.e., $\mathbf{z}(i) = [\widehat{\mathbf{z}}(i)||\mathbf{y}_i]$. As a result, we obtained three-dimensional representations of all the graphs as shown in Figure 3a. We note that although there are many graph-level dimensionality reduction methods, they are often trainable and designed by fitting various downstream tasks, such as link prediction (Kipf and Welling, 2016) and causality analysis (Ng et al., 2019). As our goal here is to efficiently obtain graph representations, we select those widely applied unsupervised methods (i.e., UMAP) as our tool and leave the discussion on obtaining the optimized graph representation in future works.

Minimum Spanning Tree and Trajectory Estimation After obtaining the low-dimensional representations of the graphs, we aim to establish an intrinsic ordering among them. To achieve this, we first construct a K-nearest neighbors (KNN) graph over the embedded representations $\widehat{\mathbf{z}}$, capturing local similarities in the latent space. We then apply a Minimum Spanning Tree (MST) to refine this structure, ensuring global connectivity while minimizing the total edge weight. This process effectively restructures the KNN graph into a hierarchical framework that preserves essential topological relationships. Biologically, this formulation shifts the focus from *modeling disease progression at an individual level to capturing its systemic evolution across an entire species*, providing a more holistic perspective on disease dynamics. Finally, by fixing pairs of starting and ending objects, e.g., objects with the lowest/highest disease severity, the disease evolution trajectory can be obtained by leveraging the shortest path distance algorithm, which provides an estimate in which every node on the path will travel to the next node with shortest distance (e.g., edge weights). We note that given different starting and ending points for one dataset, the estimated trajectory can be many, suggesting a transaction between distributions, i.e., from the collection (distribution) of all starting points to the ending points. Figure 3 illustrates our trajectory estimation process.

3.4 Temporal GCN and Transitions Between Disease Stages

The estimated trajectory from GPA offers us a chance to measure the evolution of the disease via a time-series way. To sufficiently capture the relationship between the severity of disease and graph spatial (from the node connectivity) temporal (graph time series) features, we deploy recent works on temporal graph neural networks, e.g., Temporal Graph Convolutional Network (TGCN) (Zhao et al., 2019). Temporal GNNs such as TGCN, TGAT (Xu et al., 2020) and their successors, the so-called spatial-temporal GNNs (STGNNs) such as STGCN (Yu et al., 2017) and DCRNN (Li et al., 2017) have been widely applied to many forecasting fields such as traffic flow/speed and time series. Specifically, assuming the time delay (i.e., lag) as 1, then our task is to find

$$\widehat{\mathbf{y}}(t) = g_{\boldsymbol{\theta}}(\mathbf{A}(t), \mathbf{X}(t), \mathbf{A}(t-1), \mathbf{X}(t-1)), \tag{5}$$

where we let $g_{\boldsymbol{\theta}}$ be any temporal GCN model that approximates the ground truth disease severity $\mathbf{y}(t)$ at time t using $\widehat{\mathbf{y}}(t)$ by leveraging time-dependent adjacency and feature information (i.e., $\mathbf{A}(t), \mathbf{A}(t-1), \mathbf{X}(t), \mathbf{X}(t-1)$). We note that here we assign the GPA estimated sequences of the graph as the index denote as (t) (e.g., $\mathbf{X}(t)$) instead of keep using i (e.g., $\mathbf{X}(i)$) which standards of graph i without sequence (same as $\mathbf{y}_i$ to $\mathbf{y}(t)$). Adopted from the TGCN, we start with two layers of

GCN (Kipf and Welling, 2017) followed by the so-called graph recurrent unit (GRU) and the model output (i.e., $\widehat{\mathbf{y}}$) is obtained by a single layer MLP. That is

$$\mathbf{H}(t) = \mathbf{A}(t)\mathbf{X}(t)\mathbf{W}, \quad \widetilde{\mathbf{H}}(t) = \mathrm{GRU}(\mathbf{H}(t)), \quad \widehat{\mathbf{y}}(t) = \mathrm{MLP}(\widetilde{\mathbf{H}}(t)), \tag{6}$$

in which we let $\mathbf{H}$ be the hidden state of the graph nodes and $\mathbf{W}$ learnable weights in GCN. Different from the original TGCN, where all graphs have the same structure, here, since we let the distance between nodes as the edge weights, we dynamically input $\mathbf{A}$ in our revised TGCN, and the model is trained via the MSE loss between $\mathbf{y}(t)$ and $\widehat{\mathbf{y}}(t)$.

Clustering and Transaction Between Disease Stages In order to segment disease into different stages, we provide a Gaussian Mixture Model (GMM) to the low-dimensional representations of the graphs (i.e., $\mathbf{z}$), as illustrated by Figure. 3a, in which clusters are highlighted with different colors. We define the graph pairs $(\mathcal{G}(t), \mathcal{G}(t+1))$ as the transaction pairs between disease stages if $\mathcal{G}(t) \in \mathcal{S}_1$ and $\mathcal{G}(t+1) \in \mathcal{S}_2$, where $\mathcal{S}_1$ and $\mathcal{S}_2$ are the sets (clusters) that contain all the graphs in stage one and two, estimated by GMM. For example, we will have 3 pairs of graphs for four stages of disease in Figure. 3a. To identify the most influential node that drives the changes of the disease stages, one can simply compute $\left\| \frac{\partial \widehat{\mathbf{y}}(t)}{\partial \mathbf{x}_s(t)} \right\|$ similar to Equation (2). The difference is in this case, one only needs to select all **the first graph in each stage**, as based on the formulation of TGCN above, the disease severity (i.e., $\widehat{\mathbf{y}}(t)$) of that graph depends on the adjacency and feature information from both this graph and its previous step. Accordingly, computing $\left\| \frac{\partial \widehat{\mathbf{y}}(t)}{\partial \mathbf{x}_s(t)} \right\|$ directly illustrates how sensitive the current disease severity to the previous graph features.

4 Neural Stochastic Differential Equations and Other Downstream Tasks

With the advance of leveraging GPA to provide pseudotime orders to the graph, we are now interested in analyzing the feature-changing dynamics along with the estimated trajectories. We note that given different starting and ending point settings from the original JR5558 dataset, we obtained three trajectories. Based on our inductive assumption that the transaction between graph pathway features (i.e., $\mathbf{x}$) is governed by stochastic differential equations (SDE) and can be approximated by the neural SDE method such that

$$\frac{\partial \mathbf{x}(t)}{\partial t} = \psi_{\boldsymbol{\theta}}(\mathbf{x}(t), t)dt + \xi_{\boldsymbol{\phi}}(\mathbf{x}(t), t)d\mathbf{B}(t), \tag{7}$$

where we let $\psi_{\boldsymbol{\theta}}$ and $\xi_{\boldsymbol{\phi}}$ be drifting and diffusion terms which measure the deterministic and stochastic changes of the node features, approximated by two neural networks which parametrized by $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, respectively, and $d\mathbf{B}$ is the standard Wiener process. Accordingly, each node $\mathbf{x}$ owns 23 (number of graphs) steps to reach its destination, and based on the graph forming paradigm, we have 343 nodes in each graph. Then we train a model to learn $\psi_{\boldsymbol{\theta}}$ and $\xi_{\boldsymbol{\phi}}$ via torch SDE package (Li et al., 2020).

Downstream Task One: Pathway Stability The construction of neural SDE will enable us to investigate the pathway stability (PS). Specifically, we are interested in the following:

> *Which pathways remain stable throughout disease progression, and which pathways exhibit instability due to external perturbations?*

Understanding this question is fundamental to uncovering the mechanisms underlying disease progression. Stable pathways often represent core regulatory circuits that drive disease dynamics, while unstable pathways are highly responsive to external stimuli and may serve as key intervention targets. Identifying these pathways can provide critical insights into therapeutic strategies, particularly for diseases characterized by stochastic transitions (Gupta et al., 2011; Panegyres, 2022). Numerically, we quantify the PS by analyzing their diffusion terms time variances, that is

$$\mathrm{PS}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} \|\xi_{\phi}(\mathbf{x}(t+1), t+1) - \xi_{\phi}(\mathbf{x}(t), t)\|^2, \tag{8}$$

where $T = 23$ is the total step of the pathway trajectory. The formulation of PS follows directly from classic stochastic stability theory (Khasminskii, 2012), where the diffusion term's behavior dictates long-term stability. As shown in the following Lemma.

Lemma 1 (informal)**.** *Let $\mathbf{x}(t) \in \mathbb{R}^d$ be a stochastic process governed by the Itô stochastic differential equation (SDE) given in Equation (7). Assuming the diffusion term $\xi_\phi(\mathbf{x}(t), t)$ is continuous and differentiable and has bounded partial derivatives, then under mild conditions, the mean-square stability of the system satisfies if $\exists C > 0, \sup_{t\geq 0} \mathbb{E}\|\xi_\phi(\mathbf{x}(t), t)\|^2 > C$, i.e., the system exhibits **mean-square divergence**, thus **unstable** ($\mathbb{E}\|\mathbf{x}(t)\| \to \infty$). On the other hand, if $\lim_{t\to\infty} \mathbb{E}\|\xi_\phi(\mathbf{x}(t), t)\|^2 \to 0$, then the system is **mean-square asymptotically stable**, i.e., $\mathbb{E}\|\mathbf{x}(t)\|^2$ is bounded.*

Full version of Lemma 1 is in Appendix B. As shown in Lemma 1, if the expected magnitude of $\xi_\phi(\mathbf{x}(t), t)$ remains bounded, the system is mean-square stable; otherwise, it diverges. Thus, PS serves as an empirical metric, capturing instability through the variance of $\xi_\phi(\mathbf{x}(t), t)$. We show the proof in Appendix B, where we adapted the proof from the work in (Khasminskii, 2012) and the empirical results via Section 5.

Downstream Task Two: Disease Bifurcation Points Apart from the stability analysis of the estimated SDE, we are also interested in determining whether a *bifurcation point* (BP) aka. *point of no return* exists in the progression of disease at the pathway level, as this potentially leads to irreversible disease progression (Flores-Garza et al., 2023). Specifically, we present the following.

Definition 1 (Point of No Return)**.** *Let $\mathbf{x}(t) \in \mathbb{R}^d$ be a stochastic process governed by the following Itô stochastic differential equation (SDE) given in Equation (7). Define the system stability potential function as*

$$J(\mathbf{x}, t) = -\int \psi_{\boldsymbol{\theta}}(\mathbf{x}, t) d\mathbf{x} \tag{9}$$

We define the point of no return (bifurcation time) as a time t^* at which the pathway undergoes an irreversible transition in its dynamical behavior. Specifically, it is the first time at which at least one of the following conditions holds: (1): Loss of Stability Region, that is $\nabla J(\mathbf{x}, t^*) \approx 0$ and $\frac{d}{dt} J(\mathbf{x}, t^*) > 0$; or (2): Transition to a New Steady State: $\lim_{t\to\infty} \mathbb{E}\|\mathbf{x}(t) - \mathbf{x}(t^*)\| > C, C > 0$; (3): Diffusion Variance Explode: $\sup_{t\geq t^*} \mathrm{Var}(\mathrm{PS}(\mathbf{x}, t)) > \delta$, where δ is a predefined threshold.

One can find that condition (1) checks whether the system reaches the critical point such that deterministic forces vanish $\nabla J(\mathbf{x}, t^*) \approx 0$, but the stability is deteriorating as $\frac{d}{dt} J(\mathbf{x}, t^*) > 0$. Similarly, condition (2) suggests the changes of $\mathbf{x}$ will no longer go back to the velocity at time t^*, meaning the pathway undergoes an irreversible transition into a new state. Finally, condition (3) captures a sudden increase in stochastic fluctuations, signifying a transition into a highly unstable regime. Although the quantity of C and δ vary between datasets, in our implementation, we simply set $C = 1$ and $\delta = 0.1$, and we report our results in Section 5. We further note that, in practice, the disease is detected mostly after its starting point (i.e., $t = 0$). Thus, we omit the trivial bifurcation point obviously at $t = 0$. We finally remark that in Appendix C.2, we extend our theory for the case when pathways interact with each other, yielding some additional discoveries for measuring the pathway stability and disease bifurcation points.

5 Experimental Results

In this section, we present the results of the numerical experiment and their biological interpretations. All experiments are conducted via one NVIDIA 4090 GPU with 24GB memory; the implementation details and additional discussions are included in Appendix C. Table 1 summarizes our findings.

5.1 Biology Discussion of the Findings

While KEGG-defined pathways (Section 3.1) provided a useful starting point, our analysis revealed unexpected pathway influences, emphasizing the need for recontextualization within the specific tissue environment of the retina. One key pathway cluster, node 79, was initially classified under "Caffeine Metabolism" in KEGG annotations. However, our analysis revealed that its constituent genes primarily function in metabolic detoxification and oxidative stress regulation (Iacopetta et al.,

Table 1: Summary of the findings, including top 5 SPs, SPs for disease stage changes, and top 5 stable and non-stable pathways from the results of pathway stabilities.

Tasks	Description of the Top 5 Pathways (Left to Right)				
Sensitive Pathways (SP)	Caffeine metabolism	Primary immunodeficiency	Small cell lung cancer	Long-term potentiation	Proteoglycans in cancer
SPs for Stage (0-1)	Biotin metabolism	Caffeine metabolism	Virion Human immunodeficiency virus	Virion Flavivirus	Virion Adenovirus
SPs for Stage (1-2)	alpha-Linolenic acid metabolism	Caffeine metabolism	Virion Human immunodeficiency virus	Virion Flavivirus	Axon guidance
SPs for Stage (2-3)	Caffeine metabolism	alpha-Linolenic acid metabolism	Virion Human immunodeficiency virus	Biotin metabolism	Virion Adenovirus
Stable Pathways	FoxO signaling pathway	Transcriptional misregulation in cancer	Peroxisome	PPAR signaling pathway	Biosynthesis of unsaturated fatty acids
Non-stable Pathways	Caffeine metabolism	Virion Human immunodeficiency virus	Virion Adenovirus	Phototransduction	Virion Flavivirus

2023; Chennamsetty et al., 2016; Chung et al., 1997; Álvarez-Lario and Macarrón-Vicente, 2010). These processes form a protective metabolic network that mitigates oxidative damage and toxic accumulation in retinal cells. Genes involved in xenobiotic metabolism and purine breakdown play a critical role in maintaining photoreceptor stability by reducing harmful by-products. Whether these pathways act as primary drivers of retinal degeneration or as secondary adaptive responses remains an open question. However, their high sensitivity to disease progression suggests that targeting metabolic detoxification pathways could be a novel therapeutic strategy.

In addition, another key driver of disease stage transition, from Stage 0 to Stage 1, was classified under KEGG as "Virion Human Immunodeficiency Virus." Rather than viral-related functions, this pathway consists of genes involved in immune regulation, chemokine-mediated signaling, and antigen recognition. Within the retina, an immunoprivileged tissue, this network plays a crucial role in maintaining immune homeostasis. Key molecular players include CCR5 (Crane et al., 2006) and CXCR4 (Zhu et al., 2024), which regulate T cell infiltration into the retina; the CD209 family, which monitors and responds to molecular threats; and CD4 (Caridade et al., 2013), which coordinates immune tolerance and response balance. If this immune surveillance network is overactivated, the resulting inflammatory response could drive retinal damage, accelerating degeneration. This suggests that modulating immune checkpoint pathways could be an effective approach to delaying retinal fibrosis. These findings underscore the importance of refining pathway annotations to reflect actual biological functions in the retina.

Furthermore, a key outcome of our study was the identification of a bifurcation point at Step 4 of our estimated trajectory, which represents an irreversible shift in the transcriptomic landscape. Notably, **all major pathways** exhibited their own bifurcation at this step by satisfying our proposed condition (2) (i.e., transition to a new steady state), suggesting a system-wide molecular transition that acts as a "point of no return". This finding suggests that the most effective therapeutic window is during the early stages of retinal degeneration, before this critical transition occurs. These findings are supported by clinical observations, further validating our computational approach.

6 Limitations of the Study and Future Directions

Despite our novel modeling methods and results, which serve as the strength of this study, our study has some limitations. Due to dataset constraints, our current model focused on pathway-level dynamics rather than individual gene-level interactions. Future work should refine this approach at the gene level, potentially identifying direct genetic targets for intervention. In addition, in Appendix C.2 we show a generalization of our model to measure the situation when pathways can interact with each other. Additionally, pathway annotations in KEGG are often too broad or non-specific for tissue- and disease-specific contexts, highlighting the need for retina-specific pathway ontology development.

Beyond macular degeneration, our approach could be applied to other complex diseases where molecular interactions drive progressive phenotypic changes. Future work should extend this methodology by conducting gene-level analysis within top-ranked pathways to identify direct therapeutic targets, refining pathway interpretations through advanced computational tools, and integrating multi-omics data for a more comprehensive disease model. Our study highlights the potential of computational

systems biology in uncovering actionable disease insights, paving the way for precision medicine approaches in retinal disease treatment.

From the machine learning perspective, we shall consider the interactions between both genes and their pathways. It would be ideal to construct a so-called hierarchical graph with two levels, with a gene graph as level one and a pathway graph as level two. We anticipate building a machine learning model to identify the relationship between the hierarchical graph and disease severity after the collection between different levels of graphs is built. Naturally, this modification will lead to a more complex but higher biological interpretable SDE and the formulations of the downstream tasks, we leave this promising direction for our future work.

REFERENCES

Bonifacio Álvarez-Lario and Jesús Macarrón-Vicente. Uric acid and evolution. *Rheumatology*, 49 (11):2010–2015, 2010.

Mitchell Black, Zhengchao Wan, Amir Nayyeri, and Yusu Wang. Understanding oversquashing in GNNs through the lens of effective resistance. In *International Conference on Machine Learning*, pages 2528–2547. PMLR, 2023.

Marta Caridade, Luis Graca, and Ruy M Ribeiro. Mechanisms underlying cd4+ treg immune regulation in the adult: from experiments to models. *Frontiers in immunology*, 4:378, 2013.

Wanze Chen, Orane Guillaume-Gentil, Pernille Yde Rainer, Christoph G Gäbelein, Wouter Saelens, Vincent Gardeux, Amanda Klaeger, Riccardo Dainese, Magda Zachara, Tomaso Zambelli, et al. Live-seq enables temporal transcriptomic recording of single cells. *Nature*, 608(7924):733–740, 2022.

Indumathi Chennamsetty, Michael Coronado, Kévin Contrepois, Mark P Keller, Ivan Carcamo-Orive, John Sandin, Giovanni Fajardo, Andrew J Whittle, Mohsen Fathzadeh, Michael Snyder, et al. Nat1 deficiency is associated with mitochondrial dysfunction and exercise intolerance in mice. *Cell reports*, 17(2):527–540, 2016.

Hae Young Chung, Bong Sook Baek, Sang Ho Song, Myoung Sook Kim, Jung Im Huh, Kyung Hee Shim, Kyu Won Kim, and Kyung Hee Lee. Xanthine dehydrogenase/xanthine oxidase and oxidative stress. *Age*, 20:127–140, 1997.

Isabel J Crane, Heping Xu, Carol Wallace, Ayyakkannu Manivannan, Matthias Mack, Janet Liversidge, Gabriel Marquez, Peter F Sharp, and John V Forrester. Involvement of ccr5 in the passage of th1-type cells across the blood-retina barrier in experimental autoimmune uveitis. *Journal of leukocyte biology*, 79(3):435–443, 2006.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems*, 29, 2016.

Michael B Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338, 2000.

Eliezer Flores-Garza, Rogelio Hernández-Pando, Ibrahim García-Zárate, Pablo Aguirre, and Elisa Domínguez-Hüttinger. Bifurcation analysis of a tuberculosis progression model for drug target identification. *Scientific Reports*, 13(1):17567, 2023.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/gilmer17a.html.

Piyush B Gupta, Christine M Fillmore, Guozhi Jiang, Sagi D Shapira, Kai Tao, Charlotte Kuperwasser, and Eric S Lander. Stochastic state transitions give rise to phenotypic equilibrium in populations of cancer cells. *Cell*, 146(4):633–644, 2011.

Gabriele Gut, Michelle D Tadmor, Dana Pe'er, Lucas Pelkmans, and Prisca Liberali. Trajectories of cell-cycle progression from fixed cell populations. *Nature methods*, 12(10):951–954, 2015.

Laleh Haghverdi, Maren Büttner, F Alexander Wolf, Florian Buettner, and Fabian J Theis. Diffusion pseudotime robustly reconstructs lineage branching. *Nature methods*, 13(10):845–848, 2016.

Domenico Iacopetta, Jessica Ceramella, Alessia Catalano, Elisabetta Scali, Domenica Scumaci, Michele Pellegrino, Stefano Aquaro, Carmela Saturnino, and Maria Stefania Sinicropi. Impact of cytochrome p450 enzymes on the phase i metabolism of drugs. *Applied Sciences*, 13(10):6045, 2023.

Weiwei Jiang and Jiayun Luo. Graph neural network for traffic forecasting: A survey. *Expert systems with applications*, 207:117921, 2022.

Rafail Khasminskii. *Stochastic stability of differential equations*. Springer, 2012.

Patrick Kidger, James Foster, Xuechen Li, and Terry J Lyons. Neural sdes as infinite-dimensional gans. In *International conference on machine learning*, pages 5453–5463. PMLR, 2021.

Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David Duvenaud. Scalable gradients for stochastic differential equations. *International Conference on Artificial Intelligence and Statistics*, 2020.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

Yu Liu and Mark R Chance. Pathway analyses and understanding disease associations. *Current genetic medicine reports*, 1:230–238, 2013.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Jacob D Moss, Jeremy L England, Israel Petach, and Pietro Lió. Pseudotime diffusion. *Machine Learning and the Physical Sciences Workshop, NeurIPS*, 2023.

Ignavier Ng, Shengyu Zhu, Zhitang Chen, and Zhuangyan Fang. A graph autoencoder approach to causal structure learning. *arXiv preprint arXiv:1911.07420*, 2019.

Peter K Panegyres. Stochasticity, entropy and neurodegeneration. *Brain Sciences*, 12(2):226, 2022.

Sandhya Prabhakaran, Elham Azizi, Ambrose Carr, and Dana Pe'er. Dirichlet process mixture model for correcting technical variation in single-cell gene expression data. In *International conference on machine learning*, pages 1070–1079. PMLR, 2016.

Xiaojie Qiu, Andrew Hill, Jonathan Packer, Dejun Lin, Yi-An Ma, and Cole Trapnell. Single-cell mrna quantification and differential analysis with census. *Nature methods*, 14(3):309–315, 2017.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

Kartik Sharma, Yeon-Chang Lee, Sivagami Nambi, Aditya Salian, Shlok Shah, Sang-Wook Kim, and Srijan Kumar. A survey of graph neural networks for social recommender systems. *ACM Computing Surveys*, 56(10):1–34, 2024.

Dai Shi, Andi Han, Lequan Lin, Yi Guo, and Junbin Gao. Exposition on over-squashing problem on GNNs: Current methods, benchmarks and challenges. *arXiv:2311.07073*, 2023.

Dai Shi, Andi Han, Lequan Lin, Yi Guo, Zhiyong Wang, and Junbin Gao. Design your own universe: A physics-informed agnostic method for enhancing graph neural networks. *International Journal of Machine Learning and Cybernetics*, 2024.

Cole Trapnell, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J Lennon, Kenneth J Livak, Tarjei S Mikkelsen, and John L Rinn. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature biotechnology*, 32(4):381–386, 2014.

Nafis Irtiza Tripto, Mohimenul Kabir, Md Shamsuzzoha Bayzid, and Atif Rahman. Evaluation of classification and forecasting methods on time series gene expression data. *Plos one*, 15(11): e0241686, 2020.

Yuyang Wang, Zijie Li, and Amir Barati Farimani. Graph neural networks for molecules. In *Machine Learning in Molecular Sciences*, pages 21–66. Springer, 2023.

Jiangyong Wei, Tianshou Zhou, Xinan Zhang, and Tianhai Tian. Dtflow: inference and visualization of single-cell pseudotime trajectory using diffusion propagation. *Genomics, Proteomics and Bioinformatics*, 19(2):306–318, 2021.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2020.

Chenxi Xie, Yueyuxiao Yang, Hao Yu, Qiushun He, Mingze Yuan, Bin Dong, Li Zhang, and Meng Yang. Rna velocity prediction via neural ordinary differential equation. *Iscience*, 27(4), 2024.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*, 2020.

Minkai Xu, Jiaqi Han, Aaron Lou, Jean Kossaifi, Arvind Ramanathan, Kamyar Azizzadenesheli, Jure Leskovec, Stefano Ermon, and Anima Anandkumar. Equivariant graph neural operator for modeling 3d dynamics. *arXiv preprint arXiv:2401.11037*, 2024.

Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *International Conference on Machine Learning*, pages 40–48. PMLR, 2016.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

Zehong Zhang, Lifan Chen, Feisheng Zhong, Dingyan Wang, Jiaxin Jiang, Sulin Zhang, Hualiang Jiang, Mingyue Zheng, and Xutong Li. Graph neural network approaches for drug-target interactions. *Current Opinion in Structural Biology*, 73:102327, 2022.

Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE transactions on intelligent transportation systems*, 21(9):3848–3858, 2019.

Lei Zhu, He Li, Xuening Peng, Zhaohuai Li, Sichen Zhao, Dongting Wu, Jialing Chen, Si Li, Renbing Jia, Zuohong Li, et al. Beneficial mechanisms of dimethyl fumarate in autoimmune uveitis: insights from single-cell rna sequencing. *Journal of Neuroinflammation*, 21(1):112, 2024.

A Related Works

Graph Neural Networks and Different Types of Graphs GNNs were originally proposed to resolve the challenge of data point dependencies via the traditional convolution neural networks, which, in general, treat every input data point independently of each other (Kipf and Welling, 2017; Defferrard et al., 2016). By considering the so-called adjacency information stored in the graph, GNNs propagate graph node features by aggregating its neighboring information (Wu et al., 2020). Such propagation paradigm has made GNNs one of the most successful tools in generating predictions via various types of graphs, such as citation networks (Wu et al., 2020), social networks (Sharma et al., 2024), molecules (as well as protein and ligands) (Zhang et al., 2022), traffic networks (Jiang and Luo, 2022), to name a few. These graphs vary from different levels of measurements of the data, e.g., scientific papers in citation networks (Yang et al., 2016), atoms, and amino acids in protein and ligand graphs (Wu et al., 2018; Gilmer et al., 2017) and nodes are connected with different types of attributes (e.g., citations and chemical bonds). In this work, we provide a **novel graph dataset** (known as JR5558, more details see Section 2) in which graphs are formed by the genetic pathways (as nodes) and pathway similarities (as edges), serving as the profiles of the experimental objects (e.g., mice). In addition, we also label these graphs with mice's lesion severity scores (as a continuous variable) obtained from fundus photographs of these mice, where severity was quantified by measuring subretinal lesion size.We train two different types of GNNs to capture the patterns between genetic pathways and mice lesion severity scores: one that does not consider temporal information, and another designed to account for temporal information from the estimated disease progression in the mice.

Pseudotime Analysis and Stochastic Differential Equations Pseudotime analysis (PA) was originally developed in single-cell transcriptomics to reconstruct cell differentiation trajectories from static snapshots of gene expression profiles (Trapnell et al., 2014; Haghverdi et al., 2016; Wei et al., 2021). Since time-resolved measurements of individual cells are often infeasible, pseudotime methods infer an intrinsic ordering (e.g., trajectories) of cells based on their transcriptional similarities, providing insights into dynamic biological processes. While it is a powerful tool widely applied in biology and medical science, its adoption in the machine learning community has only gained significant attention in recent years. For example, recent work has explored using diffusion models to infer pseudotime from single-cell transcriptomic data (Moss et al., 2023). In this work, we apply PA to the embeddings of graphs constructed from the genetic pathways of the mice and analyze the estimated trajectories via neural stochastic differential equations (Li et al., 2020) (see Section 4 for more details), which allow us to further exploit pathway stability and disease bifurcation points. This paradigm paves the path of incorporating advanced machine learning approaches to the exploration of the fundamental problems in complex biological systems.

B Proofs

Lemma 2 (Full Verison of Lemma 1)**.** *Let $\mathbf{x}(t) \in \mathbb{R}^d$ be a stochastic process governed by the Itô stochastic differential equation (SDE) given in Equation (7). Assuming there exist constants $C_1, C_2 > 0$ such that the ratio between the squared drift term and the trace of the diffusion term in Eq (7) remains bounded:*

$$C_1 \leq \frac{\|\psi_\theta(\mathbf{x}, t)\|^2}{\mathrm{Tr}(\xi_\phi(\mathbf{x}, t)\xi_\phi^\top(\mathbf{x}, t))} \leq C_2, \quad \forall \mathbf{x}, t. \tag{10}$$

*Further assuming the diffusion term $\xi_\phi(\mathbf{x}(t), t)$ is continuous and differentiable and has bounded partial derivatives, then the mean-square stability of the system satisfies if $\exists C > 0, \sup_{t\geq 0} \mathbb{E}\|\xi_\phi(\mathbf{x}(t), t)\|^2 > C$, i.e., the system exhibits **mean-square divergence**, thus **unstable** ($\mathbb{E}\|\mathbf{x}(t)\| \to \infty$). On the other hand, if $\lim_{t\to\infty} \mathbb{E}\|\xi_\phi(\mathbf{x}(t), t)\|^2 \to 0$, then the system is **mean-square asymptotically stable**, i.e., $\mathbb{E}\|\mathbf{x}(t)\|^2$ is bounded.*

Proof. Our conclusion is adopted from the work Khasminskii (2012) in which the stability of SDE is analyzed in a more solid manner. We start by discretizing the SDE with a small step Δt, we obtain:

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \psi_\theta(\mathbf{x}(t), t)\Delta t + \xi_\phi(\mathbf{x}(t), t)\Delta \mathbf{B}(t). \tag{11}$$

Taking the expectation of the squared difference, we get:

$$\mathbb{E}\|\mathbf{x}(t+\Delta t)-\mathbf{x}(t)\|^2 = \mathbb{E}\|\psi_\theta(\mathbf{x}(t),t)\Delta t + \xi_\phi(\mathbf{x}_t,t)\Delta\mathbf{B}(t)\|^2. \tag{12}$$

By independence of the drift and diffusion terms:

$$\mathbb{E}\|\mathbf{x}(t+\Delta t)-\mathbf{x}(t)\|^2 = \mathbb{E}\|\psi_\theta(\mathbf{x}(t),t)\Delta t\|^2 + \mathbb{E}\|\xi_\phi(\mathbf{x}(t),t)\Delta\mathbf{B}(t)\|^2. \tag{13}$$

Then, we can denote the following.

$$\mathbb{E}\|\psi_\theta(\mathbf{x}(t),t)\Delta t\|^2 = \|\psi_\theta(\mathbf{x}(t),t)\|^2\Delta t^2, \qquad \text{(Drift term contribution)}$$

and

$$\mathbb{E}\|\xi_\phi(\mathbf{x}(t),t)\Delta\mathbf{B}(t)\|^2 = \mathbb{E}\left[\mathrm{Tr}(\xi_\phi(\mathbf{x}(t),t)\xi_\phi(\mathbf{x}(t),t)^\top)\right]\Delta t. \qquad \text{(Diffusion term contribution)}$$

Thus, for sufficiently small Δt, the dominant term is:

$$\mathbb{E}\|\mathbf{x}(t+\Delta t)-\mathbf{x}(t)\|^2 \approx \mathrm{Tr}(\xi_\phi(\mathbf{x}(t),t)\xi_\phi(\mathbf{x}(t),t)^\top)\Delta t. \tag{14}$$

This is because for standard Wiener process, i.e., $\mathbf{B}(t)$ we have $\|\Delta\mathbf{B}(t)\|^2 = \Delta t$, therefore, the rate of the change of the diffusion time is proportional to Δt. One can also check the rate of change of the drifting term is proportional to Δt^2. This suggests once the steps of the disease evolution are dense enough, i.e., for any small step Δt there is an observation $\mathbf{x}(t+\Delta t)$. **One only needs to investigate the diffusion term of the SDE**. Specifically, from stochastic stability theory (Khasminskii, 2012) (Theorem 6.13), if $\sup_{t\geq 0}\mathbb{E}\|\xi_\phi(\mathbf{x}(t),t)\|^2 > C$, then $\mathbb{E}\|\mathbf{x}(t)\|^2 \to \infty$, implying instability. If $\lim_{t\to\infty}\mathbb{E}\|\xi_\phi(\mathbf{x}(t),t)\|^2 \to 0$, then $\mathbb{E}\|\mathbf{x}(t)\|^2$ remains bounded, implying stability, and this completes the graph. □

Remark 1. *Compared to the informal version of Lemma 1. The additional assumption is commonly observed in physical and biological systems, such as Langevin dynamics and gene regulatory networks (Elowitz and Leibler, 2000). Additionally, prior works in neural stochastic differential equations (Kidger et al., 2021) enforce similar regularization strategies to maintain numerical stability.*

C Experiment Details and Additional Results

C.1 Identification of SPs

In this section, we show details regarding the task of finding the SPs of the disease severity. In addition to the graph forming method, we illustrate in Section 3.1. We note that after the graph node features are constructed, the edge weight of the graph is computed via the distance of the node features from a Gaussian kernel. That is

$$\mathbf{W}_{i,j} = \mathrm{e}^{(-\|\mathbf{x}_i-\mathbf{x}_j\|^2)}, \tag{15}$$

in which we intrinsically let the variance of the Gaussian kernel as 1.

Disease Sensitivity Without GPA In terms of the model training of the first task mentioned in Section 3.2. We first did the normalization of the graph adjacency matrix that is $\widehat{\mathbf{A}} = \mathbf{D}^{\frac{1}{2}}(\mathbf{A}+\mathbf{I})\mathbf{D}^{\frac{1}{2}}$ by also adding the self-loop for each node. We split our dataset with 70% for training and 20% for testing. We deploy the GCN model together with global mean pooling with one additional MLP layer to fit the node features to the disease severity. That is

$$\mathbf{X}(\ell+1) = \mathrm{GCN}(\mathbf{X}(\ell),\widehat{\mathbf{A}}), \quad \widehat{\mathbf{y}} = \mathrm{MLP}(\mathrm{MeanPool}(\mathbf{X}(\ell+1)), \tag{16}$$

and we let ℓ be the number of layers, which is set as 2 in our modeling. For the hyperparameters, we let the hidden dimension of the GCN model as 64 and the dropout ratio as 0.5 with the learning rate as $1\mathrm{e}^{-3}$ and weight decay as $1\mathrm{e}^{-4}$. The model is trained with 10 runs in which every run owns 200 epochs. As we have illustrated in Section 3.2, the sensitivity of one pathway node s is then computed through $\mathrm{Sensitivity}(\mathbf{x}_s) = \sum_{i=1}^{m}\left\|\frac{\partial\widehat{\mathbf{y}}_i}{\partial\mathbf{x}_s}\right\|$ by directly calling the gradient from the loss backward process once the model is trained.

GPA and Temporal GCN For the GPA, after we obtained the latent representations of the graphs, i.e., $\widehat{\mathbf{z}}$. We did unsupervised clustering applying the Gaussian Mixture Model (GMM), followed by the KNN algorithm with $K = 10$ to sufficiently engage the connectivity between nodes. In terms of the MST and estimated trajectories, we fixed two starting points, which own zero severity, and ending points, which have the most server disease conditions, yielding four trajectories. To measure the temporal relationships between graphs, we deploy TGCN (Zhao et al., 2019) with specific feature propagation additional to Equation (6). That is

$$\mathbf{H}(t) = \mathbf{A}(t)\mathbf{X}(t)\mathbf{W}, \tag{17}$$

$$\mathbf{Q}(t) = \mathbf{A}(t-1)\mathbf{H}(t-1)\mathbf{W}_q + \mathbf{X}(t)\mathbf{P}_q, \tag{18}$$

$$\mathbf{R}(t) = \mathbf{A}(t-1)\mathbf{H}(t-1)\mathbf{W}_r + \mathbf{X}(t)\mathbf{P}_r, \tag{19}$$

$$\widetilde{\mathbf{H}}(t) = \tanh\left(\mathbf{A}(t-1)(\mathbf{R}(t) \odot \mathbf{H}(t-1)\mathbf{W}_h) + \mathbf{H}(t)\mathbf{P}_h\right), \tag{20}$$

$$\mathbf{H}(t) = (1 - \mathbf{Q}(t)) \odot \mathbf{H}(t-1) + \mathbf{Q}(t) \odot \widetilde{\mathbf{H}}(t), \tag{21}$$

$$\widehat{\mathbf{y}}(t) = \mathrm{MLP}(\mathbf{H}(t)), \tag{22}$$

where all the $\mathbf{W}$ and $\mathbf{P}$ are the learnable matrices. In our practice implementation, the data is split using torch static graph temporal signal split with 70% of training and the rest of the validation and testing. We let the hidden dimension be 96, the learning rate and weight decay be 0.01, and $1\mathrm{e}^{-3}$, respectively. Similarly, the model is trained 10 times with 200 epochs within each time.

C.2 More on Neural SDEs

Apart from the neural SDE models that we leveraged in Section 4, one may interested in the SDEs that measure the situation when pathway interaction matters. Recall that in Section 3.1, we developed pathway profile graphs for each object to sufficiently reflect its status, and this indicates us to **consider the interactions between pathways through the connectivities between them**. Accordingly, to measure the pathway interactions, we show the revised SDE as follows.

$$\frac{\partial \mathbf{x}(t)}{\partial t} = \big(\psi_{\boldsymbol{\theta}}(\mathbf{x}(t), t) + \sum_{j \sim i} \mathbf{A}_{i,j} \cdot \phi_\gamma(\mathbf{x}_j(t), t)\big)dt + \xi_{\boldsymbol{\phi}}(\mathbf{x}(t), t)d\mathbf{B}(t), \tag{23}$$

where $\sum_{j\sim i} \mathbf{A}_{i,j} \cdot \phi_\gamma(\mathbf{x}_j(t), t)$ is the standard graph spatial convolution between each node and its neighbors. In practice, $\phi_{\boldsymbol{\gamma}}(\mathbf{x}_j(t), t)$ simply denotes a reweighting paradigm based on the inductive bias of the graph and can be optionally omitted. The formulation in Equation (23) directly shows that graph convolution joins the predefined drifting terms to determine the feature changes from multiple facets and form the easiest case, the newly defined drifting term, i.e., $\psi_{\boldsymbol{\theta}}(\mathbf{x}(t), t) + \sum_{j\sim i} \mathbf{A}_{i,j} \cdot \phi_\gamma(\mathbf{x}_j(t), t)$, in practice, can be even implemented by a MLP plus one GNN. In fact, it is well-known that one can denote Equation (23) as

$$\frac{\partial \mathbf{x}(t)}{\partial t} = \big(\psi_{\boldsymbol{\theta}}(\mathbf{x}(t), t) + \sum_{j \sim i} \mathbf{A}_{i,j} \nabla \mathbf{x}_i\big)dt + \xi_{\boldsymbol{\phi}}(\mathbf{x}(t), t)d\mathbf{B}(t), \tag{24}$$

where we denote graph gradient $\nabla \mathbf{x}_i = \mathbf{x}_j - \mathbf{x}_i$. As in most cases, we don't have a similar assumption in Equation (10), thus to evaluate the pathway stability, one shall also count the contribution of $\nabla \mathbf{x}_i = \mathbf{x}_j - \mathbf{x}_i$. Accordingly, we can extend Equation (8) to obtain graph pathway stability (GPS) as

$$\mathrm{GPS}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} \|\xi_\phi(\mathbf{x}(t+1), t+1) - \xi_\phi(\mathbf{x}(t), t)\|^2 + \lambda \sum_{i\sim j} \mathbf{A}_{i,j}(\mathbf{x}_j(t) - \mathbf{x}_i(t))^2, \tag{25}$$

where $\sum_{i\sim j} \mathbf{A}_{i,j}(\mathbf{x}_j(t) - \mathbf{x}_i(t))^2$ is the well-known Dirichlet energy of the node features, which measures the total variation of the node features. In addition to the conclusion of Lemma equation 1, the inclusion of $\sum_{i\sim j} \mathbf{A}_{i,j}(\mathbf{x}_j(t) - \mathbf{x}_i(t))^2$ brings additional uncertainty to the system's stability. We then have the following conclusion.

Corollary 1 (GPS). *If $\mathbf{A}_{i,j} \geq 0, \forall i, j \in \mathcal{E}$, with the conditions obtained from Lemma 1, the system defined in Equation (7) is **mean-square asymptotically stable**; otherwise, it is **unstable**.*

Proof. The proof is straightforward since if $\mathbf{A}_{i,j} \geq 0$, then asymptotically, $\mathbf{AX}$ term will keep homogenizing every node feature accordingly to its neighbors, causing $\mathbf{x}_j = \mathbf{x}_i, t \to \infty$. Therefore,

the pathway stability can still be evaluated using the $\mathrm{PS}(\mathbf{x})$. However, if one has $\mathbf{A}_{i,j} \leq 0$, then the system will have Dirichlet energy explode, assuming in the continuous case, $\mathbf{x}$ is a function with infinite dimensionality. Therefore $\mathrm{GPS}(\mathbf{x}) \to \infty$, resulting an unstable system. □

Similarly, for the disease bifurcation point (BP), we can revise Definition 1 when we take the potential of node interactions into account. We provide the following definition.

Definition 2 (Graph-Interacted Point of No Return)**.** *Let $\mathbf{x}(t) \in \mathbb{R}^d$ be a stochastic process define in Equation (23). Define the system stability potential function, incorporating both self-dynamics and neighborhood influence, as*

$$J(\mathbf{x}_i, t) = -\int_0^t \Big[\psi_{\boldsymbol{\theta}}(\mathbf{x}_i, t') + \sum_{j \sim i} \mathbf{A}_{ij}(\mathbf{x}_j, t')\Big] dt'. \tag{26}$$

*We define the **point of no return** (bifurcation time) as a time t^* at which the pathway undergoes an irreversible transition in its dynamical behavior. Specifically, it is the first time at which at least one of the following conditions holds: (1): Loss of Stability Region: The pathway leaves a local stability region, $\nabla J(\mathbf{x}_i, t^*) \approx 0, \quad$ and $\quad \frac{d}{dt} J(\mathbf{x}_i, t^*) > 0$; (2): Transition to a New Steady State: The system irreversibly shifts to a new equilibrium: $\lim_{t\to\infty} \mathbb{E}\|\mathbf{x}_i(t) - \mathbf{x}_i(t^*)\|^2 > C, C > 0$; (3): Diffusion Variance Explodes: The fluctuation in the pathway dynamics grows beyond a threshold, $\sup_{t \geq t^*} \mathrm{Var}(\mathrm{PS}(\mathbf{x}_i, t)) > \delta, \delta > 0$; (4): Strong Neighborhood Influence: the pathway graph owns at least one negative eigenvalue, i.e., energy explode as illustrated in Corollary 1.*

The above definition extends the previous definition of the bifurcation point by counting the contribution of the interactions between pathways.