An Autoencoder and Generative Adversarial Networks Framework for Multi-Omics Data Analysis

by

IBRAHIM AL-HURANI

Graduate Program
in
Department of Electrical and Computer Engineering

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Ph.D.)

The Faculty of Graduate Studies
Lakehead University
Thunder Bay, Ontario, Canada

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Dr. Shadi Banitaan Professor College of Engineering & Science Department of Computer Science University of Detroit Mercy, Detroit, Michigan, USA
Internal Members:	Dr. M. Mazhar Rathore Assistant Professor Department of Computer Science Lakehead University, Canada
	Dr. Malek Alsmadi Adjunct Assistant Professor Department of Electrical Engineering Lakehead University, Canada
Supervisor:	Dr. Salama Ikki Professor Department of Electrical & Computer Engineering Lakehead University, Canada
Co-supervisor:	Dr. Abedalrhman Alkhateeb Assistant Professor Department of Computer Science Lakehead University, Canada

Declaration

I hereby declare that I am the sole author of this dissertation. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Dedication

To my father, **Ahmad Al-Hourani** *rahimahu Allah*, whose belief in me never wavered.

Though you are not here to witness this moment, your presence has been with me in every step of this journey. You gave without hesitation, supported without condition, and carried a vision for me that I could not always see for myself. There are no words that can fully capture my gratitude for your sacrifices, your trust, and your unwavering faith in me. This achievement is as much yours as it is mine.

To my mother, **Ruqaiya Al-Rawabdeh**, for her constant prayers, strength, and quiet resilience that sustained me through the most difficult times.

To my beloved wife, **Hana' Lafi**, whose patience, sacrifice, and steadfast support made this journey possible. You carried more than your share so that I could continue forward, and I am deeply grateful for everything you have done for our family.

To my sons, **Muhammad** and **Ali**, who are my greatest source of motivation and hope for the future.

To my brothers, **Muhammad Khair**, **Sulieman**, **Ali** *rahimahu Allah*, **Zeid**, **Abdul Rao'of**, and **Salem**, and to my sisters, **Wijdan**, **Asmaa'**, and **Isra'a**, whose support, encouragement, and belief in me have always been a foundation I could rely on.

Acknowledgment

First and foremost, all praise and thanks are due to Allah, whose guidance and mercy have made this journey possible. Without His will, I would not have reached this stage.

I would like to express my sincere gratitude to my supervisor, **Dr. Salama Ikki**. His patience and commitment to academic excellence have played a central role in shaping this work and my development as a researcher.

I am also deeply grateful to **Dr. Abedalrhman Alkhateeb** for his support, thoughtful advice, and encouragement. His guidance extended beyond technical input, offering perspective and motivation that were essential during challenging moments of this journey.

I would like to thank my committee members, **Dr. Shadi Banitaan**, **Dr. M. Mazhar Rathore** and **Dr. Malek Alsmadi**, for their time, careful evaluation, valuable insights and thoughtful review and constructive feedback that helped improve the quality of this dissertation.

My sincere thanks go to my colleagues and friends for their support and encouragement throughout this journey. I am especially grateful to my dear friends **Mohannad Al-Mousa**, **Hamzih Alsmadi**, **Lutfi Aniza**, **Ihab Abdel-Aal**, and **Farhan Yousef** for their companionship and support. I also extend my appreciation to everyone who, in one way or another, contributed to making this journey possible.

Abstract

The rapid advancement of high-throughput sequencing has generated vast multi-omics datasets that offer unprecedented insights into complex cancer phenotypes. However, effective integration of these modalities, including DNA methylation, gene expression, and copy number alterations, is frequently hindered by inherent high dimensionality, significant noise, and severe class imbalance, which collectively pose substantial challenges to traditional statistical and machine learning approaches. This dissertation addresses these challenges through a progressive and unified computational framework that evolves from interpretable linear modelling to advanced deep generative architectures for robust data integration and predictive modelling.

In the first stage, a linear framework was developed to identify menopause-related biomarkers in breast cancer. By utilizing a systematic preprocessing pipeline with MutSigCV, applying Synthetic Minority Oversampling TEchnique (SMOTE) to address class imbalance, and leveraging Principal Component Analysis (PCA) for dimensionality reduction, this research successfully identified and validated biologically significant markers including *RUNX1*, *PTEN*, *MAP3K1*, and *CDH1*. Interpretability was ensured via Shapley-value-based explainable AI (XGBoost), demonstrating the framework's ability to extract clinically relevant insights.

Recognizing the limitations of linear methods in capturing complex nonlinear relationships, the second stage introduced a deep learning-based framework integrating AutoEncoder (AE) with Generative Adversarial Network (GAN). This integration enabled the learning of a compact, nonlinear latent representation while simultaneously synthesizing realistic minority class samples to improve model generalization. The proposed AE–GAN framework achieved marked performance improvements, with classification accuracies of 88.82% for bladder cancer and 95.09% for breast cancer. Based on these findings, the final stage of this dissertation proposes a novel architecture-level integration of AE with Conditional Tabular Generative Adversarial Network (CTGAN). Unlike conventional approaches that generate synthetic data in the original feature space, this method trains CTGAN directly within a shared latent space, enabling the generation of high-fidelity synthetic samples that

preserve the intrinsic biological structure of the data.

Extensive evaluation demonstrates that the AE–CTGAN framework shows improved performance over earlier models, achieving near-perfect accuracies of 0.9929 for bladder cancer and 0.9748 for breast cancer. Furthermore, fidelity analysis reveals that latent space generation reduced the average Euclidean distance between real and synthetic samples by up to 84% compared to standard GANs. In general, this research contributes to a robust and scalable methodology for predicting cancer outcomes, supporting the development of personalized treatment strategies in precision medicine.

Future work will focus on adapting the framework to multi-class and longitudinal omics data, integrating attention-based or transformer architectures to improve interpretability, and validating the approach on prospective clinical cohorts to assess real-world generalizability. The proposed AE–CTGAN pipeline also holds promise beyond oncology, with potential applications in other multimodal biomedical domains such as neurodegenerative disease profiling, pharmacogenomics, and rare disease diagnosis, where high dimensionality and class imbalance are similarly pervasive. Ultimately, this dissertation establishes a foundation for robust, scalable, and fidelity-evaluated generative modelling in multi-omics research, contributing to the broader goal of precision medicine.

Table of Contents

List of Tables

List of Figures

Acronyms

ADASYN ADAptive SYNthetic Sampling. vi, 2, 35, 43, 44, 94, 99, 107

AE AutoEncoder. v–vii, 6, 8, 72, 81, 82, 84–86, 88–93, 96

AI Artificial Intelligence. 7, 28

AUROC Area Under the Receiver Operating Characteristic. 50, 51, 82

BLCA Bladder Urothelial Carcinoma. v, 6, 8, 12, 17, 18, 33, 68, 69, 81, 82, 84–86, 89, 90

BRCA Breast Cancer. v, vii, 6, 8, 12, 17, 18, 20, 33, 36, 68, 69, 81, 82, 84–87, 89, 90

CNA Copy Number Alteration. 1, 11, 12, 14, 15, 18, 23, 67, 68

CSV Comma-Separated Values. 95

CTGAN Conditional Tabular Generative Adversarial Network. v–vii, 3, 5–9, 25, 31–33, 68, 72, 75–78, 81–86, 88–93, 96, 103, 107

DDPM Denoising Diffusion Probabilistic Models. 92

DNA Deoxyribonucleic Acid. 1, 11, 12, 14, 15, 18, 22, 23, 26, 67, 68, 95

DR Dimensionality Reduction. 1–3

GAN Generative Adversarial Network. v, 2, 3, 6–9, 27, 28, 30–32, 72, 73, 78, 81–86, 89, 90, 93, 95, 96, 103, 107

miRNA microRNA. 68

ML Machine Learning. 27–29

MOVE Multi-Omics Variational AutoEncoder. 30

mRNA messenger Ribonucleic Acid. 11, 12, 14

MSE Mean Squared Error. 71

NGS Next-Generation Sequencing. 1, 12

PaCMAP Probabilistic Approximation Model with Controlled Mapping. 2

PCA Principal Component Analysis. v, 2, 35, 41, 42, 51, 52, 67, 88, 89

RBF Radial Basis Function. 35, 47, 48, 50–52

ReLU Rectified Linear Unit. 71, 79

SDV Synthetic Data Vault. 94

SHAP Shapley Additive Explanations. vi, 53–55, 63, 65

SMOTE Synthetic Minority Oversampling TEchnique. v, vi, 2, 35, 42–44, 88, 94, 98, 107

SVM Support Vector Machine. 35, 45, 47, 50–52

TCGA The Cancer Genome Atlas. vii, 17, 18, 36, 60, 62, 89, 91

TMB Tumor Mutational Burden. 17, 18, 22, 23, 68, 85, 89, 90, 93

VAE Variational Autoencoder. 29, 30

VAE-GAN Variational Autoencoder Generative Adversarial Network. 32

XAI Explainable Artificial Intelligence. 53, 91

Chapter 1

Introduction

1.1 Overview

Technological advances in Next-Generation Sequencing (NGS) have enabled the generation of diverse omics data types at unprecedented scale [1]. Multi-omics platforms are becoming increasingly accessible, facilitating comprehensive investigations into complex biological systems. Integrating heterogeneous omics data such as gene expression, Deoxyribonucleic Acid (DNA) methylation, and Copy Number Alteration (CNA) enables systems-level analysis of cancer mechanisms and outcome prediction.

Dimensionality Reduction (DR) techniques have been widely adopted for multi-omics integration [2, 3]. Nevertheless, the high dimensionality, heterogeneity, and limited sample sizes inherent in multi-omics datasets continue to pose significant analytical challenges.

1.2 Research Motivation

The motivation for this research stems from both clinical and methodological considerations in multi-omics cancer outcome prediction.

Clinically, minority outcome classes frequently correspond to high-risk or biologically distinct patient subgroups, including aggressive tumour phenotypes or rare molecular variants. In such contexts, overall accuracy alone is insufficient if minority cases are misclassified. Failure to identify these clinically critical subgroups may compromise treatment stratification and prognostic reliability. Consequently, improving minority-class recall and balanced predictive performance is a clinically meaningful objective.

Methodologically, multi-omics datasets are characterized by extreme dimensionality and heterogeneous feature distributions relative to the number of available samples. Linear DR methods such as PCA are limited in their ability to capture nonlinear biological interactions among genomic and epigenomic variables. Although nonlinear embedding approaches such as Probabilistic Approximation Model with Controlled Mapping (PaCMAP) [3] and t-SNE [4] have been explored, they remain constrained in representational capacity [5] or scalability [6].

Addressing class imbalance further complicates the modelling process. Traditional oversampling techniques such as SMOTE and ADASYN [7, 8] rely on local interpolation and may not preserve the complex distributional structure of tabular omics data. While GANs [9] provide a powerful generative framework, standard GAN architectures often exhibit instability when applied to structured tabular datasets [10].

Direct application of generative models to raw high-dimensional omics features can

further exacerbate instability and mode collapse. Learning compact latent representations prior to generative modelling offers a principled strategy for reducing dimensionality and structuring the data manifold. Autoencoders have demonstrated effectiveness in nonlinear representation learning [11, 12, 13], and variational approaches have shown promise in multi-omics integration [14]. However, integrating latent-space learning with conditional generative augmentation for imbalanced multi-omics outcome prediction remains insufficiently investigated.

These clinical and methodological factors collectively motivate the development of a unified framework that combines nonlinear latent representation learning with conditional generative modelling to enhance minority-class detection and predictive robustness.

1.3 Problem Statement

Multi-omics datasets exhibit high dimensionality, heterogeneous feature distributions, limited sample sizes, and pronounced class imbalance in clinically relevant outcomes. Linear DR methods are inadequate for modelling nonlinear biological relationships, while conventional oversampling techniques may distort the intrinsic structure of complex tabular omics data [15].

Although GAN models [9] provide a mechanism for synthetic data generation, standard architectures are not tailored for tabular data [10]. CTGAN [16] addresses structured tabular modelling through conditional mechanisms; however, its integration within autoencoder-derived latent representations for multi-omics cancer outcome prediction has not been systematically evaluated.

Accordingly, a robust framework is required that:

- Learns meaningful shared latent representations from heterogeneous multi-omics datasets. This dissertation develops a unified representation learning framework that extracts compact and biologically meaningful latent features from heterogeneous multi-omics data. Unlike linear techniques such as PCA, the proposed autoencoder captures nonlinear relationships across modalities and projects high-dimensional data into a low-dimensional latent space while preserving discriminative signals relevant to classification tasks such as menopausal status in BRCA and tumour mutational burden in BLCA. This shared latent representation reduces noise, redundancy, and modality-specific bias, resulting in a more stable and generalizable feature space for downstream learning.

- Performs minority-class augmentation within latent space. A key contribution of this work is to perform data augmentation within the learned latent space rather than in the original high-dimensional feature space.

- Compares conditional generative modelling with standard GAN approaches. This dissertation presents a systematic comparison between standard GANs and conditional GANs (CTGANs) for synthetic data generation in multi-omics classification tasks. While traditional GANs generate samples without class awareness, CTGAN incorporates conditional information to guide the generation process, enabling more accurate minority-class synthesis. The results demonstrate that CTGAN consistently produces higher-quality samples with improved distributional fidelity and class separability, as validated through quantitative metrics such as latent-space distance, highlighting the

importance of conditional generation in addressing class imbalance.

- Demonstrates generalizability across multiple cancer datasets. The proposed framework is evaluated across multiple cancer datasets, including BRCA and BLCA, which differ in biological characteristics, prediction objectives, and class imbalance severity. The consistent performance improvements observed across these datasets demonstrate that the autoencoder-based latent representation combined with CTGAN augmentation is not dataset-specific but rather a generalizable approach for multi-omics integration and classification. This cross-dataset validation strengthens the robustness and applicability of the proposed method to a wide range of biomedical prediction problems.

1.4 Research Gaps and Novelty

Existing approaches typically apply dimensionality reduction followed by classification, perform oversampling in raw feature space, or employ generative models independently of learned latent representations. A unified framework that integrates nonlinear latent representation learning with conditional generative augmentation for imbalanced multi-omics outcome prediction remains lacking.

To the best of our knowledge, no prior study has systematically evaluated CTGAN-based minority augmentation within autoencoder-derived shared latent spaces for multi-omics cancer outcome prediction, nor quantitatively assessed synthetic sample fidelity within the latent domain.

The novelty of this work lies in combining nonlinear representation learning and con-

ditional generative modelling within a single framework for robust imbalanced multi-omics classification.

1.5 Research Questions and Hypotheses

This study is guided by the following research questions and hypotheses.

RQ1: Can nonlinear latent representations learned via autoencoders effectively capture discriminative information from heterogeneous multi-omics datasets?

H1: Classifiers trained on autoencoder-derived latent representations achieve improved predictive performance compared to models trained on raw multi-omics features.

RQ2: Does minority-class augmentation in latent space improve classification performance under class imbalance?

H2: Latent-space augmentation improves recall and F1-score compared to non-augmented training.

RQ3: Does CTGAN-based augmentation outperform standard GAN augmentation?

H3: CTGAN yields higher-fidelity synthetic samples and improved F1-score and AUC compared to standard GAN.

RQ4: Does the proposed AE–CTGAN framework generalize across cancer types?

H4: The AE–CTGAN pipeline consistently achieves improved recall and F1-score across BRCA and BLCA datasets.

Performance was evaluated using accuracy, precision, recall, F1-score, AUC, and latent-space fidelity analysis.

1.6 Contributions

The contributions of this research lie at the intersection of multi-omics data integration, computational biology, generative Artificial Intelligence (AI), and biomarker discovery. Multi-omics data integration provides the foundational framework by combining heterogeneous molecular layers to enable a comprehensive, systems-level understanding of complex biological processes. In this context, computational biology provides the analytical methods needed to process and interpret high-dimensional biological data. Building upon these foundations, generative artificial intelligence techniques are employed to model, augment, and synthesize complex biological datasets, particularly in the presence of class imbalance. These integrated approaches ultimately support biomarker discovery by identifying meaningful molecular signatures that can enhance disease diagnosis, prognosis, and treatment stratification.

The main contributions of this dissertation are as follows:

- **Addressing class imbalance in multi-omics data:** This dissertation tackle the problem of *class imbalance*, which often biases machine learning models towards majority classes, by proposing a novel data augmentation pipeline, where the novelty is augmenting the tabular data based on the extracted latent space.

- **Utilizing an Autoencoder for latent space extraction:** An *autoencoder* was employed to extract a compressed latent representation of the multi-omics datasets, effectively reducing dimensionality while preserving meaningful information.

- **Comparative analysis of GAN and CTGAN for oversampling:** Synthetic samples

for the minority class were generated from the latent space using both *standard GAN* and *CTGAN* to compare their effectiveness.

- **First application in cancer outcome prediction:** To our knowledge, this is the *first study to apply CTGAN to latent spaces derived from autoencoders for cancer outcome prediction*, using its conditional sampling ability to generate high-quality minority-class samples.

- **Evaluation using downstream neural network classification:** The impact of generated samples by training a *neural network classifier* on the augmented data was assessed, using F1-score, precision, recall, and accuracy as performance metrics.

- **Demonstrating superior performance of AE–CTGAN pipeline:** Experimental results show that our proposed *AE–CTGAN approach significantly outperforms AE–GAN*, achieving higher F1-scores and recall, thus proving its effectiveness in improving minority class representation and classification performance.

- **Validation on two different datasets:** Our proposed technique was validated using *two distinct publicly available multi-omics datasets: BRCA (breast cancer) and BLCA (bladder cancer)*, to demonstrate the robustness and generalizability of our approach.

1.7 Dissertation Organization

The remainder of this dissertation is organized as follows:

- Chapter 2 provides background and preliminaries on multi-omics integration, cancer biology, dimensionality reduction techniques, and related work.

- Chapter 3 presents the linear framework, including the materials used, preprocessing, and the predictive modelling framework.

- Chapter 4 presents autoencoder-based latent representation learning with GAN and CTGAN augmentation strategies to handle class imbalance in multi-omics data analysis.

- Chapter 5 concludes the study and outlines future research directions.

1.8 List of Publications

1. Alghanim, Firas, Ibrahim Al-Hurani, Hazem Qattous, Abdullah Al-Refai, Osamah Batiha, Abedalrhman Alkhateeb, and Salama Ikki. "Machine learning model for multiomics biomarkers identification for menopause status in breast cancer." Algorithms 17, no. 1 (2023): 13.
   https://doi.org/10.3390/a17010013

2. Al-Hurani, I., Alkhateeb, A., Ikki, S.: An autoencoder and generative adversarial networks approach for multi-omics data imbalanced class handling and classification. arXiv preprint arXiv:2405.09756 (2024)
   https://doi.org/10.48550/arXiv.2405.09756

3. Al-Hurani I, Alkhateeb A and Ikki S.: An autoencoder and generative adversarial networks approach for multi-omics data imbalanced class handling and classification [version 1; not peer reviewed]. F1000Research 2025, 14:2 (poster).
   https://doi.org/10.7490/f1000research.1120071.1

4. Al-Hurani, Ibrahim, Sara H. ElFar, Abedalrhman Alkhateeb, and Salama Ikki. "AE-CTGAN: Autoencoder–Conditional Tabular GAN for Multi-Omics Imbalanced Class Handling and Cancer Outcome Prediction." Algorithms 19, no. 2 (2026): 95. https://doi.org/10.3390/a19020095

5. (Unrelated to thesis) Venkatraman, Santosh, Aniruddha Chakravarty, Nethan Shaik, Ibrahim Al-Hurani, and Abedalrhman Alkhateeb. "Machine Learning-Driven Prediction of Gleason Score 7 Prostate Cancer Patterns Using Multi-Omics Data." In 2025 International Conference on New Trends in Computing Sciences (ICTCS), pp. 353-360. IEEE, 2025. https://doi.org/10.1109/ICTCS65341.2025.10989373

Chapter 2

Background and Preliminaries

This chapter introduces the following preliminaries to help understand the outcomes of the proposed methodologies.

2.1 Multi-omics

Multi-omics refers to an analytical strategy that integrates multiple layers of biological data, including genomics, transcriptomics, proteomics, epigenomics, and metabolomics, to study biological systems in a unified manner. By jointly analyzing complementary data modalities such as DNA, CNA and messenger Ribonucleic Acid (mRNA) expression profiles, multi-omics approaches provide deeper insight into the molecular complexity of biological processes and disease states compared to analyses based on a single omics layer. However, integrating multi-omics data presents substantial challenges, including high-dimensional feature spaces, heterogeneity across data types, and increased computational demands. Addressing these challenges through robust computational and machine learning

approaches is essential for identifying meaningful biomarkers, stratifying disease subtypes, and uncovering potential therapeutic targets, particularly in oncology research [17].

Advances in high-throughput experimental technologies, including NGS and microarray-based platforms, have led to rapid growth in genomic and transcriptomic datasets. These technologies enable the simultaneous characterization of DNA-level alterations, such as CNA variations, alongside mRNA expression patterns within a unified experimental framework [18]. Consequently, the integration of genomic and transcriptomic data has become increasingly important for achieving systems-level biological insights and supporting precision medicine initiatives [19].

In this study, the multi-omics data consist of DNA, CNA profiles representing genomic alterations and mRNA expression data capturing transcriptomic activity. These datasets were obtained from The Cancer Genome Atlas (TCGA) and focus on breast cancer (BRCA) and bladder cancer (BLCA) cohorts.

2.1.1 Clinical and Research Benefits of Multi-omics Integration

Integrative multi-omics approaches improve diagnostic accuracy by capturing complementary molecular alterations across genomic, transcriptomic, epigenomic, and proteomic layers. The integration of these diverse molecular signatures enables the identification of distinct cancer subtypes and supports earlier and more precise disease detection compared with single-omics analysis [20]. By revealing tumour-specific molecular patterns, multi-omics strategies enhance classification systems and contribute to more reliable molecular diagnosis.

Multi-omics integration facilitates the discovery of robust and biologically meaningful

biomarkers that are strongly associated with clinical outcomes, including overall survival, disease recurrence, and therapeutic response. These biomarkers support improved risk stratification and more accurate prognosis estimation at the individual patient level [21].

For example, Chaudhary et al. demonstrated that deep learning-based integration of genomic, transcriptomic, and epigenomic features significantly improved survival prediction in liver cancer, highlighting the clinical value of multi-omics-driven biomarker discovery [22]. Such integrative approaches enable more reliable prediction models by capturing complex molecular interactions underlying tumour progression.

The comprehensive characterization of tumours through multi-layered molecular profiling supports the identification of actionable therapeutic targets and improves the prediction of treatment response. This integrative framework underpins precision oncology, where therapeutic decisions are guided by individual molecular profiles rather than generalized clinical characteristics [21, 20].

Multi-omics integration enables personalized treatment by identifying biomarkers associated with drug sensitivity, resistance mechanisms, and pathway-specific vulnerabilities. These insights facilitate targeted therapeutic interventions and contribute to improved clinical outcomes for cancer patients [23].

Overall, multi-omics analysis plays a critical role in advancing personalized medicine by enhancing diagnostic precision, improving prognostic assessment, and enabling more effective, individualized treatment strategies.

2.2 Molecular Data Modalities (Omics)

CNA, mRNA expression, DNA methylation, and gene expression collectively represent key molecular layers for characterizing cancer biology. CNAs arise from structural genomic changes that lead to gains or losses of DNA segments and can substantially influence gene dosage, expression levels, and cellular function. These alterations play critical roles in cancer initiation, progression, and resistance to therapy [24]. At the transcriptomic level, mRNA expression profiling provides insight into gene regulatory networks and the functional state of cells, with aberrant expression patterns closely associated with tumour progression, metastatic potential, and treatment response, particularly in breast cancer.

In parallel, DNA methylation serves as a major epigenetic regulatory mechanism that modulates gene expression without altering the underlying DNA sequence. Dysregulated methylation patterns contribute to tumourigenesis by silencing tumour suppressor genes or activating oncogenes, thereby affecting key cellular processes such as proliferation, apoptosis, and differentiation [25, 26]. Gene expression profiles further enable the identification of molecular cancer subtypes, which is essential for predicting patient prognosis and determining appropriate therapeutic strategies. Consequently, modern cancer research increasingly relies on the integrated analysis of gene expression data, alongside other molecular alterations, to achieve a more comprehensive and precise understanding of tumour [27].

2.3 Multi-Omics Data Integration

The integration of multi-omics data has become a critical component in modern cancer research, as complex diseases such as cancer cannot be fully characterized by a single molecular modality. For instance, gene expression profiles provide insights into transcriptional activity, while DNA methylation reflects epigenetic regulation, and CNA captures structural genomic variations. Combining these heterogeneous data sources enables a more comprehensive understanding of tumour behaviour and disease progression [28].

Figure 2.1 illustrates the integration of complementary omics layers, including genomics, transcriptomics, epigenomics, proteomics, and metabolomics, to provide a systems-level view of cancer biology. This integrative perspective motivates the use of machine learning methods to extract informative latent representations and improve downstream predictive performance.

Multi-omics integration strategies can be broadly categorized into three main approaches: early integration, intermediate integration, and late integration. Early integration, also known as feature-level integration, involves concatenating features from multiple omics datasets into a single high-dimensional matrix. While straightforward, this approach often exacerbates the curse of dimensionality and may introduce redundancy. Late integration, or decision-level integration, combines predictions from models trained separately on each omics type. Although this method preserves modality-specific information, it may fail to capture cross-omics interactions [29].

In contrast, intermediate integration focuses on learning a shared representation across multiple data modalities. This approach is particularly suitable for capturing complex

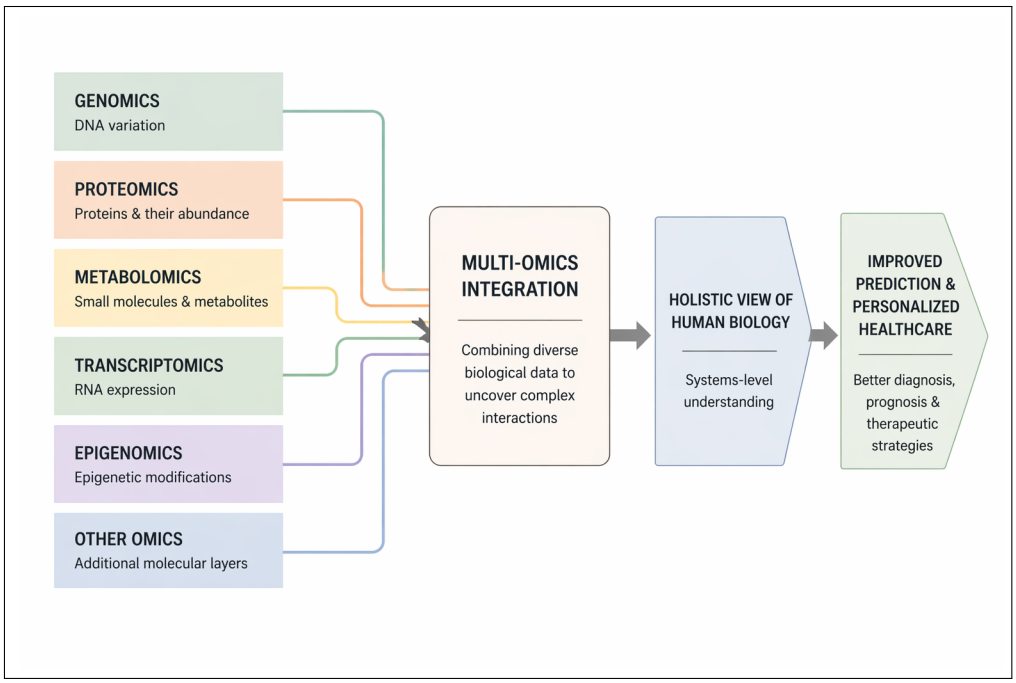


Figure 2.1: Schematic representation of multi-omics integration, showing how complementary molecular layers, including genomics, transcriptomics, epigenomics, proteomics, and metabolomics, can be combined to support systems-level biological analysis, computational prediction, and deeper insight into cancer biology.

nonlinear relationships among features and is adopted in this work. Specifically, multi-omics features are first combined and then transformed into a lower-dimensional latent representation, which serves as the basis for downstream analysis. This strategy enables the model to learn meaningful feature interactions while mitigating the challenges of high dimensionality [29].

Despite its advantages, multi-omics integration presents several challenges, including differences in data distributions, measurement scales, and noise levels across omics types. These challenges necessitate robust preprocessing and representation learning techniques to ensure effective integration and meaningful downstream analysis.

2.4 Cancer Context and Study Cohorts

Breast cancer BRCA and bladder cancer BLCA are among the most extensively studied malignancies in genomic research, particularly within large-scale initiatives such as TCGA [30]. These datasets provide comprehensive multi-omics profiles that enable the investigation of complex biological mechanisms and the development of predictive models.

In the context of BRCA, tumour heterogeneity is influenced by a combination of genetic, epigenetic, and hormonal factors. One clinically relevant aspect of breast cancer is menopausal status, which significantly affects disease progression, treatment response, and patient outcomes. The classification of pre-menopausal and post-menopausal breast cancer cases presents a challenging task due to subtle molecular differences and overlapping feature distributions across classes.

Bladder cancer BLCA, on the other hand, is often characterized by variations in Tumor Mutational Burden (TMB), which represents the number of mutations per coding region of the tumour genome. TMB has become an important biomarker for predicting the response to immunotherapy, particularly in the context of immune checkpoint inhibitors [31]. However, classifying tumours into high and low TMB categories is challenging due to data imbalance and complex genomic patterns.

Both data sets BRCA and BLCA exhibit key characteristics that make them suitable for the evaluation of advanced machine learning techniques. First, they are inherently high-dimensional, often containing tens of thousands of features across multiple omics layers. Second, they exhibit significant biological variability and noise, which complicates the generalization of the model. Third, they frequently suffer from class imbalance, where one

class is underrepresented, leading to biased model performance if not properly addressed.

In this work, multi-omics data from TCGA is utilized for both BRCA and BLCA cohorts. The datasets include gene expression, DNA methylation, and CNA features, which are integrated into a unified representation. The classification tasks focus on predicting menopausal status in BRCA and TMB status in BLCA, providing two distinct yet complementary case studies for evaluating the proposed framework.

The combination of these datasets enables the assessment of the proposed methodology across different biological contexts, demonstrating its robustness and generalizability in handling heterogeneous and imbalanced multi-omics data. Cancer is a complex and heterogeneous disease influenced by genetic, epigenetic, and environmental factors. Single-omic approaches may not fully capture this complexity due to the lack of omics interactions, highlighting the need for multi-omics integration. By analyzing interactions and correlations across multiple omics layers, researchers can identify robust molecular drivers of cancer, thereby facilitating more accurate and comprehensive models of biology [32].

2.4.1 Cancer

Cancer is a multi-factorial and heterogeneous group of diseases characterized by the progressive accumulation of genetic mutations and epigenetic alterations that disrupt normal cellular homeostasis. These molecular abnormalities promote uncontrolled cellular proliferation, resistance to apoptosis, sustained angiogenesis, and tumour progression [33]. The cancer cells grow abnormally as seen in Figure 2.2 [34].

The initiation and progression of cancer are driven by complex interactions among intrinsic genetic susceptibility, environmental exposures, lifestyle factors, and dysregulation

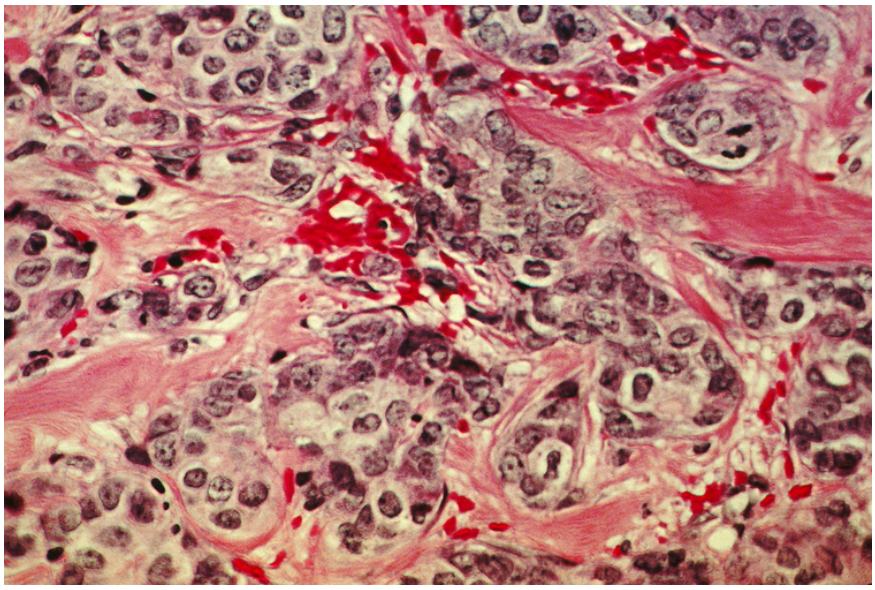


Figure 2.2: Histopathological image of human breast cancer tissue, illustrating malignant cells (dark purple) embedded within surrounding connective tissue (pink). This visual highlights the structural heterogeneity and complex cellular organization characteristic of tumour microenvironments, which pose significant challenges for computational analysis and modeling. Image credit: Cecil Fox, National Cancer Institute (NIH)

of multiple interconnected molecular pathways [35]. Due to this biological complexity, single-layer molecular analysis is often insufficient to fully characterize tumour behaviour.

Consequently, integrative multi-omics approaches combining genomic, transcriptomic, epigenomic, and proteomic data have become essential for achieving a systems-level understanding of cancer biology [36]. Such integrative analyses enable the identification of robust biomarkers, improved disease subtyping, and discovery of clinically actionable therapeutic targets, thereby enhancing cancer diagnosis, prognosis, and precision treatment strategies [37].

2.4.2 Breast Cancer

Breast cancer is a malignant disease originating from epithelial cells of the breast, most commonly arising in the ducts or lobules. It is the most frequently diagnosed cancer

among women worldwide and remains a leading cause of cancer-related mortality [38]. The etiology of breast cancer is multifactorial, involving genetic susceptibility, hormonal and reproductive factors, environmental influences, and lifestyle behaviours. Pathogenic variants in high-penetrance genes such as *BRCA1* and *BRCA2* substantially increase lifetime risk [39]. Additional risk factors include prolonged estrogen exposure, obesity, alcohol consumption, and reproductive history [40].

Breast cancer exhibits marked molecular and clinical heterogeneity, which has driven the adoption of integrative multi-omics approaches to improve subtype classification, prognostic assessment, and therapeutic stratification [41].

2.4.3 Menopause and Breast Cancer

Menopause represents the permanent cessation of ovarian function and is clinically defined as the absence of menstruation for 12 consecutive months. Women are broadly categorized as pre-menopausal or post-menopausal based on ovarian hormonal activity. This physiological transition has important implications not only for reproductive health but also for breast cancer biology and treatment response [42].

Menopausal status significantly influences tumour molecular characteristics and is commonly incorporated as a key clinical variable in breast cancer research and clinical studies [43]. Integrating menopausal status within multi-omics frameworks enhances the characterization of tumour heterogeneity and facilitates the identification of subtype-specific biomarkers and therapeutic targets.

Breast Cancer in pre-menopausal Women

pre-menopausal women typically exhibit higher circulating levels of estrogen and progesterone, which can promote hormone-driven tumourigenesis [42]. Breast cancers diagnosed in younger, pre-menopausal women often display more aggressive clinical and molecular features. These tumours are more frequently triple-negative-lacking expression of estrogen receptor (ER), progesterone receptor (PR), and HER2-which is associated with poorer prognosis and limited targeted treatment options [44].

Furthermore, pre-menopausal patients are more likely to present with higher tumour grade and increased lymph node involvement at diagnosis, reflecting more aggressive disease biology.

Breast Cancer in Post-menopausal Women

Post-menopausal women experience reduced ovarian estrogen production, resulting in distinct tumour biology and therapeutic responses [42]. Breast cancers in this group are more commonly hormone receptor-positive and therefore responsive to endocrine therapies such as aromatase inhibitors and selective estrogen receptor modulators (e.g., tamoxifen) [45].

Understanding differences between pre-menopausal and post-menopausal breast cancer is essential for accurate risk stratification, treatment selection, and integrative multi-omics analysis.

2.4.4 Bladder Cancer

Bladder cancer arises from the epithelial lining of the urinary bladder, with urothelial carcinoma representing the predominant histological subtype. It is among the most common malignancies of the urinary tract and is strongly associated with environmental and lifestyle risk factors, particularly tobacco smoking, occupational exposure to aromatic amines and industrial chemicals, and chronic bladder irritation or inflammation [46].

Bladder cancer exhibits considerable molecular heterogeneity and variable clinical behaviour, ranging from non-muscle-invasive disease to highly aggressive muscle-invasive and metastatic forms. Treatment strategies are determined primarily by tumour stage and grade and typically include surgical resection, intravesical therapy, systemic chemotherapy, immunotherapy, and radiation as part of multimodal management [47].

Advances in molecular profiling and multi-omics integration have improved understanding of bladder cancer pathogenesis and have facilitated the identification of biomarkers for prognosis and therapeutic response [48].

2.4.5 Tumour Mutational Burden

TMB is a quantitative biomarker defined as the total number of somatic coding mutations-including base substitutions and small insertions/deletions-per megabase of tumour DNA. TMB serves as a surrogate measure of tumour neoantigen load and reflects the likelihood of immune recognition. Bladder cancer is characterized by relatively high genomic instability, and TMB has emerged as an important biomarker for predicting prognosis and response to immunotherapy [49].

In this study, TMB is used to stratify bladder cancer samples into low- and high-mutation groups to investigate differential molecular patterns across multi-omics layers, including CNA, gene expression, and DNA methylation.

High TMB

High TMB tumours exhibit elevated numbers of somatic mutations and increased genomic instability, leading to the generation of a large repertoire of neoantigens. These neoantigens enhance tumour immunogenicity and are frequently associated with increased infiltration of cytotoxic $CD8^+$ T cells and heightened immune activity within the tumour microenvironment [50].

Clinically, high TMB is strongly associated with improved response to immune checkpoint inhibitors, including therapies targeting the PD-1/PD-L1 axis [49]. Clinical trials have demonstrated that patients with high TMB in bladder cancer often experience improved survival and higher response rates following immunotherapy [51].

In this work, high TMB samples are analyzed to identify multi-omics biomarkers linked to immune activation, tumour progression, and therapeutic responsiveness.

Low TMB

Low TMB tumours are characterized by fewer somatic mutations and reduced neoantigen burden, resulting in lower immune recognition. These tumours frequently display an immune-cold phenotype, marked by limited T-cell infiltration, decreased antigen presentation, and a more suppressive tumour microenvironment. Consequently, patients with low TMB bladder cancer are less likely to respond to immune checkpoint blockade [31].

The inclusion of low TMB samples in this study enables the identification of transcriptomic and epigenetic signatures associated with immune exclusion and tumour progression. Understanding these molecular differences may support the development of alternative therapeutic strategies for patients who are less responsive to current immunotherapy approaches.

2.5 Challenges in Multi-Omics Data Integration

Despite its potential to provide comprehensive molecular insight, multi-omics data integration in cancer research presents substantial technical, statistical, and biological challenges. These challenges arise from the intrinsic complexity of high-throughput molecular data and the heterogeneity of tumour biology.

2.5.1 High Dimensionality and Limited Sample Size

Multi-omics datasets typically contain tens of thousands of molecular features across genomic, transcriptomic, epigenomic, and proteomic layers, while the number of available patient samples is often relatively small. This scenario of "large p, small n" increases the risk of overfitting, reduces the generalizability of the model, and substantially increases computational complexity [52]. Dimensionality reduction, feature selection, and latent representation learning are therefore essential components of multi-omics modeling pipelines. Although classical dimensionality reduction techniques such as PCA, LDA, and feature selection methods have been widely adopted in multi-omics studies, they exhibit inherent limitations when applied to high-dimensional biological data. These approaches are predominantly linear and may fail to capture complex nonlinear relationships and interactions

that exist across different omics layers. Furthermore, feature selection techniques, although useful for reducing dimensionality, often depend on predefined criteria and may overlook subtle but biologically meaningful patterns embedded in the data.

In addition to these limitations, multi-omics datasets pose further challenges due to their heterogeneity, sparsity, and differing data distributions across omics platforms. As highlighted by Reel et al., the integration of such diverse data sources often involves fragmented pipelines in which dimensionality reduction, data integration, and model learning are treated as separate stages [53]. This separation can limit machine learning models' ability to learn unified representations that effectively capture cross-omics dependencies and underlying biological mechanisms.

To address these challenges, there is an increasing need for representation learning approaches capable of modelling complex, nonlinear structures within multi-omics data. In this work, an autoencoder-based framework is employed to learn a shared latent feature space that integrates multiple omics layers into a compact and informative representation. This learned latent space not only mitigates the curse of dimensionality but also preserves meaningful biological relationships, providing a robust foundation for downstream analysis. Moreover, by operating within this learned representation space, the proposed approach facilitates more effective synthetic data generation with CTGAN, thereby improving handling of class imbalance while maintaining the structural fidelity of the original multi-omics data [53].

2.5.2 Data Heterogeneity and Scale Differences

Different omics modalities exhibit diverse data structures, statistical distributions, and measurement scales. For example, gene expression data are typically continuous, mutation profiles are binary or categorical, and DNA methylation is represented as beta values. These heterogeneous formats complicate integration and require careful normalization and transformation to prevent a single omics layer from dominating due to scale differences [52]. The absence of universally standardized preprocessing and integration frameworks remains a significant barrier to reproducible multi-omics analysis.

2.5.3 Integration Complexity and Technical Variability

Integrating multi-omics data requires combining measurements generated from different experimental platforms and preprocessing pipelines. Differences in sequencing depth, batch effects, missing values, and platform-specific noise introduce additional technical variability that can obscure true biological signals [54]. Robust normalization, batch-effect correction, and data harmonization strategies are therefore critical for reliable multi-omics integration.

2.5.4 Class Imbalance and Model Bias

Class imbalance is a common issue in cancer datasets, where certain tumour subtypes or clinical groups are significantly underrepresented compared to others. Machine learning models trained on imbalanced data tend to favour the majority class, leading to biased predictions and reduced performance for minority groups [55].

To address this problem, several strategies have been proposed, including oversampling the minority class, undersampling the majority class, generating synthetic data, and cost-sensitive learning [56]. Among these approaches, synthetic data generation using generative models-particularly GAN has emerged as an effective technique for mitigating class imbalance by producing realistic artificial samples that enhance model generalization and improve classification performance.

2.5.5 Interpretability and Biological Relevance

Although advanced machine learning and deep learning approaches have improved predictive performance in multi-omics analysis, model interpretability remains a critical challenge. Translating computational findings into biologically meaningful insights, such as identifying causal molecular mechanisms or clinically actionable biomarkers, requires careful validation and integration with domain knowledge [57].

Overall, multi-omics integration faces challenges, including high dimensionality, heterogeneous and noisy data, class imbalance, and limited interpretability. Ongoing methodological advances, including representation learning, robust feature integration, and scalable computational frameworks, continue to improve the reliability and biological relevance of multi-omics cancer studies [58, 59].

2.6 Machine Learning Foundations for Multi-omics

Machine Learning (ML) is a core subfield of artificial intelligence that enables computational systems to learn patterns from data and make predictions or decisions without explicit rule-

based programming [60]. As illustrated in Figure 2.3, ML resides within the broader domain of AI, which itself is a subdomain of Computer Science. The figure highlights the major learning paradigms within ML, including supervised, semi-supervised, unsupervised, and reinforcement learning, while also illustrating the role of deep learning as a powerful methodological framework capable of modelling complex nonlinear relationships.

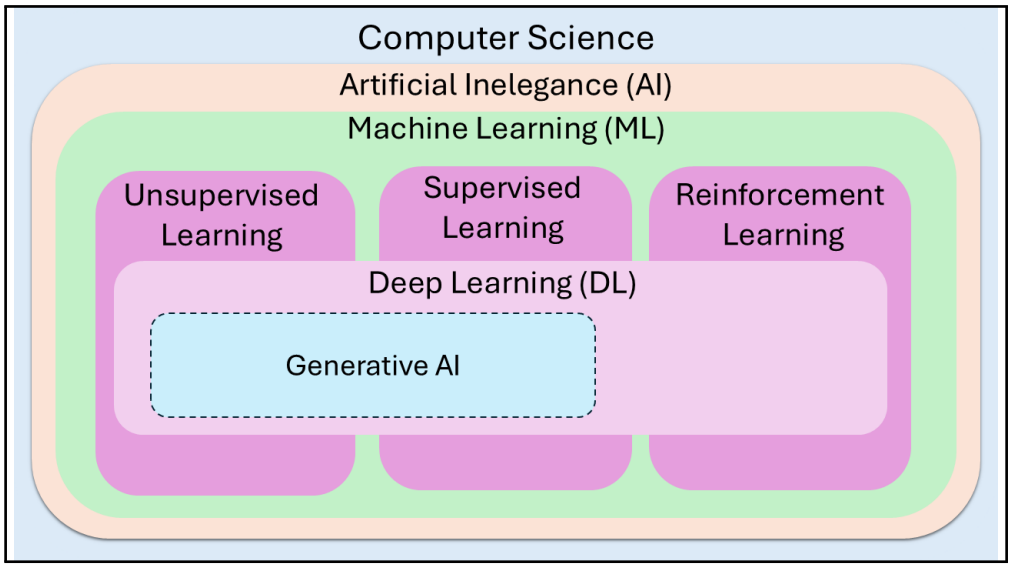


Figure 2.3: Conceptual overview of machine learning paradigms, including supervised, unsupervised, and reinforcement learning, as well as the role of deep learning and generative models within the broader machine learning framework.

Building upon this conceptual hierarchy, it is important to emphasize that deep learning does not constitute a separate learning paradigm, but rather a set of techniques that can be applied across different paradigms. Through the use of multi-layer neural networks, deep learning enables hierarchical feature extraction and representation learning, which is particularly effective for high-dimensional and heterogeneous data such as multi-omics datasets. In addition, generative artificial intelligence represents a specialized class of machine learning methods that focus on learning the underlying data distribution in order to generate new, realistic samples. Notably, many generative models, including GAN and

Variational Autoencoder (VAE), are implemented using deep learning architectures, thereby creating a natural overlap between deep learning and generative modelling.

2.6.1 Learning Paradigms

In this study, both unsupervised and supervised learning approaches are employed. Specifically, an unsupervised dimensionality reduction technique is utilized to capture the intrinsic structure of high-dimensional multi-omics data and to generate compact latent representations. These representations are subsequently incorporated into a supervised learning framework for classification, enabling accurate prediction of clinically relevant groups. The integration of unsupervised feature learning with supervised classification improves model robustness, reduces noise, and enhances the discovery of biologically meaningful patterns within integrated multi-omics data.

In bioinformatics and computational oncology, ML techniques play a critical role in analyzing high-dimensional multi-omics data. These methods facilitate biomarker discovery, molecular subtype classification, disease diagnosis, prognosis prediction, and the development of personalized therapeutic strategies [61, 62, 63]. By integrating heterogeneous biological data and capturing hidden molecular patterns, ML enables more accurate modeling of complex biological systems and supports precision medicine initiatives.

Furthermore, the combination of representation learning and generative modeling provides a powerful framework for addressing key challenges in multi-omics analysis, including high dimensionality and class imbalance. In this work, unsupervised deep learning is employed to learn a structured latent feature space, which serves as the basis for subsequent generative modeling using CTGAN to synthesize realistic samples for minority classes.

This is followed by supervised learning for classification, forming a unified pipeline that leverages multiple machine learning paradigms to improve predictive performance while preserving the structural fidelity of the original data.

2.7 Related Work

In multi-omics research, GANs are used to generate synthetic datasets that can enhance predictive models. For example, OmicsGAN integrates multiple omics data and their interaction networks to generate synthetic data with better predictive signals [64]. However, traditional GANs can face issues such as mode collapse and training instability [65], which have been addressed by models like Wasserstein GANs. These incorporate the Wasserstein distance to enhance training stability [66]. Still, Wasserstein GANs were not specifically designed for tabular data [16]. The recent models in the literature are highlighted in Table 2.1.

VAEs are another type of generative model that encode input data into a latent space and then decode it back to the original space. This process allows VAEs to learn the underlying distribution of the data and generate new samples. In multi-omics research, VAEs are used for data integration and dimensionality reduction [67]. For instance, the Multi-Omics Variational AutoEncoder (MOVE) framework integrates various omics data and clinical variables to identify cross-modal associations and improve data resolution [68]. VAEs are particularly useful for handling high-dimensional data and uncovering complex biological patterns [69, 70]. These generative models offer powerful tools to overcome the challenges of complex, sparse, and high-dimensional multi-omics data, enabling new in-

sights and advances in biological research [71]. Apellániz et al. propose a synthetic tabular

Table 2.1: Comparison of generative models for cancer predictio

Model	Year	Author	Key Contribution	Application	Performance
OmicsGAN	2022	Ahmed et al. [64]	Conditional GAN generating class-specific samples	Rare cancer subtype prediction	Improved recall and F1-score compared to SMOTE and ADASYN
CTGAN	2024	Kim et al. [72]	GTGAN applied to gene expression and survival data	12 cancer types (incl. BRCA)	Better prediction under imbalance
LASSO-MOGAT	2024	Alharbi et al. [73]	Graph-attention for multi-omics integration	Cancer outcome prediction	Handles high dimensionality and complexity
VAE-GAN	2021	Nußberger et al. [74]	Hybrid variational autoencoder and GAN model	SNP data	Effective with binary omics input
Precious2GPT	2024	Sidorenko et al. [75]	Combines Diffusion Models and Transformers	Class imbalance scenarios	Higher quality than GANs and VAEs

data generation framework for low-data medical scenarios that leverages transfer learning and meta-learning to introduce an artificial inductive bias, demonstrating improved distributional similarity and robustness across cancer-related datasets. However, their approach relies heavily on raw feature-space generation and validation metrics that lack sensitivity to true data fidelity, particularly in low-sample settings, whereas this work addresses these limitations by performing generative modelling in an autoencoder-derived latent space and explicitly evaluating fidelity through CTGAN-based augmentation and quantitative similarity measures, enabling more reliable preservation of underlying multi-omics data structures [76].

Unlike traditional GANs, which model the marginal distribution $p(x)$ without explicitly accounting for feature dependencies or class structure, CTGAN [16] introduces conditional generation tailored for tabular data. By conditioning both the generator and discriminator

on discrete variables, CTGAN models the conditional distribution *p(x|c)*, enabling the generation of samples consistent with specific categories. Furthermore, CTGAN incorporates mode-specific normalization for continuous variables, allowing it to capture multimodal and non-Gaussian feature distributions commonly observed in tabular multi-omics data. In addition, its training-by-sampling strategy increases the frequency of underrepresented categories during training, mitigating mode collapse and improving coverage of minority classes. These architectural modifications make CTGAN more suitable than traditional GANs for structured tabular data, particularly in scenarios characterized by heterogeneity and class imbalance.

Ahmed et al. conduct a comprehensive evaluation of multiple GAN variants, including CTGAN, demonstrating their effectiveness in generating synthetic medical tabular data and improving classification performance across several healthcare datasets. However, their evaluation primarily relies on classifier-based metrics and correlation analyses, which may not fully capture data fidelity and complex feature dependencies; in contrast, this work addresses these limitations by leveraging CTGAN within an autoencoder-derived latent space and incorporating explicit fidelity assessments to better preserve the underlying structure of multi-omics data [77].

Although recent work like Variational Autoencoder Generative Adversarial Network (VAE-GAN) [74] has explored the integration of autoencoders and GANs, standard GANs struggle with data that contains a mix of discrete and continuous features [16], while multi-omics data have heterogeneous tabular data. This study addresses two key research gaps in the literature. First, existing approaches to class imbalance in multi-omics cancer datasets commonly rely on traditional oversampling techniques or apply generative augmentation

directly in the original high-dimensional feature space, often leading to noise amplification and poor minority class representation. To address this limitation, our work employs an autoencoder to learn a compact, informative latent representation, enabling data augmentation in a reduced, denoised space. Second, the application of CTGAN to structured tabular biomedical multi-omics data, particularly within a learned latent space, remains largely unexplored. This study explicitly integrates CTGAN with autoencoder-derived latent features, leveraging its conditional sampling capability to generate high-quality synthetic minority class samples, thereby improving the effectiveness of data augmentation for imbalanced cancer outcome prediction.

Diffusion models are a newer class of generative models that simulate the process of data diffusion over time. They are particularly effective in modelling complex data distributions and generating high-quality synthetic data. In multi-omics research, diffusion models like Precious2GPT integrate pre-trained transformers with conditional diffusion to generate multi-omics, multi-species, and multi-tissue data. These models excel in generating representative synthetic data that captures tissue- and age-specific information, making them valuable for drug discovery and aging research [75]. While Precious2GPT is a powerful multimodal model, it is more generalized and not specifically optimized for tabular data.

2.8 Chapter Summary

This chapter reviewed multi-omics concepts, molecular modalities, cancer context (BRCA and BLCA), and key challenges in multi-omics integration. It also introduced machine learning foundations and summarized related work on generative modelling for imbalanced

multi-omics prediction, motivating the methodological framework developed in Chapter 3.

Chapter 3

Linear Latent Space Extraction and Baseline Classification

This chapter discusses the extraction of linear latent space using PCA, followed by handling the class imbalance problem through linear approaches, including SMOTE and ADASYN. Then, the standard baseline classifiers were applied, including Support Vector Machine (SVM)-Radial Basis Function (RBF), Random Forest, and Na ïve Bayes.

The main objective is to identify multi-omics signatures that capture the underlying biological differences between pre-menopausal and post-menopausal breast cancer patients.

3.1 Materials and Preprocessing

3.1.1 Materials

The publicly available TCGA BRCA dataset was employed to investigate menopausal states in breast cancer patients [78]. This dataset comprises three heterogeneous omics layers: gene expression (GE), DNA methylation (DM), and copy number alteration (CNA), thereby enabling a comprehensive multi-omics analysis [36, 79].

Only samples with clearly defined menopausal status (pre-menopause and post-menopause) were retained, while intermediate or ambiguous cases were excluded to ensure a well-defined binary classification task. Initially, 818 samples were available. A filtering process was applied to retain only those samples that contained complete data across all three omics layers. This resulted in a final cohort of 344 samples. Among these, 89 samples correspond to pre-menopausal patients, while 255 samples correspond to post-menopausal patients.

The dataset exhibits high dimensionality, with each omics layer containing thousands of features. The data were obtained from the cBioPortal platform (https://www.cbioportal.org/study/summary?id=brca_tcga_pub2015, accessed April 2026).

3.1.2 Multi-Omics Data Characteristics

Each omics dataset has distinct statistical and biological properties:

- **Gene Expression (GE):** Continuous-valued measurements representing transcript abundance levels obtained from RNA sequencing experiments.

- **DNA Methylation (DM):** Continuous beta values ranging between 0 and 1, representing the methylation level at specific genomic loci.
- **Copy Number Alteration (CNA):** Discrete or continuous values representing genomic amplifications and deletions.

To ensure consistency across modalities, samples were aligned using unique patient identifiers (`SAMPLE_ID`). Only the intersection of samples present in all three omics datasets was retained, ensuring that each sample is represented across all modalities.

3.1.3 Preprocessing

To ensure data quality and suitability for machine learning, several preprocessing steps were applied in a sequential manner. The overall preprocessing pipeline can be summarized as seen in Figure 3.1:

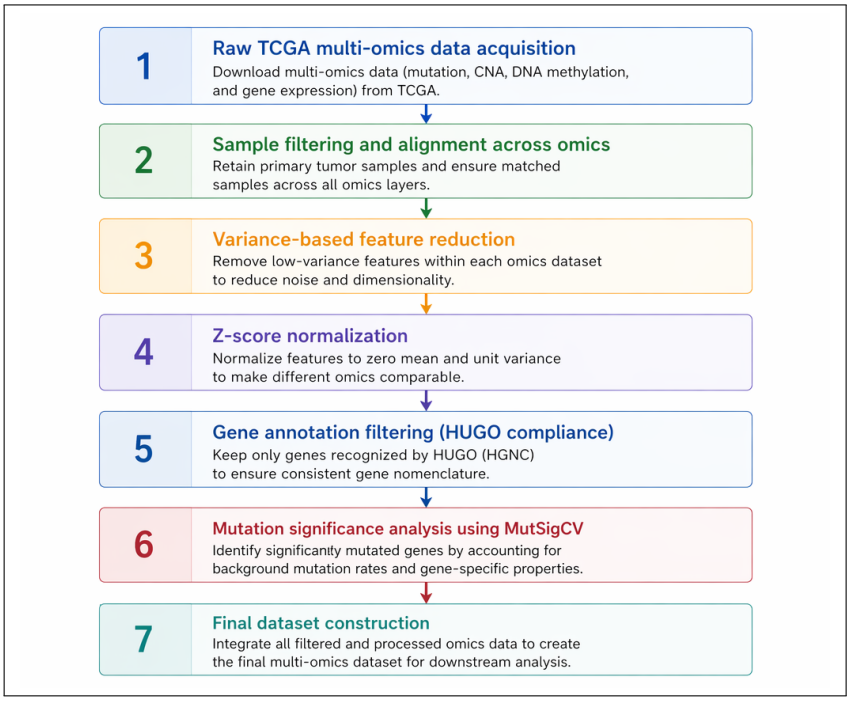


Figure 3.1: The preprocessing pipeline.

Variance Filtering

Gene expression features with extremely low variance were removed, as they contribute minimal discriminative information for classification tasks. Specifically, features with variance below 0.2 were excluded. This step reduced the number of gene expression features from approximately 39,000 to 16,000.

Removing low-variance features mitigates the curse of dimensionality, reduces noise, and improves computational efficiency [80].

Normalization

To ensure comparability across features and omics layers, z-score normalization was applied. For each feature, the normalized value is computed as:

$$z = \frac{x - \mu}{\sigma}$$

where x is the original feature value, μ is the mean, and σ is the standard deviation of that feature.

Normalization ensures that all features are centered around zero with unit variance, thereby preventing features with larger scales from dominating the learning process [81].

Gene Annotation Filtering

Genes that do not conform to the Human Genome Organization (HUGO) nomenclature were removed. This step ensures consistency in gene identifiers and facilitates biological interpretability and reproducibility when comparing results with existing literature and

databases [82].

Mutation Significance Analysis

To identify biologically relevant features, the MutSigCV algorithm was employed [83]. MutSigCV estimates the background mutation rate while accounting for gene-specific covariates, thereby enabling the identification of significantly mutated genes.

Genes with statistical significance defined by $p < 0.05$ and a false discovery rate (FDR) below 0.1 were retained. Based on these criteria, 14 significantly mutated genes were selected and included in the analysis.

This step provides a biologically driven dimensionality reduction, complementing statistical filtering methods.

Class Imbalance Analysis

The dataset exhibits a clear class imbalance, with 89 pre-menopausal samples and 255 post-menopausal samples. Figure 3.2 illustrates this distribution.

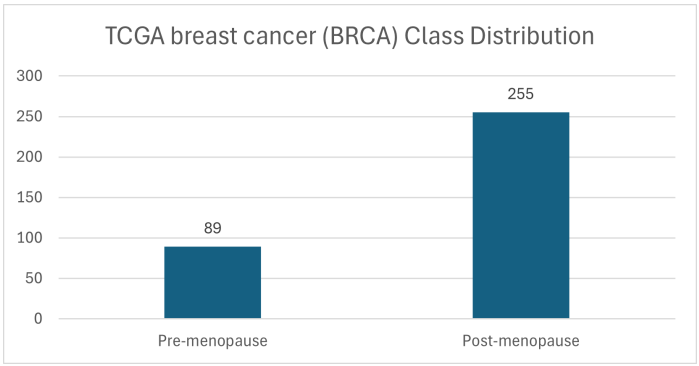


Figure 3.2: Distribution of samples across pre-menopausal and post-menopausal classes.

This imbalance can bias machine learning models towards the majority class, leading

to poor predictive performance in the minority class [84]. This motivates the use of data balancing techniques, which are addressed in subsequent chapters.

Table 3.1 summarizes the dimensionality of the dataset before and after preprocessing.

Table 3.1: Summary of feature dimensionality across omics datasets.

Omics Type	**Initial Features**	**After Filtering**
Gene Expression (GE)	~39,000	~16,000
DNA Methylation (DM)	High-dimensional	Filtered (post-normalization)
Copy Number Alteration (CNA)	High-dimensional	Filtered (post-normalization)

3.2 The Linear Model Workflow

The various types of data and the proposed method are depicted in Figure 3.3. It starts by converting data to lower dimensions, then merging the converted data, generating synthetic samples, and finally building a classification model to predict the status of menopause among breast cancer patients.

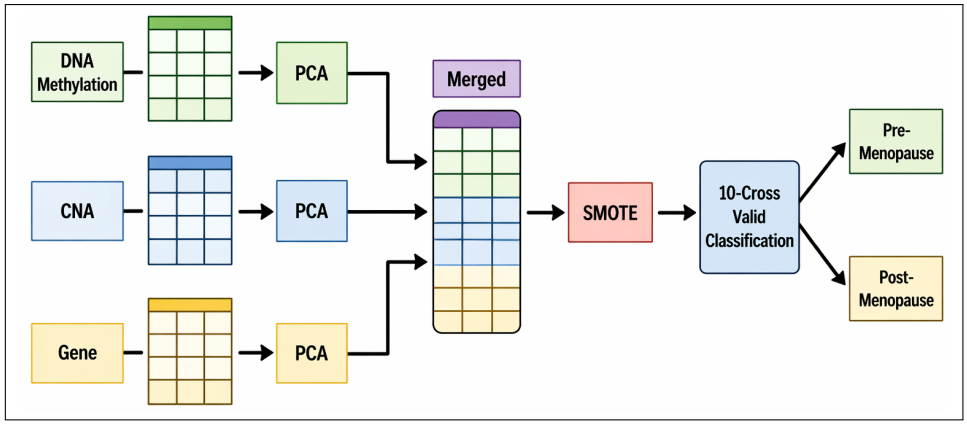


Figure 3.3: The workflow of the linear model, where the input is the multi-omics vectors and the output is the prediction of the menopause status.

3.3 Principal Component Analysis

The model applies PCA to individual omic datasets to reduce complexities, remove noise, and identify shared patterns. Subsequently, these transformed components PCA from different omics datasets can be merged, providing a more harmonized representation that encompasses critical features across various omics layers [85]. This merged representation can then serve as a more robust foundation for constructing a predictive model, improving interpretability, reducing overfitting, and enhancing generalization. The combined utilization of PCA and omics data integration thus equips us with a powerful means to derive meaningful insights and make accurate predictions in complex biological contexts [86]. The first step of PCA is to compute the covariance matrix C of the centred data Z. The covariance between two variables X_{i} and X_{j} is given by the following:

$$\mathrm{Cov}(X_i, X_j) = \frac{1}{n-1} \sum_{k=1}^{n} \left(Z_{ik} - \bar{Z}_i\right) \left(Z_{jk} - \bar{Z}_j\right) \tag{3.1}$$

Therein:

- n is the number of observations (data points),
- Z_{ik} is the value of variable i at observation k,
- $\bar{Z}_i$ is the mean of variable i.

The data matrix is defined as $Z \in \mathbb{R}^{n \times p}$, where n is the number of samples and p is the number of features. Then, the model sorts the eigenvalues in decreasing order and chooses the top-k eigenvectors (principal components) corresponding to the largest eigenvalues. The

last step is to project the centered data Z onto the new basis formed by the selected principal components. The covariance covered in the PCA was 0.95, and the transformed data Y is obtained by the following:

$$Y = Z \times V \tag{3.2}$$

Therein, we then have the following:

- V is the matrix containing the selected eigenvectors as columns.

3.4 Synthetic Minority Over-sampling Technique (SMOTE)

SMOTE is a widely used up-sampling technique for handling imbalanced datasets by generating synthetic samples for the minority class. The method operates by selecting a minority-class sample and interpolating new samples along the line segments connecting it to its nearest minority neighbors. A synthetic sample is generated as follows:

$$x_{\text{new}} = x_i + \lambda(x_j - x_i), \quad \lambda \in [0, 1] \tag{3.3}$$

where x_i and x_j are neighboring minority-class samples and λ is a randomly selected scalar. In this way, SMOTE increases the representation of the minority class while preserving the local structure of the data distribution [7].

As shown in Figure 3.4(B), SMOTE generates synthetic samples mainly within dense regions of the minority-class distribution. This results in a relatively compact synthetic sample distribution, where the generated points remain close to the original observations. Such behaviour is desirable in applications where preserving the observed structure of the

data is important.

3.5 Adaptive Synthetic (ADASYN)

ADASYN is an adaptive synthetic sampling method that also addresses class imbalance by generating minority-class samples, but unlike SMOTE, it allocates more synthetic samples to regions that are harder to learn [8]. In particular, ADASYN focuses on minority samples located in sparse regions or near the class boundary, thereby adapting sample generation according to local data complexity.

As illustrated in Figure 3.4(C), ADASYN produces a wider spread of synthetic samples, especially in low-density areas and near class boundaries. This behaviour may improve the representation of difficult regions; however, it also introduces potential uncertainty regarding the reliability of synthetic samples generated in weakly supported areas, particularly when the original minority class is itself sparsely distributed.

3.6 Comparison Between SMOTE and ADASYN

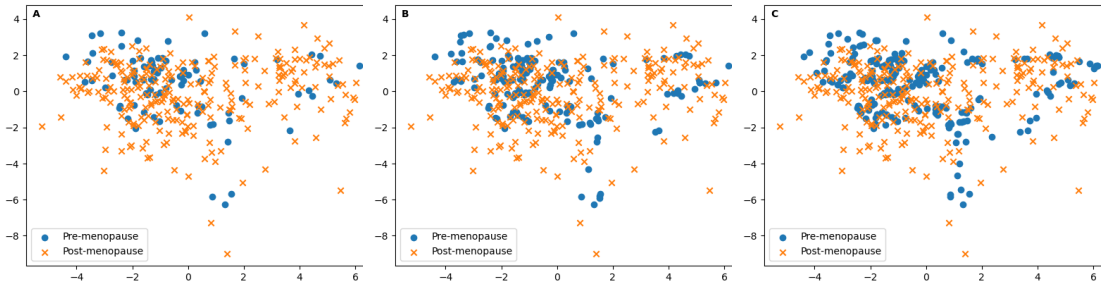
A direct comparison between SMOTE and ADASYN highlights an important difference in how each method expands the minority class. Figure 3.4(A) shows the original sample distribution projected onto the first two principal components, where the minority class is visibly underrepresented. After up-sampling, Figure 3.4(B) shows that SMOTE generates synthetic samples primarily within existing minority-class clusters, preserving local density and structural coherence. In contrast, Figure 3.4(C) shows that ADASYN produces more synthetic samples in sparse regions and near class boundaries, leading to a broader and more

dispersed distribution.

To further compare the similarity of the generated samples to the original minority-class distribution, a Kolmogorov–Smirnov test [87] was performed on the first two principal components (PCA1 and PCA2). The first test compared the SMOTE synthetic samples with the original minority-class samples, while the second test compared the ADASYN synthetic samples with the same original minority-class samples.

The resulting p-values for the SMOTE comparison were [PCA1 = 0.9999, PCA2 = 0.9999], whereas the corresponding values for ADASYN were [PCA1 = 0.9895, PCA2 = 0.9539]. Although both tests produced statistically insignificant results, indicating no strong evidence of distributional difference, the consistently higher p-values obtained for SMOTE suggest a closer empirical alignment with the original minority-class distribution under the current dataset conditions.

Overall, both SMOTE and ADASYN effectively mitigate class imbalance; however, SMOTE demonstrates stronger consistency with the observed minority-class structure in this study. Within the linear resampling framework, this makes SMOTE the more suitable method when preserving distributional fidelity is a primary concern.



(A) Original **(B) SMOTE** **(C) ADASYN**

Figure 3.4: Class distribution scatter plots based on the first and second principal components: (A) original samples, (B) samples after up-sampling using SMOTE, and (C) samples after up-sampling using ADASYN.

3.7 Classification Models

The model combines the reduced data set from all omics into one data set labeled with 0 for pre-menopause samples and 1 for post-menopause samples. Three standard machine learning classifiers were applied to compare and select the appropriate method that works with the reduced representation of various types of omics data, including Naïve Bayes, Random Forest, and Support Vector Machine (SVM) with a Gaussian kernel.

3.7.1 Naïve Bayes Classifier

The Naïve Bayes classifier is a fundamental implementation of the Bayesian classifier. Despite its simplistic assumptions, such as feature independence given the class, Naïve Bayes often demonstrates impressive performance across various tasks. Given a dataset with features $X = x_1, x_2, \ldots, x_\mathrm{m}$ and a set of classes $C = c_1, c_2, \ldots, c_\mathrm{m}$, the Naïve Bayes classifier computes the posterior probability of each class given the features using Bayes' theorem:

$$P(c_\mathrm{i} \mid x_1, x_2, \ldots, x_\mathrm{m} \prod_{j=1}^{n} P(x_\mathrm{j}|c_\mathrm{i})) \tag{3.4}$$

Therein, we have the following:

- $P(c_\mathrm{i})$ is the prior probability of class c_i.
- $P(x_\mathrm{j} \mid c_\mathrm{i})$ is the likelihood of feature x_j given class c_i.

Due to its computational simplicity and ability to handle high-dimensional data, Naïve

Bayes remains a popular choice for general classification, especially where efficiency and interpretability are crucial [88].

3.7.2 Random Forest Classifier

The Random Forest classifier is a prominent ensemble learning method that has gained substantial attention in various machine learning applications. By aggregating the predictions of multiple decision trees, this approach provides enhanced accuracy, robustness, and flexibility.

Given an ensemble of T decision trees, denoted by $\{h_1, h_2, \ldots, h_T\}$, each tree produces a predicted class label for an input instance x. The final predicted class label $C(\mathrm{x})$ is determined through majority voting as follows:

$$C(\mathbf{x}) = \arg\max_{c \in C} \sum_{t=1}^{T} \delta\left(c, h_t(\mathbf{x})\right) \tag{3.5}$$

Therein, we have the following:

- T is the total number of decision trees in the ensemble.
- $h_t(\mathbf{x})$ is the class label predicted by the t-th decision tree for input instance $\mathbf{x}$.
- C is the set of possible class labels.
- $\delta(a, b)$ is the Kronecker delta function, which equals 1 if $a = b$ and 0 otherwise [89].

3.7.3 Support Vector Machine

The SVM is a supervised learning model used for classification and regression tasks [90]. It aims to find an optimal separating hyperplane that maximizes the margin between classes. The margin is defined as the distance between the decision boundary and the closest data points from each class, known as support vectors [91].

In high-dimensional and nonlinear settings, SVM leverages kernel functions to project the data into a higher-dimensional feature space where linear separation becomes feasible. In this work, the Gaussian RBF kernel is employed.

The decision function of SVM is given by:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \tag{3.6}$$

Therein, we have the following:

- $f(x)$ is the decision function.
- α_i are the Lagrange multipliers associated with the support vectors.
- x_i is the i-th training input vector.
- $y_i \in \{-1, +1\}$ is the class label of the i-th instance.
- $K(x, x_i)$ is the kernel function measuring similarity between x and x_i.
- b is the bias term.

The predicted class label is determined as:

$$\hat{y} = \text{sign}(f(\mathbf{x})) \tag{3.7}$$

The Gaussian RBF kernel is defined as:

$$K(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\gamma \|\mathbf{x} - \mathbf{x}_i\|^2\right) \tag{3.8}$$

Therein, we have the following:

- $\exp(\cdot)$ is the exponential function.
- $\gamma > 0$ is a hyperparameter controlling the width of the kernel and the smoothness of the decision boundary.
- $\|x - x_i\|^2$ is the squared Euclidean distance between x and x_i.

3.8 Results and Experiments

This section presents the experimental setup, evaluation metrics, and results obtained from the proposed classification framework.

3.8.1 Running Environment

The experiments were conducted on a Microsoft Azure virtual machine (*Standard_F4s_v2*). This instance belongs to the compute-optimized F-series and is configured with 4 virtual CPUs (vCPUs), 8 GB of RAM, and supports premium SSD-based storage, making it suitable

for CPU-based machine learning workloads and moderate-scale data processing tasks. The coding and execution environment was implemented using Jupyter Notebook.

3.8.2 Evaluation Metrics

The three classification models were evaluated using 10-fold cross-validation to ensure robust estimation of generalization performance. To assess the models from multiple perspectives, several standard evaluation metrics were employed, all derived from the confusion matrix.

In this context, true positives (TP) denote correctly predicted positive instances, true negatives (TN) denote correctly predicted negative instances, false positives (FP) represent negative instances incorrectly classified as positive, and false negatives (FN) represent positive instances incorrectly classified as negative.

The evaluation metrics are defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.9}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.10}$$

$$F1\text{-score} = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.11}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.12}$$

$$\text{AUROC} = \int_0^1 \text{TPR}(\text{FPR})\, d(\text{FPR}) \tag{3.13}$$

where the true positive rate (TPR) and false positive rate (FPR) are defined as:

$$\text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN} \tag{3.14}$$

Precision measures the proportion of correctly predicted positive instances among all predicted positives, while recall quantifies the ability of the model to identify actual positive instances. The F1-score provides a harmonic balance between precision and recall, particularly useful when there is an uneven class distribution. Accuracy reflects the overall proportion of correctly classified instances but may be misleading in imbalanced datasets. Therefore, Area Under the Receiver Operating Characteristic (AUROC) is additionally used as a threshold-independent metric to evaluate the model's ability to discriminate between classes across all possible decision thresholds.

3.8.3 Hyper-parameters Settings

The hyper-parameters of the SVM-RBF classifier were configured with a kernel coefficient $\gamma = 0.02$, while the regularization parameter C was retained at its default value of 1. For the Random Forest classifier, the number of trees was set to 1,000 to ensure stable ensemble estimates, and the number of features considered at each split was defined as $\log_2(m)$, where m denotes the total number of input features. The Naïve Bayes classifier was implemented using the default settings provided in the scikit-learn library.

3.8.4 Results

Table 3.2 presents the classification performance of Naïve Bayes, Random Forest, and SVM-RBF across multiple evaluation metrics, including accuracy and AUROC. Among the evaluated models, Random Forest achieved the highest AUROC (0.962), indicating superior discriminative capability across varying classification thresholds. This performance can be attributed to the ensemble nature of Random Forest, which captures complex nonlinear relationships and higher-order feature interactions through aggregation of multiple decision trees. Given the high-dimensional structure of the data, even after dimensionality reduction via PCA, such flexibility enables Random Forest to better model residual nonlinear dependencies and reduce variance through bootstrap aggregation.

In contrast, the SVM-RBF classifier achieved the highest accuracy (89.53%), marginally outperforming Random Forest (88.54%) at a fixed decision threshold. This suggests that the SVM decision boundary is well-optimized for the dominant class distribution under the selected hyper-parameters. However, its lower AUROC (0.886) relative to Random Forest indicates reduced ranking performance across thresholds. This discrepancy highlights a key distinction between threshold-dependent and threshold-independent metrics: while accuracy reflects performance at a single operating point, AUROC captures the model's overall ability to separate classes. The results suggest that Random Forest provides more robust classification performance in settings where class distributions are imbalanced or decision thresholds may vary.

Naïve Bayes exhibited the weakest performance across all evaluation metrics. This outcome is consistent with its strong assumption of conditional independence among features,

whereby the joint likelihood is factorized as the product of individual feature likelihoods given the class label. In high-dimensional biological datasets, such as the one considered in this study, features often exhibit substantial interdependencies due to underlying biological processes and regulatory mechanisms. Even after applying PCA, which primarily removes linear correlations, nonlinear dependencies may persist. As a result, the independence assumption is violated, leading to model misspecification. Consequently, correlated features are effectively double-counted, producing biased posterior probability estimates and limiting the model's ability to capture the true decision boundary.

From a bias-variance perspective, Naïve Bayes is characterized by high bias and low variance, which results in underfitting when feature dependencies play a significant role in class discrimination. In contrast, SVM-RBF provides a balance between bias and variance through its kernel-based nonlinear mapping, while Random Forest reduces variance through ensemble averaging without substantially increasing bias. This balance enables Random Forest to achieve better generalization performance in complex, high-dimensional settings.

Overall, the results demonstrate that models capable of capturing nonlinear relationships and feature interactions-such as Random Forest and SVM-RBF are better suited for this classification task. The observed performance differences further emphasize the importance of aligning model assumptions with the statistical properties of the data, particularly in domains characterized by high dimensionality and feature dependency.

Table 3.2: Performance measurements of the three classification models on the breast cancer multi-omics dataset.

Model	**Precision**	**Recall**	**F1 Measure**	**AUCROC**	**Accuracy**
Naïve Bayes	0.7614	0.788	0.775	0.816	72.98%
Random Forest	0.914	0.886	0.900	0.962	88.45%
SVM-RBF	0.919	0.894	0.907	0.886	89.14%

3.9 Gene Expression Feature Importance Validation

To quantify the contribution of individual gene expression features to the classification model, the techniques Explainable Artificial Intelligence (XAI), specifically the SHAP method [92] was applied, in conjunction with the XGBoost model [93]. SHAP is a model-agnostic interpretability framework grounded in cooperative game theory, which assigns each feature a contribution score reflecting its impact on the model's prediction.

In this context, the values of SHAP provide a local explanation for each prediction by quantifying how much each gene expression characteristic increases or decreases the predicted outcome relative to a baseline. A positive SHAP value indicates that the feature contributes toward predicting a particular class (e.g., post-menopause), whereas a negative value indicates a contribution toward the opposite class. The magnitude of the SHAP value reflects the strength of this contribution.

The SHAP values $\phi_i(f)$ for feature i are formally defined as:

$$\phi_i(f) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[f(S \cup \{i\}) - f(S) \right] \tag{3.15}$$

where:

- N represents the set of all features,
- S is a subset of features excluding feature i,
- $|S|$ denotes the cardinality of S,
- $f(S)$ represents the prediction of the model using only features in S.

This formulation ensures that feature contributions are fairly attributed by considering all possible feature combinations. In practice, SHAP enables both local interpretability (for individual predictions) and global interpretability (through aggregation between samples), allowing the identification of the most influential gene expression features in distinguishing between pre-menopause and post-menopause breast cancer samples.

To compute SHAP values, the XGBoost model was utilized, which is an ensemble of decision trees. The prediction for a sample i is given by:

$$\hat{y}_i = \sum_{k=1}^{K} \psi_k(\mathbf{x}_i), \quad \psi_k \in \ominus \tag{3.16}$$

where:

- K is the number of trees in the ensemble,
- ψ_k represents an individual decision tree.

The combination of SHAP with XGBoost is particularly effective due to the Tree-SHAP algorithm [94], which enables efficient and exact computation of the contribution of characteristics for tree-based models. This allows for scalable interpretation even in high-dimensional multi-omics datasets.

In general, this approach provides a robust framework for interpreting complex nonlinear models and facilitates the identification of biologically significant gene expression patterns associated with menopausal status in breast cancer.

3.9.1 Feature Importance Analysis Results

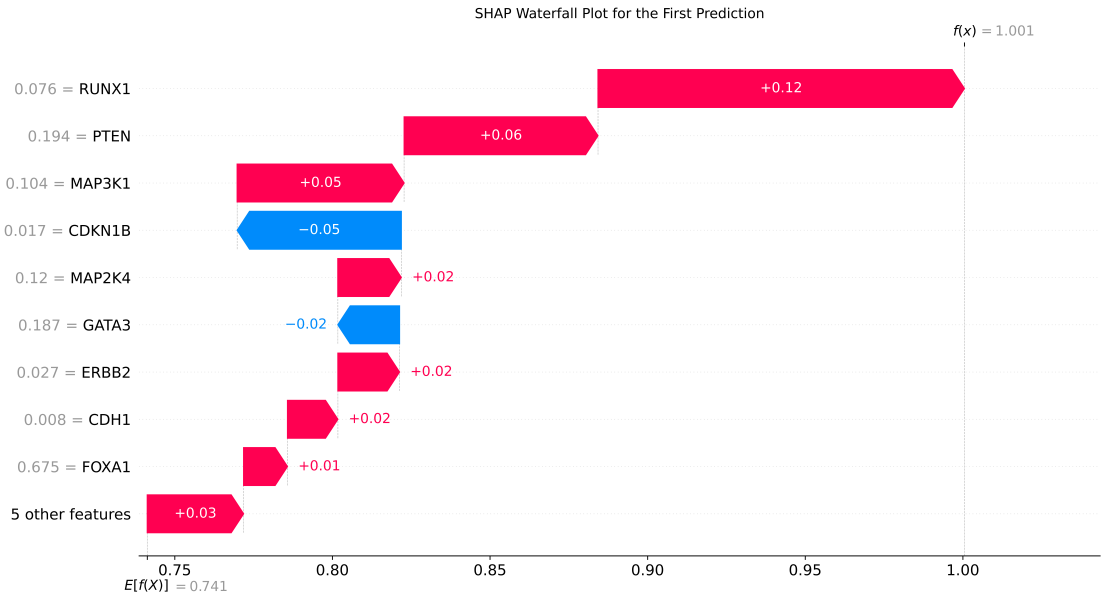


Figure 3.5: SHAP waterfall plot illustrating the contribution of individual gene expression features to the XGBoost prediction for a representative sample.

To show the importance of gene expression values, the SHAP model was utilized in XGBoost to determine the most discriminative genes. Figure 3.5 shows how individual gene expression features contributed to the prediction of the model for a representative sample. The results suggest that *RUNX1*, *PTEN*, *MAP3K1*, and *CDH1* significantly distinguished

the two classes.

This visualization represents a local explanation for a single sample, illustrating how feature contributions combine to produce the final prediction.

3.9.2 Kaplan-Meier Survival Results

The Kaplan–Meier estimator is a nonparametric method that is widely used for estimating the survival function from time-to-event data. The survival function, denoted as $S(t)$, represents the probability that the death event occurs after a specified time t. The estimator is calculated based on the observed survival times in a given dataset. Let n be the number of individuals in the sample, d be the number of observed events, and $t_1, t_2, \ldots t_i, t_j$ be the observed event times. The Kaplan–Meier estimator at time t is given by the following formula:

$$S(t) = \prod_{i=1}^{j} \left(1 - \frac{d_i}{n_i}\right) \tag{3.17}$$

Therein, we have the following:

- j represents the distinct event times.
- d_i is the number of events at time t_i.
- n_i is the number of individuals at risk just before time t_i.

The product is taken over all event times t_i less than or equal to t. This estimator allows us to visualize and analyze survival curves over time, thereby providing valuable insights into the probability of survival at different time points in a study [95].

3.9.3 Survival Analysis Results

The Kaplan-Meier survival curves for the pre-menopause and post-menopause cohorts (Figure 3.6) show a higher survival probability for the pre-menopause class throughout the follow-up period. The clear separation between the two curves is supported by the log-rank test ($p < 0.05$), confirming that the identified multi-omics signatures are statistically significant predictors of patient survival.

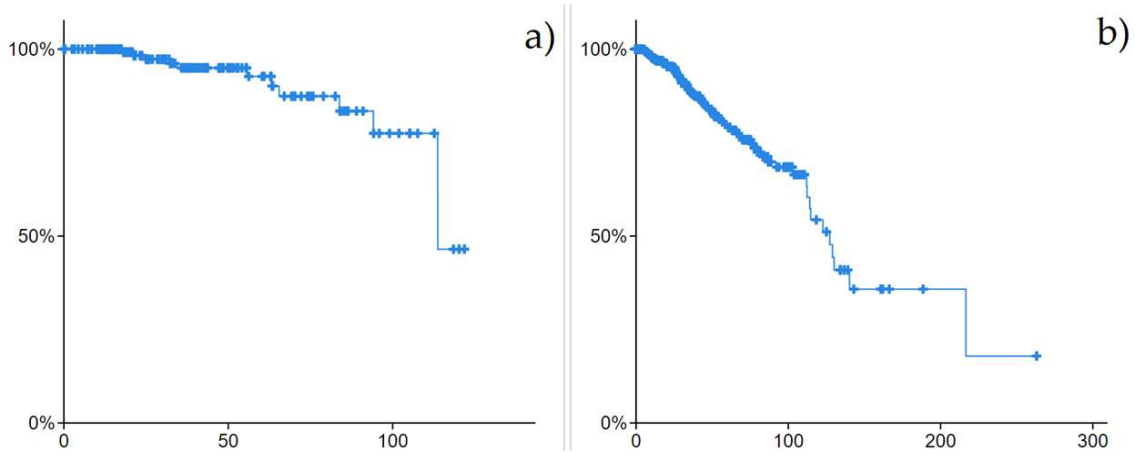


Figure 3.6: Kaplan−Meier survival curve for (**a**) pre-menopause breast cancer cohort and (**b**) post-menopause breast cancer cohort.

In particular, the post-menopause (high-risk) group exhibits a steeper decline in survival probability during the initial 24 months, suggesting that integrated features are particularly effective in capturing early-stage aggressive disease progression. In contrast, the plateau observed in the pre-menopause curve (low-risk) indicates a subset of patients with sustained long-term survival, likely characterized by the absence of specific copy number variations and methylation patterns identified as "critical" by the SHAP analysis. This visual divergence reinforces the utility of the model in clinical stratification, providing a potential window for clinicians to implement more aggressive therapeutic interventions for patients who fall into

the high-risk quadrant at the time of diagnosis.

3.9.4 Pathway and Functional Enrichment Results

By running GO enrichment analysis using ShinyGO [96] on the selected genes, many of these genes were confirmed to be related to various types of cancer, as seen in Figure 3.7. It can be noticed in Figure 3.7 that the "ErbB signaling pathway" had the highest fold enrichment score. KEGG pathway [97] analysis was applied to the selected genes to visualize the ErbB signaling pathway as shown in Figure 3.8.

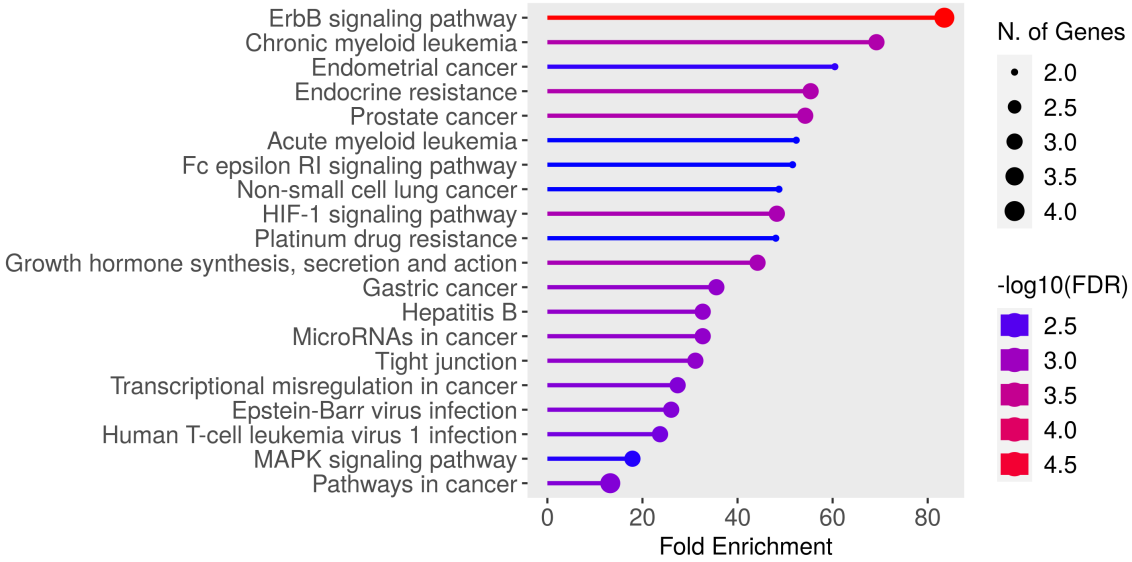


Figure 3.7: GO enrichment analysis of the selected genes, showing fold enrichment scores on the x-axis and biomedical terms on the y-axis.

3.10 Validation of Differential Gene Expression Using Raw TCGA Data

To further support the findings obtained from the proposed nonlinear framework, an additional analysis was conducted directly on the raw TCGA gene expression data. The purpose

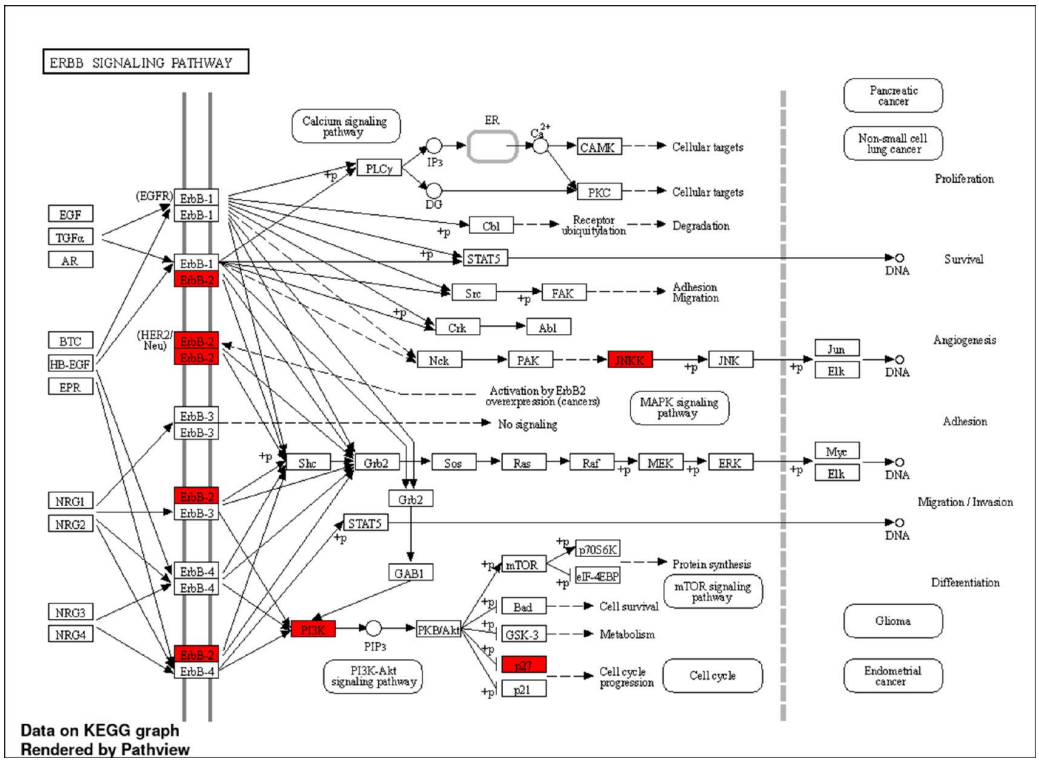


Figure 3.8: KEGG pathway analysis for ErbB signalling rendered by Pathview, with selected genes highlighted in red.

of this experiment was to examine whether the differences observed between the two clinical groups are also visible in the original expression space, without relying on the latent representation produced by the autoencoder or on any synthetic samples generated during the augmentation stage. In this way, the analysis serves as a complementary confirmation that the observed class separation is grounded in the biological signal present in the raw data.

A subset of fourteen genes was selected for this experiment based on their biological relevance and their importance in the earlier analyses. The raw mRNA expression file obtained from the TCGA portal was used directly, and the samples were divided into two groups according to menopausal status. For each gene, the distribution of expression values was compared between the pre-menopausal and post-menopausal groups. Because gene

expression data frequently deviates from normality and may contain skewness and outliers, the Mann-Whitney U test was employed to assess whether the difference in expression between the two groups was statistically significant.

Figure 3.9 presents the boxplot-based comparison of the fourteen selected genes across the two clinical groups. For each gene, two adjacent boxplots are shown, representing the pre-menopausal and post-menopausal samples, respectively. The boxplots summarize the median, interquartile range, dispersion, and outlier structure of the expression values. In addition, the corresponding p-value from the Mann–Whitney U test is displayed above each gene, and significance is indicated using the standard notation of one star for $p < 0.05$, two stars for $p < 0.01$, and three stars for $p < 0.001$.

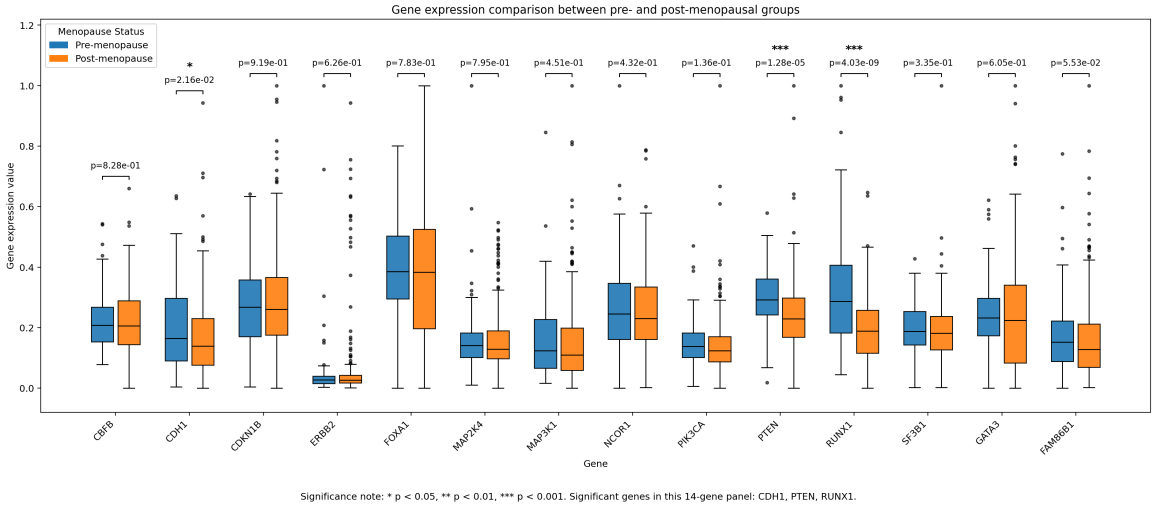


Figure 3.9: Boxplot comparison of raw TCGA gene expression values for fourteen selected genes across pre-menopausal and post-menopausal groups. For each gene, the p-value from the Mann-Whitney U test is shown above the corresponding pair of boxplots, and statistical significance is indicated using star notation. Significant differences were observed for *CDH1*, *PTEN*, and *RUNX1*, whereas the remaining genes showed substantial overlap between the two groups.

The results indicate that only a subset of the fourteen genes exhibits statistically significant differences between the two menopausal groups. Among all genes, *RUNX1* shows

the strongest separation, with a highly significant difference between the two groups. This behaviour is visually evident in the boxplots, where the distribution of expression values is shifted more clearly than in the majority of the remaining genes. Similarly, *PTEN* also demonstrates a strong statistically significant difference, indicating that its expression profile varies meaningfully between the pre-menopausal and post-menopausal samples.

The gene *CDH1* also shows a statistically significant difference, although its separation is weaker than that of *RUNX1* and *PTEN*. Its boxplots suggest a moderate shift in expression between the two groups, which is sufficient to reach significance in the direct comparison of the raw data. In contrast, genes such as *MAP3K1*, *GATA3*, *PIK3CA*, *NCOR1*, *CBFB*, *MAP2K4*, *SF3B1*, and others display substantial overlap between the two distributions. This overlap suggests that these genes have relatively stable expression profiles across the two clinical conditions and may contribute less strongly to direct group separation when considered individually in the raw expression space.

Overall, the boxplot analysis shows that the discriminatory signal is not uniformly distributed across all genes. Instead, a smaller subset of genes appears to drive the most evident differences between the two classes, while the remaining genes behave in a more stable or weakly differentiating manner.

These observations are consistent with the results obtained from the nonlinear approach presented earlier in this chapter. In particular, the genes that show clearer differences in the raw expression space, especially *RUNX1* and *PTEN*, are in agreement with the broader class separation observed after nonlinear feature extraction and classification. This consistency is important because it shows that the proposed model is not creating artificial separation, but rather is capturing and enhancing biologically meaningful variation that is already present

in the original TCGA data.

At the same time, the presence of several genes with overlapping expression distributions reinforces the motivation for employing nonlinear dimensionality reduction and representation learning. When genes are examined individually, many of them do not provide strong direct separation between the two classes. However, when these genes are considered jointly within the nonlinear framework, their combined contribution can still improve class discrimination. Therefore, this raw-data analysis complements the nonlinear results by showing that while only a subset of genes exhibits individually significant expression changes, the overall predictive structure emerges more clearly when the multivariate relationships among genes are modeled through the proposed framework.

This experiment should be interpreted as a supportive analysis based directly on raw data obtained from the TCGA portal. Its role is not to replace the nonlinear pipeline, but to provide an additional layer of interpretability and transparency. By showing that selected genes already demonstrate meaningful differential behaviour in the original expression space, the analysis strengthens confidence in the validity of the overall modelling framework. At the same time, the results also illustrate why relying solely on single-gene comparisons may be insufficient for fully characterizing the distinction between the two menopausal groups. The proposed nonlinear method remains essential for capturing the more complex multigene structure underlying the classification task.

3.11 Discussion

The identified gene expression features demonstrated strong discriminative power in classifying pre-menopause and post-menopause breast cancer samples, as evidenced by the SHAP-based feature importance analysis and classification performance results.

The explainable AI analysis highlighted *RUNX1*, *PTEN*, *MAP3K1*, and *CDH1* as the most influential features contributing to the classification model. However, the direct analysis of the raw TCGA gene expression data revealed that not all of these genes exhibit statistically significant differences when considered individually. In particular, *RUNX1* and *PTEN* demonstrated strong and highly significant differential expression between the two menopausal groups, while *CDH1* showed a moderate level of significance. In contrast, *MAP3K1* did not exhibit statistically significant differences in the raw expression space, despite its importance in the model.

This observation highlights an important distinction between the importance of features in a multivariate nonlinear model and the univariate statistical significance. Although some genes may not show strong individual separation, they can still contribute meaningfully to classification when combined with other features in a nonlinear framework.

Tian et al. reported the ErbB signaling pathway in association with menopausal syndrome in their ontological analysis [98]. Pei et al. found a cardiorenal disease connection during post-menopause involving ErbB signaling pathway genes [99], which motivated the survival analysis performed on the two cohorts in this study.

Riggio stated that *RUNX1* acts as a tumour suppressor in the early stages of breast cancer, while acting as a pro-oncogene in the later stages of mammary tumourigenesis [100]. This

dual behaviour aligns with the strong differential expression observed in this study. While Zhang et al. reported no significant correlation between *RUNX1* expression and menopause status, the results presented here suggest that *RUNX1* may still play a discriminative role when considered within a broader multigene context. Similarly, *PTEN* remains a critical gene in tumourigenesis and prognosis of breast cancer [101], consistent with its strong significance in both the model and the analysis of raw data.

Rebbeck et al. reported that *MAP3K1* influences breast cancer susceptibility through interactions with hormone exposure [102]. Although *MAP3K1* did not show significant individual expression differences in this study, its contribution to the model suggests that its role may be context-dependent and mediated through interactions with other genes. Post-menopausal women with a higher histological grade and PR-negative tumours exhibited increased methylation of the *CDH1* promoter [103], which is consistent with the moderate differential expression observed for *CDH1*.

In future research, expanding the proposed framework to include additional omics data types, including proteomics and metabolomics, can provide a more holistic understanding of the molecular landscape associated with menopausal status in breast cancer. PCA-based reduction may assist in integrating metabolomic and proteomic measurements into a unified representation. Investigating longitudinal menopausal transitions and incorporating clinical variables such as hormone receptor status and treatment history can further improve the interpretability and clinical relevance of identified biomarkers.

The status of hormone receptors can be investigated by studying the resulting folded enrichment pathways in Figure 3.7, and the treatment history can be incorporated as an additional label along with the menopausal status.

Although multi-omics datasets specific for menopausal breast cancer remain limited, advances in sequencing technologies are expected to provide richer datasets for future validation. Additionally, Pearson correlation analysis across genes in each omic layer has been included in the Supplementary Materials (Figures S1–S3). The crosstalk between omics layers remains a limitation and can be further explored using network-based approaches.

3.12 Conclusions

This work proposed a multi-omics based machine learning pipeline to classify menopausal status in breast cancer patients. The study aimed to identify biomarkers that reflect the molecular differences between pre-menopausal and post-menopausal conditions using gene expression, copy number alteration (CNA), and DNA methylation data.

To address class imbalance, the Synthetic Minority Over-sampling Technique (SMOTE) was employed, enabling improved model training and generalization. Among the evaluated classifiers, Random Forest achieved the highest AUCROC (0.962), while SVM-RBF demonstrated the highest accuracy (89.53%).

The SHAP-based explainable AI analysis identified *RUNX1*, *PTEN*, *MAP3K1*, and *CDH1* as the most influential features in distinguishing menopausal status. However, further validation using raw TCGA gene expression data demonstrated that only a subset of these genes exhibits statistically significant individual differences. In particular, *RUNX1* and *PTEN* showed strong and consistent differential expression, while *CDH1* showed moderate significance and *MAP3K1* did not exhibit significant individual separation.

These findings emphasize that while some biomarkers are directly observable in the raw

data, others contribute to classification performance through complex multivariate interactions captured by the nonlinear framework. This highlights the importance of combining statistical analysis with machine learning and explainable AI techniques.

Pathway analysis revealed associations with key cancer-related pathways, including the ErbB signaling pathway, which has been linked to menopausal syndrome and cardiorenal disease. Furthermore, survival analysis demonstrated significant differences between the two groups, supporting the hypothesis that pre-menopausal and post-menopausal breast cancer represent biologically distinct conditions.

Overall, the integration of multi-omics data with advanced machine learning and explainable AI provides a robust and interpretable framework for identifying meaningful biomarkers in complex biomedical datasets.

Chapter 4

Nonlinear Representation Learning Using Autoencoders and Generative Modelling for Imbalanced Multi-Omics Data

4.1 Introduction

In Chapter 3, PCA was employed as a linear dimensionality reduction technique to address the high dimensionality of multi-omics data. While PCA provides an efficient projection into a lower-dimensional space, it is inherently limited to linear transformations and may fail to capture the complex nonlinear relationships that exist across biological data modalities such as gene expression, DNA methylation, and CNA.

To overcome these limitations, this chapter introduces autoencoders as a nonlinear representation learning framework. Autoencoders are neural network-based models capable

of learning compact and informative latent representations of high-dimensional data by capturing nonlinear dependencies. This makes them particularly suitable for multi-omics integration, where interactions between features are often complex and nonlinearly correlated.

The primary objective of this chapter is to construct a shared latent representation of multi-omics data using autoencoders, which will later serve as the foundation for addressing class imbalance using generative models.

4.2 Materials and Methods

In addition to the BRCA dataset used in the linear modelling experiments, a second dataset corresponding to bladder cancer (BLCA) was introduced in the nonlinear generative modelling stage. This inclusion allows for evaluating the generalizability of the proposed framework across different cancer types. The same preprocessing pipeline described in Section 3.1 (Chapter 3) was applied to ensure consistency and comparability across datasets.

A schematic representation of the proposed model is provided in Figure 4.1, with further details explained below as follows.

4.2.1 Materials

The proposed model was applied to two publicly available datasets. The first is a CTGAN BRCA dataset, which contains DNA methylation, CNA, and gene expression data [104, 105] to predict the meno-pausal status (pre- versus post-). The second is a CTGAN BLCA dataset, which contains DNA methylation, CNA, and microRNA (miRNA) [106] to predict the TMB

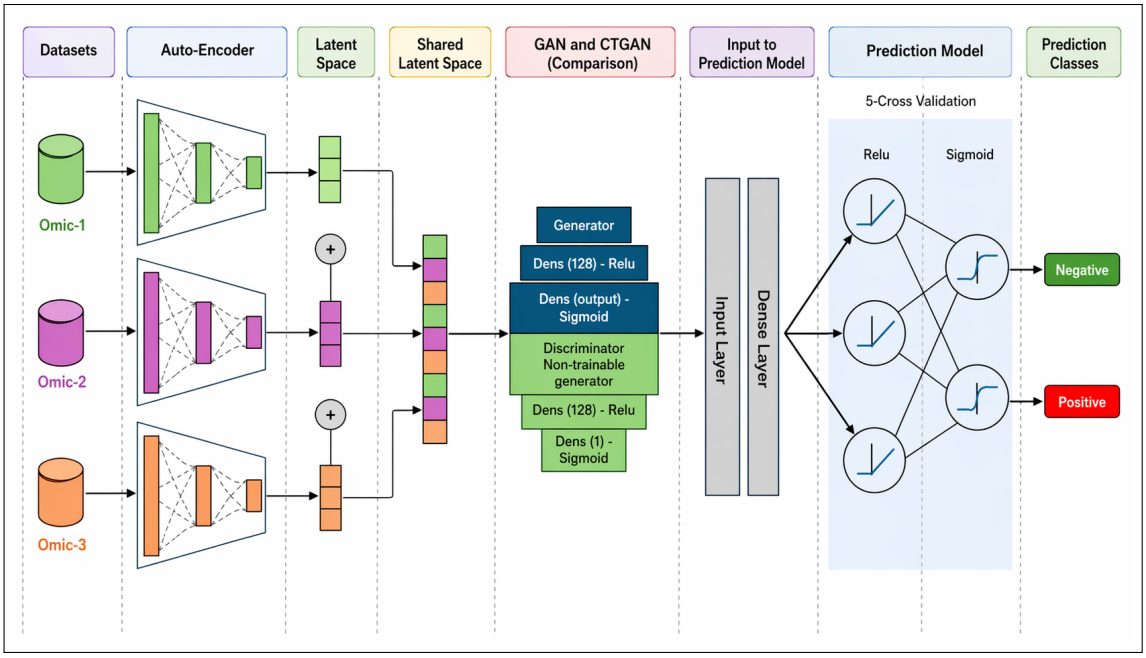


Figure 4.1: The workflow of the proposed model.

level (low- versus high-). From the distribution of the classes, as seen in Table 4.1, it shows that both datasets suffer from class-imbalance.

Table 4.1: Metadata summary of the datasets used in this study.

Dataset	**Omics Data Types**	**Class Distribution**
TCGA Breast Cancer (BRCA)	DNA methylation, gene expression	Pre-menopause: 89 Post-menopause: 255
TCGA Bladder Cancer (BLCA)	DNA methylation, mRNA expression	Low-TMB: 297 High-TMB: 107

4.3 Autoencoder Architecture

An autoencoder is a neural network composed of two main components: an encoder and a decoder. The encoder compresses the input data into a lower-dimensional latent representation, while the decoder reconstructs the original input from this compressed representation.

4.3.1 Encoder

The encoder is responsible for mapping the input data into a latent space. The overall architecture of the autoencoder, including the encoder, latent space, and decoder components, is illustrated in Figure 4.2.

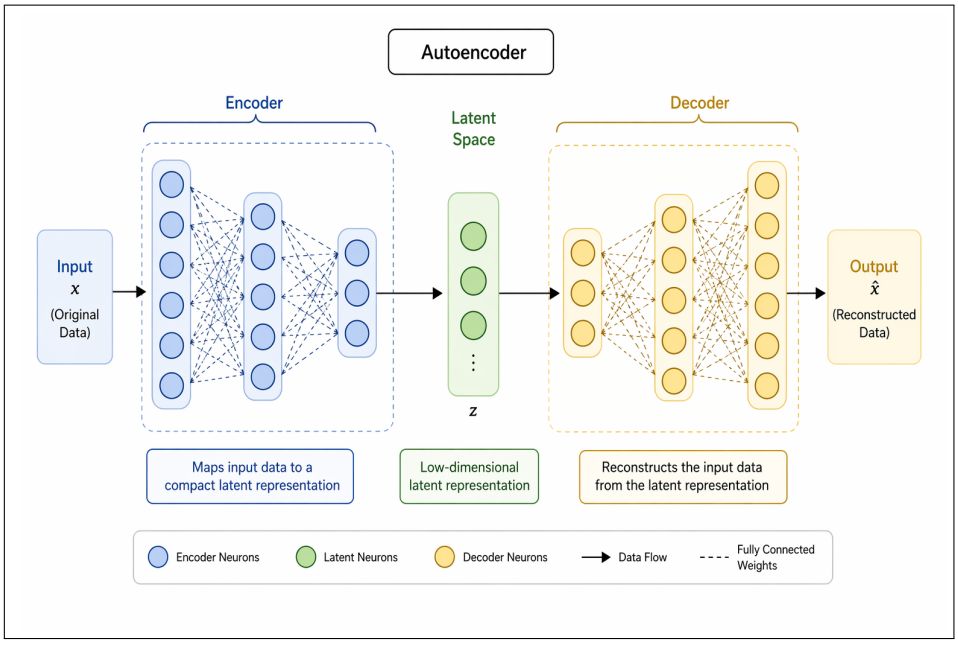


Figure 4.2: Autoencoder architecture illustrating the encoder-decoder structure. The encoder maps the input data into a compact latent representation, while the decoder reconstructs the original input from this representation. The neural network layers highlight the nonlinear transformation and compression process.

It is implemented as a dense neural network layer with a Rectified Linear Unit (ReLU) activation function. The mathematical formulation of the encoder is given by:

$$\mathbf{h} = f_{\text{encoder}}(\mathbf{x}) = \phi(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1), \tag{4.1}$$

where x represents the input data, h is the latent representation, W_1 and b_1 denote the weights and biases of the encoder, respectively, and $\phi(x) = \max(0, x)$ is the ReLU activation function.

4.3.2 Decoder

The decoder reconstructs the original input from the latent representation. It is implemented as a dense neural network layer with a sigmoid activation function:

$$\mathbf{x}' = f_{\text{decoder}}(\mathbf{h}) = \sigma(\mathbf{W}_2\mathbf{h} + \mathbf{b}_2), \tag{4.2}$$

where x' is the reconstructed input, and W_2 and b_2 represent the decoder parameters.

4.3.3 Training Objective

The autoencoder is trained to minimize the reconstruction error between the input and its reconstruction. In this work, the Mean Squared Error (MSE) is used as the loss function:

$$\mathcal{L}(\mathbf{x}, \mathbf{x}') = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{x}'_i)^2, \tag{4.3}$$

where N is the number of samples in the batch.

During training, the parameters (W_1, b_1, W_2, b_2) are optimized to minimize this loss, enabling the model to learn a compressed representation that preserves the essential structure of the input data.

4.3.4 Latent Space Representation

After training, the encoder component of the autoencoder generates latent representations of the input data. It is important to emphasize that the latent space used in this work is derived from the encoder output h, rather than the reconstructed output x'.

This distinction is critical, as the encoder output captures the compressed feature representation, while the reconstructed output reflects the model's attempt to reproduce the original input. Using the encoder output ensures that the learned representation retains meaningful structural information suitable for downstream tasks such as data augmentation and classification.

A separate autoencoder is trained for each omics data type, and the resulting latent representations are concatenated to form a unified shared latent space. This integrated representation enables the model to capture complementary information across multiple biological data sources.

4.4 Generative Models

This section presents generative modelling techniques for addressing class imbalance in multi-omics data. Building upon the latent representations learned via AE, GAN and CT-GAN are introduced to generate synthetic samples and improve classification performance.

Generative Adversarial Networks (GAN)

GAN was utilized to up-sample the minor class from the shared latent space samples. A GAN consists of two competing neural networks: a generator and a discriminator. The generator G maps random noise z sampled from a prior distribution $p_z(z)$ to synthetic samples $\hat{x}$:

$$\hat{x} = G(z).$$

and a discriminator D takes a sample x and outputs a probability $D(x)$ indicating the likelihood that x is a real sample. The generator network was constructed by first initializing a sequential model, that consists of

- Dense layer with 128 neurons.
- A ReLU activation function, which helps the model learn complex patterns in the data by allowing it to model non-linear relationships.
- Dense layer with a specific output dimension and a Sigmoid activation function is added to produce the final output, which generates synthetic samples.

A discriminator network was constructed using a sequential model, which consists of:

- Dense layer with 128 neurons and ReLU activation function to process the input data.
- Dense layer with a single neuron and Sigmoid activation function is added to classify the input samples as real or fake.

These networks are key components of the GAN framework, where the generator creates synthetic samples, and the discriminator tries to differentiate between real and fake data.

By fostering a competitive learning process, GAN boosts the generator's ability to produce alike realistic data. The generator network is defined as follows:

$$G(z) = \sigma(\phi(W_g z + b_g)), \tag{4.4}$$

where $G(z)$ represents the synthetic sample generated by the generator, z is the input noise vector, W_g and b_g are the weights and biases of the generator's dense layers, ReLU (ϕ) is the Rectified Linear Unit activation function, which is defined as $\phi(x) = \max(0, x)$, helping the model learn nonlinear patterns, and Sigmoid (σ) is the Sigmoid activation function, which squashes the output values between 0 and 1, suitable for generating data samples. For the discriminator network,

$$D(x) = \sigma(\phi(W_d x + b_d)). \tag{4.5}$$

The objective of the GAN is to train the generator to produce samples that are indistinguishable from real samples and to train the discriminator to distinguish between real and fake samples. This adversarial training process can be formulated as a minimax game:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))], \tag{4.6}$$

where $\mathbb{E}$ denotes the expectation over real data x and noise z, $p_{\text{data}}(x)$ is the distribution of real data, and $p_z(z)$ is the prior distribution of noise.

Training Process

During training, the discriminator and generator were alternatively updated as the following:

- Discriminator Training: The discriminator is trained to maximize its ability to differentiate between real and fake samples. It was trained on both real samples x and generated samples $\hat{x}$, and its loss is calculated using binary cross-entropy:

$$\mathcal{L}_{\mathrm{D}} = -\left(\log D(x) + \log(1 - D(\hat{x}))\right). \tag{4.7}$$

- Generator Training: The generator is trained to minimize the discriminator's ability to distinguish between real and fake samples. Its loss is calculated using the discriminator's output when fed with generated samples:

$$\mathcal{L}_{\mathrm{G}} = -\log(D(G(z))). \tag{4.8}$$

Convergence in GAN is achieved when the generator produces data that the discriminator cannot distinguish from real data. This is theoretically when the Jensen–Shannon divergence between the real data distribution p_{data} and the generated data distribution p_G is minimized:

$$D_{\mathrm{JS}}(p_{\mathrm{data}} \| p_G) = \frac{1}{2}\left(D_{\mathrm{KL}}(p_{\mathrm{data}} \| M) + D_{\mathrm{KL}}(p_G \| M)\right), \tag{4.9}$$

where $M = \frac{1}{2}(p_{\mathrm{data}} + p_G)$, and D_{KL} is the Kullback–Leibler divergence.

Conditional Tabular Generative Adversarial Network (CTGAN)

CTGAN is designed to generate synthetic tabular data by addressing the challenges of imbalanced data and mixed data types. The generator G maps random noise z and conditional

vector c sampled from a prior distribution $p_z(z)$ and $p_c(c)$ to synthetic samples $\hat{x}$: $\hat{x} = G(z, c)$. The discriminator D takes a sample x and outputs a probability $D(x)$ indicating the likelihood that x is a real sample: $[D(x)]$.

The generator network is constructed by first initializing a sequential model that consists of the following:

- Dense layer with 256 neurons.
- A Batch Normalization layer to stabilize and accelerate training.
- A Leaky ReLU activation function, which helps the model learn complex patterns in the data by allowing it to model nonlinear relationships.
- Dense layer with a specific output dimension and a Tanh activation function is added to produce the final output, which generates synthetic samples.

The discriminator network is constructed using a sequential model, that consists of:

- Dense layer with 256 neurons and Leaky ReLU activation function to process the input data.
- Dense layer with a single neuron and Sigmoid activation function is added to classify the input samples as real or fake.

These networks are key components of the CTGAN framework, where the generator creates synthetic samples and the discriminator tries to differentiate between real and fake data. By fostering a competitive learning process, CTGAN boosts the generator's ability to produce realistic tabular data. The generator network is defined as follows:

$$G(z, c) = \tanh(\phi(W_g[z, c] + b_g)), \tag{4.10}$$

where $G(z, c)$ represents the synthetic sample generated by the generator, z is the input noise vector, c is the conditional vector, W_g and b_g are the weights and biases of the generator's dense layers, Leaky ReLU ϕ is the Leaky Rectified Linear Unit activation function, which is defined as $\phi(x) = \max(\alpha.x, x)$, helping the model learn nonlinear patterns with α is a tiny number and Tanh(tanh) is the Tanh activation function, which squashes the output values between -1 and 1, suitable for generating data samples.

For the discriminator network,

$$D(x) = \sigma(\phi(W_d x + b_d)). \tag{4.11}$$

The objective of the CTGAN is to train the generator to produce samples that are indistinguishable from real samples and to train the discriminator to distinguish between real and fake samples. This adversarial training process can be formulated as a minimax game:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z), c \sim p_c(c)}[\log(1 - D(G(z, c)))], \tag{4.12}$$

where $\mathbb{E}$ denotes the expectation over real data x, noise z, and conditional vector c, $p_{\text{data}}(x)$ is the distribution of real data, $p_z(z)$ is the prior distribution of noise, and $p_c(c)$ is the prior distribution of the conditional vector.

Training Process

During training, the discriminator and generator are alternatively updated as the following:

- Discriminator Training: The discriminator is trained to maximize its ability to differentiate between real and fake samples. It is trained on both real samples x and generated samples $\hat{x}$, and its loss is calculated using binary cross-entropy:

$$\mathcal{L}\mathrm{D} = -\left(\log D(x) + \log(1 - D(\hat{x}))\right). \tag{4.13}$$

- Generator Training: The generator is trained to minimize the discriminator's ability to distinguish between real and fake samples. Its loss is calculated using the discriminator's output when fed with generated samples:

$$\mathcal{L}\mathrm{G} = -\log(D(G(z, c))). \tag{4.14}$$

CTGAN address the unique challenges of tabular data using techniques including conditional sampling and balancing utility against disclosure risk. This results in more stable and efficient convergence compared to traditional GANs. The convergence can be analyzed using the conditional Jensen–Shannon divergence:

$$D_{\mathrm{JS}}(p_{\mathrm{data}}(\mathbf{x} \mid \mathbf{c}) \parallel p_G(\mathbf{x} \mid \mathbf{c})) = \frac{1}{2}\Big(D_{\mathrm{KL}}(p_{\mathrm{data}}(\mathbf{x} \mid \mathbf{c}) \parallel M(\mathbf{x} \mid \mathbf{c})) + D_{\mathrm{KL}}(p_G(\mathbf{x} \mid \mathbf{c}) \parallel M(\mathbf{x} \mid \mathbf{c}))\Big) \tag{4.15}$$

where

$$M(\mathbf{x} \mid \mathbf{c}) = \frac{1}{2}\left(p_{\mathrm{data}}(\mathbf{x} \mid \mathbf{c}) + p_G(\mathbf{x} \mid \mathbf{c})\right).$$

Prediction Model

Firstly, we split the data into training and testing sets using 5-fold cross-validation, which was also used to construct the prediction model. A neural network was built with two dense layers. The first has 128 neurons with the ReLU activation function, and the other has 1 neuron with the Sigmoid activation function. The model utilizes the Adam optimizer [107] and the binary cross-entropy loss function to optimize accuracy. The model was trained with parameters specifying 10 training epochs, a batch size of 32, and a validation split of 0.2. The Sigmoid function returns the probability of the class, which represents the predicted class. To avoid overfitting and underfitting, the hyper-parameters were optimized, and training and validation performance were plotted to monitor model performance. Early stoppage technique was used to stop earlier than the point where the training performance trends are different than the validation performance of the models.

4.5 Novelty of the Proposed Approach

To the best of our knowledge, this work is the first to systematically apply CTGAN in an autoencoder-derived latent space for multi-omics data imbalance handling and classification.

Unlike traditional approaches that perform oversampling in the original feature space, the proposed method leverages nonlinear representation learning to create a structured latent space, followed by conditional generative modelling to improve class balance.

This integration enables more effective data augmentation while preserving the biological relevance of the features.

4.6 Experimental Setup and Results

The generator and discriminator networks were implemented using fully connected neural network architectures. The models were trained using the Adam optimizer with binary cross-entropy loss.

Training was performed on the latent-space representations. Hyper-parameters such as batch size, number of epochs, and learning rates were tuned empirically.

The model hyper-parameters were empirically optimized to improve performance while reducing the risk of overfitting or underfitting, as summarized in Table 4.2.

Table 4.2: Models and hyper-parameters used in the experiments.

Model	**hyper-parameters and Values**
Omics Autoencoders	epochs: 30; batch size: 64; latent space dimension: 128
GAN and CTGAN	epochs: 300; batch size: 500; learning rate (α): 0.01
Prediction Model (Neural Network)	hidden layer units: 128; activation function: ReLU; dropout rate: 0.2; output layer units: 1; optimizer: Adam; loss function: binary cross-entropy; epochs: 30 (maximum, limited by early stopping); batch size: 32; early stopping patience: 5
k-Fold Cross-Validation	number of splits (n_{splits}): 5

Table 4.3 presents the performance measurements of applying AE with CTGAN and AE with GAN for the BLCA dataset. The results demonstrate that AE with CTGAN significantly outperforms AE with GAN across all metrics. AE with CTGAN achieves an average accuracy of 0.9929, a perfect precision of 1.0000, recall of 0.9846, and an average F1-score of 0.9922. In contrast, AE with GAN achieves an average accuracy of 0.88827, with a precision of 0.85417, recall of 0.75926, and an average F1-score of 0.80392.

Table 4.4 presents the performance measurements of applying AE with CTGAN and AE with GAN for the BRCA dataset. The results show that both models achieve high accuracy, withAE with CTGAN demonstrating a better average accuracy of 0.9748 compared to AE with GAN with accuracy of 0.9509. Both models exhibit perfect precision (1.0000). AE with CTGAN achieves a recall of 0.9777, while AE with GAN achieves a recall of 0.81481, suggesting that AE with CTGAN is better at identifying all true positive instances. In terms of the average F1-score, AE with CTGAN achieves 0.9922, further supporting the superior performance of AE with CTGAN in capturing both precision and recall.

From both Tables 4.3 and 4.4, the strength of AE with CTGAN in precision in both cancer outcomes suggests that all predicted positive instances are indeed positive. Generally, predicting the positive class, often representing adverse events including relapse [108] or metastasis [109], is a key focus in cancer outcome prediction due to its significant clinical implications. The overall performance suggests that AE with CTGAN is a more effective model for generating high-quality synthetic data for the BLCA dataset.

Table 4.3: Performance measurements for AE with CTGAN and AE with GAN on the BLCA dataset.

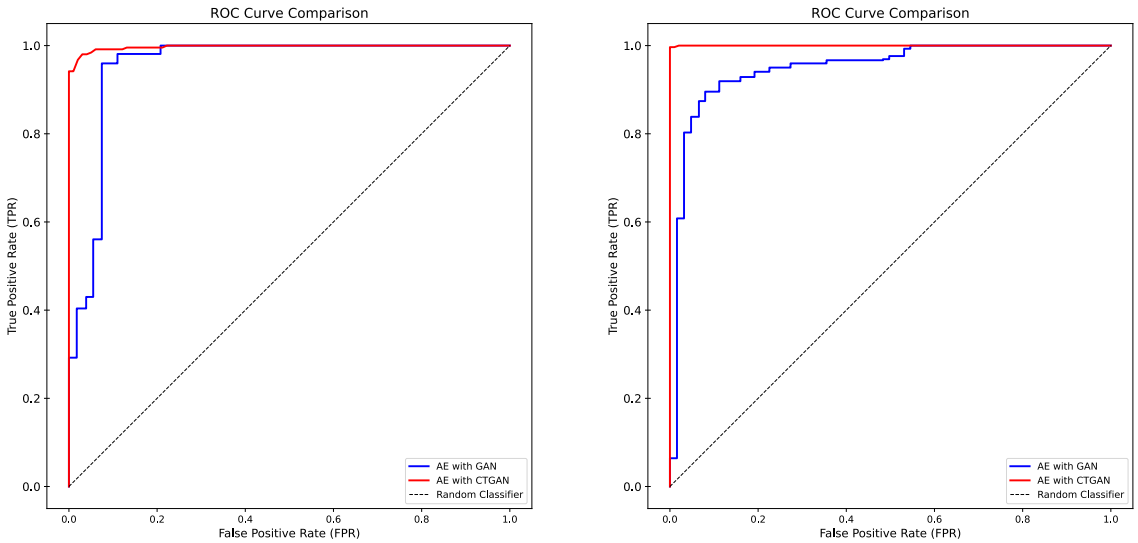
Classifier	**Average Accuracy**	**Precision**	**Recall**	**Average F1-Score**
AE with GAN	0.8882	0.8541	0.7592	0.8039
AE with CTGAN	0.9929	1.0000	0.9846	0.9922

Table 4.4: Performance measurements for AE with CTGAN and AE with GAN on the BRCA dataset.

Classifier	**Average Accuracy**	**Precision**	**Recall**	**Average F1-Score**
AE-GAN	0.9509	1.0000	0.8148	0.8979
AE-CTGAN	0.9748	1.0000	0.9777	0.9922

Figure 4.3a and 4.3b show the AUROC curves for applying AE with GAN versus AE with CTGAN models on both BLCA and BRCA datasets. The comparison shows that AE with CTGAN curves tend more to the north-west, demonstrates a larger area under the curve (AUC) across multiple validation folds. This indicates a dominance in better determining true positive rates while minimizing false positives. Consequently, AE with CTGAN demonstrates superior performance in distinguishing between classes, making it a more effective model for both datasets.

To validate the best performing model that is AE with CTGAN, the training versus validation loss for the model on the BLCA dataset as seen in Figure 4.4b and on the BRCA



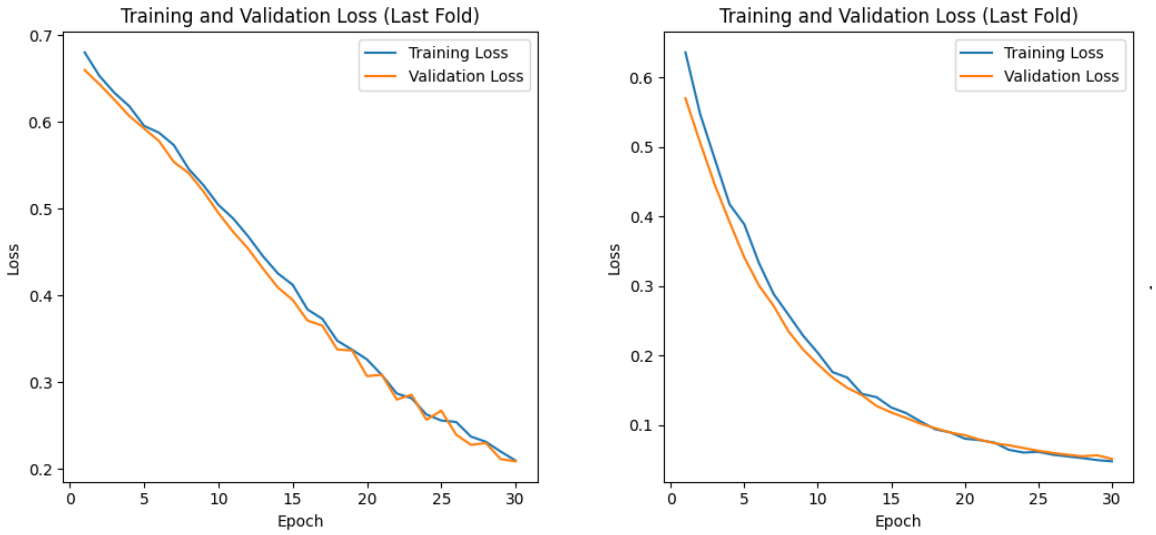
(a) BRCA dataset (b) BLCA dataset

Figure 4.3: Comparison of ROC curves for the BRCA and BLCA datasets.

dataset as seen in Figure 4.4a. The figures show a clear downward trend in both training and validation loss, which is generally a positive sign. It indicates that the model is learning and improving its performance over time. The similar trends for both training and validation curves in both figures rule out overfitting. The gap between the training and validation loss is relatively small, especially in the initial epochs. This suggests that the model is generalizing well to unseen data and not overfitting significantly in the early stages.

To evaluate the quality of the synthetic data beyond its predictive utility, we conducted a comparative fidelity analysis between the standard GAN and the CTGAN. In this context, fidelity was quantified by calculating the average Euclidean distance between the generated synthetic samples and their nearest neighbors in the original latent space. A lower Euclidean distance signifies that the synthetic data points are more representative of the real data distribution, maintaining the structural integrity of the original multi-omics features.

As shown in Table 4.5, the CTGAN architecture demonstrated a significantly higher



(a) BRCA dataset

(b) BLCA dataset

Figure 4.4: Training and validation loss curves for AE with CTGAN on the BRCA and BLCA datasets.

degree of fidelity compared to the standard GAN. Specifically, for the BLCA dataset, CTGAN reduced the average Euclidean distance by approximately 72%. For the BRCA dataset, the performance gap was even more substantial, with CTGAN achieving a distance of 0.5013, outperforming the standard GAN by a factor of six.

These findings suggest that the specialized components of CTGAN, such as mode-specific normalization and the conditional generator are better equipped to navigate the non-Gaussian and sparse distributions typically found in multi-omics latent representations. The high fidelity of the CTGAN-generated samples ensures that the synthetic data used for class balancing is not merely "noise" but a high-quality approximation of the minority class. This structural similarity to real data provides a robust foundation for the downstream classifier, explaining the high precision (1.000 for the positive class) achieved in our final predictive models.

Table 4.5: Fidelity comparison: average Euclidean distance between real and synthetic samples in the latent space.

Generative Model	**BLCA (TMB)**	**BRCA (Menopausal Status)**
AE with GAN	3.9600	3.1112
AE with CTGAN	1.0950	0.5013

4.7 Discussion

Figure 4.5 illustrates the analysis of latent space of AE in the BRCA data set, comprising two components: Figure 4.5A Correlation Heatmap displays pairwise correlations between 45 latent features (indices 0–44) in the bottleneck layer. The color gradient (from -1 to $+1$) reveals feature dependencies, where red indicates strong positive correlations, blue indicates negative correlations, and white denotes independence. Block-like patterns suggest clustered feature interactions, while sparse correlations imply disentangled representations. Figure 4.5B Feature Importance ranks latent features by their contribution (weight magnitude) to reconstruction. Dominant features (peaks) encode critical data patterns, while low-weight features may represent noise or redundant information. This analysis validates the AE's ability to compress input data into meaningful, non-redundant latent dimensions.

The proposed model demonstrates superior performance when compared with previously reported machine learning approaches in the literature. For bladder cancer (BLCA), the NMF-guided feature selection and genetic algorithm-based framework utilizing a convolutional neural network proposed by Al-Ghafer et al. [110] achieved a precision of 0.7789,

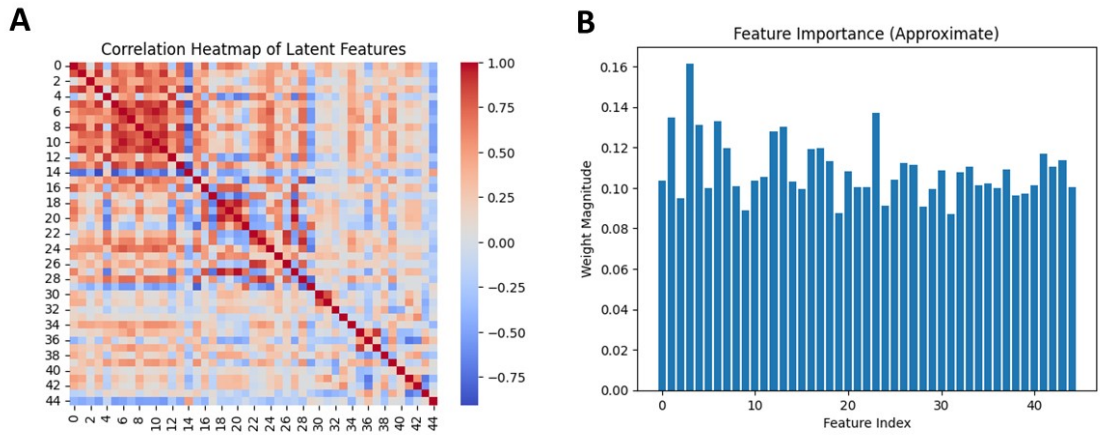


Figure 4.5: Applying AE to the BRCA dataset: (**A**) Correlation heatmap of latent features in the AE's bottleneck layer (scale: -1 to $+1$). (**B**) Approximate feature importance ranking based on weight magnitudes for the extracted latent features.

whereas the proposed model attained a precision of 1.0000. Similarly, for breast cancer (BRCA), the Support Vector Machine with radial basis function (RBF) kernel trained on SMOTE-synthesized samples reported by Alghanim et al. [2] achieved a precision of 0.919, compared to a precision of 1.0000 obtained by the proposed model.

GANs were initially developed to generate images and other forms of continuous data. Their architectures and training strategies are often optimized for these types of data [111]. CTGAN is designed to generate synthetic tabular data, an improvement over standard GANs [16]. CTGAN typically uses different network architectures for the generator and discriminator compared to standard GANs. These architectures are often tailored to the specific characteristics of tabular data, such as the presence of categorical features and the need to capture complex relationships between variables [16].

Both BLCA and BRCA datasets contain three omics, but applying this model to datasets with more or fewer omics is not restricted, as each omic will undergo the same workflow before integration. Meanwhile, AE with CTGAN demonstrated promising results in handling

multi-omics data by extracting the latent space and generating synthetic data.

Several limitations warrant consideration. First, the interpretability of the model is limited. Understanding the specific features driving the model's predictions and gaining deeper biological insights remains challenging. Secondly, the model's generalizability needs further validation. Applying the model to predict various cancer outcomes and other diseases would provide a more comprehensive assessment of its performance and robustness. Finally, while the model outperforms previous models in the literature [2, 110], it still needs rigorous comparisons with deep learning baseline models.

The correlation among the extracted latent space features from the BRCA dataset that can be seen in Figure 4.5A suggests that most of these features are uncorrelated; however, some dark-red areas suggest that some of these captured latent space features can be removed to extract only the unique latent features. While the proposed model works near perfection on both datasets, the strongly correlated latent features may affect the performance on more complex data. Addressing these limitations is crucial for future work. Techniques such as feature importance analysis and model-agnostic interpretation methods could be explored to enhance interpretability. Additionally, applying the model to diverse cancer datasets with different characteristics will broaden its applicability and provide valuable insights into its strengths and weaknesses. Furthermore, the potential of this approach can be explored beyond the omics data. Investigating its performance on other types of tabular multi-modal data domains could reveal its broader applicability and extraction power in diverse fields.

Chapter 5

Conclusion and Future Work

5.1 Overview of the Research

This dissertation addressed two fundamental, intertwined challenges in multi-omics cancer outcome prediction: the "curse of dimensionality" arising from high-throughput data and the degradation of model performance due to severe class imbalance. While linear methods like PCA and traditional oversampling techniques such as SMOTE offer baseline solutions, they fail to capture the nonlinear biological relationships across omics layers and often introduce noise when generating synthetic samples.

To overcome these limitations, this work developed and progressively refined a unified deep learning framework. The final architecture integrates an AE for nonlinear latent feature extraction with a CTGAN for targeted minority-class augmentation. By shifting the generative process from the original high-dimensional feature space to the compact, information-rich latent space, the framework preserves biological fidelity while effectively rebalancing class distributions. The methodology was rigorously validated on two distinct

TCGA datasets: breast cancer BRCA for menopausal status prediction and bladder cancer BLCA for TMB classification.

5.2 Summary of Key Findings and Contributions

This research makes three core contributions that advance multi-omics analysis:

5.2.1 Demonstrating the Effectiveness of Latent-Space Representation Learning

This work highlights the advantage of learning representations in a latent space derived from autoencoders when modelling complex multi-omics data. Unlike linear dimensionality reduction techniques such as PCA, which rely on orthogonal projections to capture global variance, the AE framework enables nonlinear transformations that better capture the underlying structure of the data.

5.2.2 Establishing CTGAN as the Preferred Generative Model for Tabular Omics Data

The systematic comparison between standard GAN and CTGAN revealed that conditional generation is critical for imbalanced multi-omics data. CTGAN's ability to target specific minority classes resulted in near-perfect precision (1.00) for both cancer types, whereas the standard GAN showed significant recall degradation (e.g., recall dropped to 0.76 for BLCA). This confirms that unconditional GANs struggle with the sparse, non-Gaussian distributions

typical of omics data.

Fidelity analysis showed that CTGAN reduced the average Euclidean distance between real and synthetic samples by up to 84% compared to standard GANs, indicating that latent-space generation produces structurally authentic samples that do not distort the underlying biological signal.

5.2.3 Proposing a Generalizable Framework for Cancer Outcome Prediction

The proposed AE-CTGAN pipeline achieved robust performance across two biologically distinct cancer types and prediction tasks: for BLCA (TMB classification), accuracy reached 0.9929, and for BRCA (menopausal status), accuracy reached 0.9748. This cross-dataset consistency demonstrates that the framework is not overfitted to a specific cancer type but captures generalizable principles of multi-omics integration.

5.3 Critical Appraisal of Limitations

Despite its strong performance, the framework has several limitations that temper its current applicability:

- **Limited Interpretability:** The model operates as a "black box." While the autoencoder compresses data into a latent space, the biological meaning of individual latent dimensions remains opaque. This hinders the discovery of mechanistic biomarkers, a key goal of multi-omics research.

- **Computational Cost:** Hyperparameter tuning for the combined AE-CTGAN architecture is computationally intensive. The current implementation requires separate training of the autoencoder, the CTGAN, and the downstream classifier, which may limit scalability to datasets with thousands of samples.

- **Binary-Only Focus:** The current formulation is restricted to binary classification. Many clinically relevant tasks (e.g., cancer subtyping, grade scoring) involve multiple classes, which this framework has not yet addressed.

- **Validation Scope:** Performance was evaluated using cross-validation on TCGA data. Prospective validation using independent clinical cohorts and wet-lab experiments is necessary to assess generalizability and rule out hidden batch effects or overfitting.

5.4 Future Research Directions

Addressing the above limitations defines a clear path forward:

- **Enhancing Interpretability.** Future work should integrate attention mechanisms or XAI techniques directly into the autoencoder or the generator. For instance, a transformer-based encoder could provide feature-attention maps, revealing which original genes contribute most to specific latent dimensions.

- **Extending to Multi-Class and Longitudinal Data.** The framework can be adapted for multi-class problems using conditional generators with multiple categorical conditions. Furthermore, extending it to longitudinal omics data (time-series) would enable modelling of tumour evolution and treatment response.

- **Exploring Advanced Generative Architectures:** While CTGAN performs well, newer generative approaches diffusion models like Denoising Diffusion Probabilistic Models (DDPM) may offer advantages in sample quality and training stability. A comparative study between latent-space CTGAN and latent-space diffusion models would be valuable insight into the effectiveness of alternative generative frameworks for multi-omics data augmentation.

- **Beyond Oncology.** The proposed framework is modality-agnostic, requiring only tabular data with class imbalance. It holds promise for other biomedical domains, including pharmacogenomics (predicting rare adverse drug reactions), neurodegenerative disease profiling (e.g., Alzheimer's staging), and rare disease diagnosis.

5.5 Final Remarks

This dissertation demonstrates that the strategic integration of nonlinear representation learning and conditional generative modelling offers a powerful solution to the enduring challenges of high dimensionality and class imbalance in multi-omics data. The AE-CTGANs framework consistently outperforms both linear baselines and standard GANs by preserving biological fidelity while effectively rebalancing class distributions.

The results demonstrate that latent space enhancement is a powerful paradigm for handling imbalanced datasets, and the proposed AE-CTGAN framework provides a strong foundation for future research in this domain. Beyond its technical contributions, this work provides a practical, scalable pipeline for precision medicine, enabling more accurate identification of high-risk patient subgroups. The framework achieves average accuracies

of 0.9929 and 0.9748, recall values of 0.9846 and 0.9777, and an F1-score of 0.9922 on the bladder and breast cancer datasets, respectively.

A key contribution of our model is the incorporation of CTGAN to generate synthetic data. Unlike standard GANs, CTGAN leverages conditional generation, enabling it to target and augment minority-class samples. This capability effectively mitigates class imbalance, a critical challenge in cancer datasets, and improves recall and F1-score for minority-class predictions.

Furthermore, our approach focuses on clinically meaningful outcomes such as TMB and menopausal status, ensuring their relevance and applicability to precision medicine, where these predicted outcomes can directly support patient stratification and risk assessment. The model can be used in academic medical centers and research hospitals to analyze multi-omics data, explore biomarker associations, and evaluate predictive models before clinical deployment. The model can also be applied to different multi-modal datasets in biomedical analysis and other fields. This sets our work apart from existing studies that often emphasize only technical model improvements without addressing specific clinical needs.

Overall, this study shows that combining AE and CTGAN not only addresses class imbalance and high-dimensional data challenges but also enhances predictive capabilities in cancer outcome prediction, contributing to the advancement of personalized healthcare.

Appendix A

Representative Implementation of the Proposed Framework

This appendix provides representative implementations of the key methods used in this dissertation, including linear oversampling techniques (SMOTE and ADASYN), standard GAN, and the proposed CTGAN-based approach. All implementations are presented for reproducibility. The complete pipeline is based on Python with scikit-learn, TensorFlow/Keras, and the Synthetic Data Vault (SDV) library.

A.1 Required Libraries

Listing A.1: Required library imports label

```
import numpy as np
import pandas as pd
from sklearn.utils import resample
from sklearn.metrics import accuracy_score, roc_auc_score, roc_curve,
    auc, confusion_matrix
from sklearn.model_selection import KFold, train_test_split
from sklearn.preprocessing import StandardScaler
import tensorflow as tf
from tensorflow.keras import layers, models
```

```
from tensorflow.keras.callbacks import EarlyStopping
import matplotlib.pyplot as plt

# Imbalanced-learn for SMOTE and ADASYN
from imblearn.over_sampling import SMOTE, ADASYN

# SDV library for CTGAN
!pip install sdv
from sdv.single_table import CTGANSynthesizer
from sdv.metadata import SingleTableMetadata
```

A.2 Data Loading and Preprocessing

Multi-omics data (gene expression, DNA methylation, and GAN) are loaded from Comma-Separated Values (CSV) files. Only samples with complete data across all three omics layers are retained.

Listing A.2: Loading and preprocessing multi-omics data

```
]
# Load multi-omics data
rna_data = pd.read_csv('mGE.csv')      # Gene expression
dna_data = pd.read_csv('mDM.csv')      # DNA methylation
cna_data = pd.read_csv('mCNA.csv')     # Copy number alteration

# Find common sample IDs across all three omics types
common_ids = set(rna_data['SAMPLE_ID']).intersection(
    dna_data['SAMPLE_ID'], cna_data['SAMPLE_ID']
)
```

```
11
12 # Filter each dataset to include only common samples
13 rna_data = rna_data[rna_data['SAMPLE_ID'].isin(common_ids)]
14 dna_data = dna_data[dna_data['SAMPLE_ID'].isin(common_ids)]
15 cna_data = cna_data[cna_data['SAMPLE_ID'].isin(common_ids)]
16
17 # Extract feature matrices (exclude the SAMPLE_ID and CLASS columns)
18 X_train_rna = rna_data.iloc[:, 1:].values
19 X_train_dna = dna_data.iloc[:, 1:].values
20 X_train_cna = cna_data.iloc[:, 1:].values
21
22 # Extract class labels (assumed to be in the 'CLASS' column of RNA
       dataset)
23 y_train = rna_data['CLASS'].values
24
25 # Optional: Standardize features
26 scaler = StandardScaler()
27 X_train_rna = scaler.fit_transform(X_train_rna)
28 X_train_dna = scaler.fit_transform(X_train_dna)
29 X_train_cna = scaler.fit_transform(X_train_cna)
```

A.3 Autoencoder for Latent Space Extraction

A separate AE is trained for each omics data type. Each autoencoder compresses high-dimensional features into a 128-dimensional latent representation. This is used as the foundation for all generative methods (GAN and CTGAN).

Listing A.3: Building and training autoencoders

```
def build_autoencoder(input_dim):
    """Build a simple autoencoder with one hidden layer."""
    model = models.Sequential()
    model.add(layers.Dense(128, activation='relu', input_dim=input_dim))
    model.add(layers.Dense(input_dim, activation='sigmoid'))
    model.compile(optimizer='adam', loss='mse')
    return model

# Train autoencoder for gene expression data
autoencoder_rna = build_autoencoder(X_train_rna.shape[1])
autoencoder_rna.fit(X_train_rna, X_train_rna,
                    epochs=30, batch_size=64, validation_split=0.2,
    verbose=0)

# Train autoencoder for DNA methylation data
autoencoder_dna = build_autoencoder(X_train_dna.shape[1])
autoencoder_dna.fit(X_train_dna, X_train_dna,
                    epochs=30, batch_size=64, validation_split=0.2,
    verbose=0)

# Train autoencoder for CNA data
autoencoder_cna = build_autoencoder(X_train_cna.shape[1])
autoencoder_cna.fit(X_train_cna, X_train_cna,
                    epochs=30, batch_size=64, validation_split=0.2,
    verbose=0)

```

```
# Extract latent space representations (encoder output)
latent_rna = autoencoder_rna.predict(X_train_rna)
latent_dna = autoencoder_dna.predict(X_train_dna)
latent_cna = autoencoder_cna.predict(X_train_cna)

# Concatenate latent spaces from all three omics types
latent_space = np.concatenate((latent_rna, latent_dna, latent_cna),
    axis=1)

print(f"Original feature dimension: {X_train_rna.shape[1] +
    X_train_dna.shape[1] + X_train_cna.shape[1]}")
print(f"Latent space dimension: {latent_space.shape[1]}")
```

A.4 Method 1: SMOTE (Synthetic Minority Over-sampling Technique)

SMOTE generates synthetic samples by interpolating between existing minority class samples in the feature space. This method was used in Chapter 3 as a baseline.

Listing A.4: SMOTE implementation

```
from imblearn.over_sampling import SMOTE

# Apply SMOTE to the latent space (or directly to original features)
smote = SMOTE(random_state=42)
X_smote, y_smote = smote.fit_resample(latent_space, y_train)

print(f"Original class distribution: {np.bincount(y_train)}")
print(f"SMOTE-balanced class distribution: {np.bincount(y_smote)}")
```

A.5 Method 2: ADASYN (Adaptive Synthetic Sampling)

ADASYN adaptively generates synthetic samples, focusing more on minority samples that are harder to learn (those near class boundaries). This method was also evaluated in Chapter 3.

Listing A.5: ADASYN implementation

```
from imblearn.over_sampling import ADASYN


# Apply ADASYN to the latent space
adasyn = ADASYN(random_state=42)
X_adasyn, y_adasyn = adasyn.fit_resample(latent_space, y_train)

print(f"Original class distribution: {np.bincount(y_train)}")
print(f"ADASYN-balanced class distribution: {np.bincount(y_adasyn)}")
```

A.6 Method 3: Standard GAN (Generative Adversarial Network)

A standard GAN is trained on the minority class samples in the latent space. The generator creates synthetic samples from random noise, while the discriminator tries to distinguish real from fake samples. This method was used as a baseline comparison in Chapter 4.

Listing A.6: Standard GAN implementation label

```
# Automatically determine the minority class
class_counts = np.unique(y_train, return_counts=True)
minority_class = class_counts[0][np.argmin(class_counts[1])]
```

```
4
5  # Separate majority and minority class samples in latent space
6  majority_samples = latent_space[y_train != minority_class]
7  minority_samples = latent_space[y_train == minority_class]
8
9  # GAN hyper-parameters
10 latent_dim = 100
11 batch_size = 64
12 epochs_gan = 200
13
14 # Build generator
15 def build_generator(latent_dim, output_dim):
16     model = models.Sequential()
17     model.add(layers.Dense(128, input_dim=latent_dim,
       activation='relu'))
18     model.add(layers.Dense(256, activation='relu'))
19     model.add(layers.Dense(output_dim, activation='sigmoid'))
20     return model
21
22 # Build discriminator
23 def build_discriminator(input_dim):
24     model = models.Sequential()
25     model.add(layers.Dense(256, input_dim=input_dim, activation='relu'))
26     model.add(layers.Dense(128, activation='relu'))
27     model.add(layers.Dense(1, activation='sigmoid'))
28     return model
29
```

```
# Compile discriminator
discriminator = build_discriminator(minority_samples.shape[1])
discriminator.compile(optimizer='adam', loss='binary_crossentropy',
    metrics=['accuracy'])

# Build and compile GAN
def build_gan(generator, discriminator):
    discriminator.trainable = False
    model = models.Sequential()
    model.add(generator)
    model.add(discriminator)
    return model

generator = build_generator(latent_dim, minority_samples.shape[1])
gan = build_gan(generator, discriminator)
gan.compile(optimizer='adam', loss='binary_crossentropy')

# Train GAN on minority class
def train_gan(generator, discriminator, gan, minority_samples,
              latent_dim, batch_size, epochs):
    for epoch in range(epochs):
        # Train discriminator
        idx = np.random.randint(0, minority_samples.shape[0],
    batch_size)
        real_samples = minority_samples[idx]
        real_labels = np.ones((batch_size, 1))

```

```
        noise = np.random.normal(0, 1, (batch_size, latent_dim))
        fake_samples = generator.predict(noise, verbose=0)
        fake_labels = np.zeros((batch_size, 1))

        d_loss_real = discriminator.train_on_batch(real_samples,
    real_labels)
        d_loss_fake = discriminator.train_on_batch(fake_samples,
    fake_labels)

        # Train generator
        noise = np.random.normal(0, 1, (batch_size, latent_dim))
        misleading_labels = np.ones((batch_size, 1))
        g_loss = gan.train_on_batch(noise, misleading_labels)

        if epoch % 50 == 0:
            print(f"Epoch {epoch}: D Loss Real: {d_loss_real[0]:.4f}, "
                  f"D Loss Fake: {d_loss_fake[0]:.4f}, G Loss:
    {g_loss:.4f}")

    return generator

# Train the GAN
generator = train_gan(generator, discriminator, gan, minority_samples,
                      latent_dim, batch_size, epochs_gan)

# Generate synthetic samples
num_synthetic = len(majority_samples)
```

```
noise = np.random.normal(0, 1, (num_synthetic, latent_dim))
synthetic_samples_gan = generator.predict(noise, verbose=0)

# Combine with original data
augmented_data_gan = np.vstack([latent_space, synthetic_samples_gan])
augmented_labels_gan = np.concatenate([
    np.zeros(len(y_train)),
    np.ones(num_synthetic)
])

# Shuffle
shuffle_idx = np.random.permutation(len(augmented_data_gan))
augmented_data_gan = augmented_data_gan[shuffle_idx]
augmented_labels_gan = augmented_labels_gan[shuffle_idx]

print(f"GAN-augmented dataset size: {len(augmented_data_gan)}")
```

A.7 Method 4: CTGAN (Conditional Tabular GAN) - Proposed Method

The CTGAN is trained on the minority class samples within the learned latent space. Unlike standard GAN, CTGAN uses conditional generation and mode-specific normalization, making it more suitable for tabular data. This is the proposed method from Chapter 4.

Listing A.7: CTGAN implementation using SDV library

```
# Prepare minority class data for CTGAN (as DataFrame)
minority_data_latent = pd.DataFrame(minority_samples)
```

```
3
4  # Create metadata for CTGAN
5  metadata = SingleTableMetadata()
6  metadata.detect_from_dataframe(data=minority_data_latent)
7
8  # Initialize and train CTGAN synthesizer
9  ctgan = CTGANSynthesizer(
10     metadata=metadata,
11     epochs=300,         # Number of training epochs
12     batch_size=500      # Batch size for training
13 )
14
15 ctgan.fit(minority_data_latent)
16
17 # Generate synthetic samples (match the number of majority samples)
18 num_synthetic_samples = len(majority_samples)
19 synthetic_data_ctgan = ctgan.sample(num_synthetic_samples)
20
21 # Combine original latent space with synthetic samples
22 augmented_data_ctgan = np.vstack([latent_space, synthetic_data_ctgan])
23
24 # Create labels: 0 for original samples, 1 for synthetic samples
25 augmented_labels_ctgan = np.concatenate([
26     np.zeros(len(y_train)),           # Original samples
27     np.ones(num_synthetic_samples)    # Synthetic samples
28 ])
29
```

```
# Shuffle the augmented dataset
shuffle_idx = np.random.permutation(len(augmented_data_ctgan))
augmented_data_ctgan = augmented_data_ctgan[shuffle_idx]
augmented_labels_ctgan = augmented_labels_ctgan[shuffle_idx]

print(f"CTGAN-augmented dataset size: {len(augmented_data_ctgan)}")
```

A.8 Cross-Validation and Neural Network Classification

The augmented datasets (from any method) are evaluated using 5-fold cross-validation. A simple neural network with dropout is used as the classifier.

Listing A.8: 5-fold cross-validation evaluation function. label

```
def evaluate_with_cv(augmented_data, augmented_labels, n_folds=5):
    """Evaluate augmented data using k-fold cross-validation."""
    kf = KFold(n_splits=n_folds, shuffle=True, random_state=42)

    results = {
        'accuracy': [],
        'auc_roc': [],
        'fpr': [],
        'tpr': []
    }

    for train_index, test_index in kf.split(augmented_data):
        X_train, X_test = augmented_data[train_index],
    augmented_data[test_index]
        y_train, y_test = augmented_labels[train_index],
```

```
    augmented_labels[test_index]

        # Build classifier
        model = models.Sequential()
        model.add(layers.Dense(128, activation='relu',
    input_dim=X_train.shape[1]))
        model.add(layers.Dropout(0.2))
        model.add(layers.Dense(1, activation='sigmoid'))
        model.compile(optimizer='adam', loss='binary_crossentropy',
    metrics=['accuracy'])

        # Early stopping
        early_stopping = EarlyStopping(monitor='val_loss', patience=5,
                                       restore_best_weights=True)

        # Train
        model.fit(X_train, y_train, epochs=30, batch_size=32,
                  validation_split=0.2, callbacks=[early_stopping],
    verbose=0)

        # Evaluate
        y_pred = model.predict(X_test, verbose=0)
        y_pred_binary = (y_pred > 0.5).astype(int)

        accuracy = accuracy_score(y_test, y_pred_binary)
        auc_roc = roc_auc_score(y_test, y_pred)
        fpr, tpr, _ = roc_curve(y_test, y_pred)
```

```
38
39          results['accuracy'].append(accuracy)
40          results['auc_roc'].append(auc_roc)
41          results['fpr'].append(fpr)
42          results['tpr'].append(tpr)
43
44      # Return average metrics
45      return {
46          'accuracy': np.mean(results['accuracy']),
47          'auc_roc': np.mean(results['auc_roc']),
48          'accuracy_std': np.std(results['accuracy']),
49          'auc_roc_std': np.std(results['auc_roc']),
50          'fpr_list': results['fpr'],
51          'tpr_list': results['tpr']
52      }
53
54  # Example: Evaluate CTGAN-augmented data
55  results_ctgan = evaluate_with_cv(augmented_data_ctgan,
        augmented_labels_ctgan)
56  print(f"CTGAN - Accuracy: {results_ctgan['accuracy']:.4f} (+/-
        {results_ctgan['accuracy_std']:.4f})")
57  print(f"CTGAN - AUC-ROC: {results_ctgan['auc_roc']:.4f} (+/-
        {results_ctgan['auc_roc_std']:.4f})")
```

A.9 Comparison of All Methods

The following code evaluates and compares SMOTE, ADASYN, GAN, and CTGAN using the same cross-validation pipeline.

Listing A.9: Comparative evaluation of all methods

```
# Dictionary to store results for each method
all_results = {}

# Method 1: SMOTE
X_smote, y_smote = SMOTE(random_state=42).fit_resample(latent_space,
    y_train)
all_results['SMOTE'] = evaluate_with_cv(X_smote, y_smote)

# Method 2: ADASYN
X_adasyn, y_adasyn = ADASYN(random_state=42).fit_resample(latent_space,
    y_train)
all_results['ADASYN'] = evaluate_with_cv(X_adasyn, y_adasyn)

# Method 3: Standard GAN (using previously augmented data)
all_results['GAN'] = evaluate_with_cv(augmented_data_gan,
    augmented_labels_gan)

# Method 4: CTGAN (proposed)
all_results['CTGAN'] = evaluate_with_cv(augmented_data_ctgan,
    augmented_labels_ctgan)

# Print comparison table
print("\n" + "="*60)
print("COMPARATIVE RESULTS ACROSS METHODS")
print("="*60)
print(f"{'Method':<10} {'Accuracy':<12} {'AUC-ROC':<12}")
```

```
print("-"*60)
for method, results in all_results.items():
    print(f"{method:<10} {results['accuracy']:.4f} +/-
    {results['accuracy_std']:.4f}  "
          f"{results['auc_roc']:.4f} +/- {results['auc_roc_std']:.4f}")
print("="*60)
```

A.10 Visualization: ROC Curves Comparison

The following code generates comparative ROC curves for all four methods.

Listing A.10: Comparative ROC curves

```
plt.figure(figsize=(10, 8))
colors = {'SMOTE': 'blue', 'ADASYN': 'green', 'GAN': 'orange', 'CTGAN':
    'red'}

for method, results in all_results.items():
    # Compute mean ROC curve
    all_fpr = np.unique(np.concatenate(results['fpr_list']))
    mean_tpr = np.zeros_like(all_fpr)
    for i in range(len(results['fpr_list'])):
        mean_tpr += np.interp(all_fpr, results['fpr_list'][i],
    results['tpr_list'][i])
    mean_tpr /= len(results['fpr_list'])

    # Add (0,0) point
    all_fpr = np.concatenate(([0], all_fpr))
    mean_tpr = np.concatenate(([0], mean_tpr))
```

```
    roc_auc = auc(all_fpr, mean_tpr)

    plt.plot(all_fpr, mean_tpr, color=colors[method], lw=2,
             label=f'{method} (AUC = {roc_auc:.3f})')

plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curves Comparison: SMOTE, ADASYN, GAN, and CTGAN')
plt.legend(loc='lower right')
plt.grid(True, linestyle='--', alpha=0.7)
plt.show()
```

A.11 Fidelity Analysis: Euclidean Distance

To evaluate the quality of synthetic samples, the average Euclidean distance between synthetic samples and their nearest neighbors in the real latent space is computed. A lower distance indicates higher fidelity.

Listing A.11: Fidelity analysis using Euclidean distance

```
from sklearn.neighbors import NearestNeighbors

def compute_fidelity(real_samples, synthetic_samples):
    """Compute average Euclidean distance from synthetic to nearest
    real sample."""
    nbrs = NearestNeighbors(n_neighbors=1, metric='euclidean')
```

```
    nbrs.fit(real_samples)
    distances, _ = nbrs.kneighbors(synthetic_samples)
    return np.mean(distances)

# Compute fidelity for each method
real_latent = latent_space

# For GAN
fidelity_gan = compute_fidelity(real_latent, synthetic_samples_gan)

# For CTGAN
fidelity_ctgan = compute_fidelity(real_latent,
    synthetic_data_ctgan.values)

print("\n" + "="*50)
print("FIDELITY ANALYSIS (Lower is Better)")
print("="*50)
print(f"Standard GAN  - Average Euclidean Distance: {fidelity_gan:.4f}")
print(f"CTGAN (Proposed) - Average Euclidean Distance:
    {fidelity_ctgan:.4f}")
print(f"Improvement: {(1 - fidelity_ctgan/fidelity_gan) * 100:.1f}%
    reduction")
print("="*50)
```

A.12 Code Availability

The complete implementation of all methods (SMOTE, ADASYN, standard GAN, and the proposed AE-CTGAN framework) is publicly available at:

https://github.com/Ibrahimalhurani/Dataset_and_Code

Appendix B

Software and Packages Used

This research was implemented using the following software tools, libraries, and computational environments:

Table B.1: Software tools, libraries, and computational environments

Category	**Tools and Libraries (with Citations)**
Programming Language	Python [112] - Used for model development, data processing, evaluation, and pipeline integration.
Deep Learning Framework	PyTorch [113] – Used to implement the transformer-based representation learning module and cGAN, as well as model training and tensor operations.
	Continued on next page

Table B.1: Software tools, libraries, and computational environments (continued)

Category	**Tools and Libraries (with Citations)**
Data Manipulation	NumPy [114] and Pandas [115] – Used for numerical computation, matrix manipulation, tabular data processing, and result aggregation.
Machine Learning Utilities	Scikit-learn [116] – Used for stratified cross-validation, preprocessing, baseline models, anomaly detection, and performance evaluation.
Baseline Models	Logistic Regression, Support Vector Machine, and Random Forest implemented through Scikit-learn [116] – Used as baseline or comparison models.
Data Balancing	Imbalanced-learn (SMOTE) [117, 7] – Applied for minority-class oversampling in baseline comparison settings.
Gradient Boosting Models	LightGBM [118] and XGBoost [93] – Used as additional comparison models in the classification experiments.

Continued on next page

Table B.1: Software tools, libraries, and computational environments (continued)

Category	**Tools and Libraries (with Citations)**
Visualization	Matplotlib [119] – Used for plotting ROC curves, precision–recall curves, confusion matrices, and other performance visualizations.
Runtime and Utilities	JSON, OS, Glob, Time, Datetime, Shutil, Random, and Warnings (Python standard library) – Used for file handling, reproducibility control, runtime management, and result organization.
Development Environment	Jupyter Notebook [120] and Google Colab [121] – Used for experimentation, interactive execution, and workflow prototyping.

Bibliography

[1] Q. Wang, W.-X. Peng, L. Wang, and L. Ye, "Toward multiomics-based next-generation diagnostics for precision medicine," *Personalized Medicine*, vol. 16, no. 2, pp. 157–170, 2019.

[2] F. Alghanim, I. Al-Hurani, H. Qattous, A. Al-Refai, O. Batiha, A. Alkhateeb, and S. Ikki, "Machine learning model for multiomics biomarkers identification for menopause status in breast cancer," *Algorithms*, vol. 17, no. 1, p. 13, 2023.

[3] H. Qattous, M. Azzeh, R. Ibrahim, I. A. Al-Ghafer, M. Al Sorkhy, and A. Alkhateeb, "Pacmap-embedded convolutional neural network for multi-omics data integration," *Heliyon*, vol. 10, no. 1, 2024.

[4] V. H. Do and S. Canzar, "A generalization of t-sne and umap to single-cell multimodal omics," *Genome Biology*, vol. 22, no. 1, p. 130, 2021.

[5] H. Huang, Y. Wang, C. Rudin, and E. P. Browne, "Towards a comprehensive evaluation of dimension reduction methods for transcriptomic data visualization," *Communications biology*, vol. 5, no. 1, p. 719, 2022.

[6] B. Doğan, "Cbmap: Clustering-based manifold approximation and projection for dimensionality reduction," *IEEE Access*, 2025.

[7] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[8] H. He, Y. Bai, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pp. 1322–1328, Ieee, 2008.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[10] W. Ågren and V. Ú. Sosa, "Hierarchical conditional tabular gan for multi-tabular synthetic data generation," *arXiv preprint arXiv:2411.07009*, 2024.

[11] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. Li, and Y. Xu, "Autoencoders and their applications in machine learning: a survey," *Artificial intelligence review*, vol. 57, no. 2, p. 28, 2024.

[12] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation, parallel distributed processing, explorations in the microstructure of cognition, ed. de rumelhart and j. mcclelland. vol. 1. 1986," *Biometrika*, vol. 71, pp. 599–607, 1986.

[13] F. Xhafa, *Machine Learning, Big Data, and IoT for Medical Informatics*. Academic Press, 2021.

[14] X. Zhang, Y. Xing, K. Sun, and Y. Guo, "Omiembed: a unified multi-task deep learning framework for multi-omics data," *Cancers*, vol. 13, no. 12, p. 3047, 2021.

[15] V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci, "Deep neural networks and tabular data: A survey," *IEEE transactions on neural networks and learning systems*, vol. 35, no. 6, pp. 7499–7519, 2022.

[16] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," *Advances in neural information processing systems*, vol. 32, 2019.

[17] K. J. Karczewski and M. P. Snyder, "Integrative omics for health and disease," *Nature Reviews Genetics*, vol. 19, no. 5, pp. 299–310, 2018.

[18] B. B. Misra, C. Langefeld, M. Olivier, and L. A. Cox, "Integrated omics: tools, advances and future approaches," *Journal of molecular endocrinology*, vol. 62, no. 1, pp. R21–R45, 2019.

[19] A. Mukherjee, S. Abraham, A. Singh, S. Balaji, and K. Mukunthan, "From data to cure: A comprehensive exploration of multi-omics data analysis for targeted therapies," *Molecular biotechnology*, vol. 67, no. 4, pp. 1269–1289, 2025.

[20] N. Rappoport and R. Shamir, "Multi-omic and multi-view clustering algorithms: review and cancer benchmark," *Nucleic acids research*, vol. 46, no. 20, pp. 10546–10562, 2018.

[21] K. A. Hoadley, C. Yau, T. Hinoue, D. M. Wolf, A. J. Lazar, E. Drill, R. Shen, A. M. Taylor, A. D. Cherniack, V. Thorsson, *et al.*, "Cell-of-origin patterns dominate the molecular classification of 10,000 tumors from 33 types of cancer," *Cell*, vol. 173, no. 2, pp. 291–304, 2018.

[22] K. Chaudhary, O. B. Poirion, L. Lu, and L. X. Garmire, "Deep learning–based multi-omics integration robustly predicts survival in liver cancer," *Clinical cancer research*, vol. 24, no. 6, pp. 1248–1259, 2018.

[23] V. Raufaste-Cazavieille, R. Santiago, and A. Droit, "Multi-omics analysis: Paving the path toward achieving precision medicine in cancer treatment and immuno-oncology," *Frontiers in Molecular Biosciences*, vol. 9, p. 962743, 2022.

[24] R. Beroukhim, C. H. Mermel, D. Porter, G. Wei, S. Raychaudhuri, J. Donovan, J. Barretina, J. S. Boehm, J. Dobson, M. Urashima, K. T. McHenry, R. M. Pinchback, K. L. Ligon, Y. J. Cho, L. Haery, H. Greulich, M. Reich, W. Winckler, M. S. Lawrence, B. A. Weir, K. E. Tanaka, D. Y. Chiang, A. J. Bass, A. Loo, C. Hoffman, J. Prensner, Q. Liefeld, Ted Gao, D. Yecies, S. Signoretti, E. Maher, F. J. Kaye, H. Sasaki, J. E. Tepper, C. D. M. Fletcher, J. Tabernero, J. Baselga, M.-S. Tsao, F. Demichelis, M. A. Rubin, P. A. Jänne, M. J. Daly, C. Nucera, R. L. Levine, B. L. Ebert, S. Gabriel, A. K. Rustgi, C. R. Antonescu, M. Ladanyi, A. Letai, L. A. Garraway, M. Loda, D. G. Beer, L. D. True, A. Okamoto, S. L. Pomeroy, S. Singer, T. R. Golub, E. S. Lander, G. Getz, W. R. Sellers, and M. Meyerson, "The landscape of somatic copy-number alteration across human cancers," *Nature*, vol. 463, no. 7283, pp. 899–905, 2010.

[25] S. Rasheed, J. S. Yan, A. Hussain, and B. Lai, "Proteomic characterization of HIV-modulated membrane receptors, kinases and signaling proteins involved in novel angiogenic pathways," *J. Transl. Med.*, vol. 7, p. 75, Aug. 2009.

[26] P. A. Jones, "Functions of DNA methylation: islands, start sites, gene bodies and beyond," *Nat. Rev. Genet.*, vol. 13, pp. 484–492, May 2012.

[27] T. Sørlie, C. M. Perou, R. Tibshirani, T. Aas, S. Geisler, H. Johnsen, T. Hastie, M. B. Eisen, M. van de Rijn, S. S. Jeffrey, T. Thorsen, H. Quist, J. C. Matese, P. O. Brown, D. Botstein, P. E. Lønning, and A. L. Børresen-Dale, "Gene expression patterns

of breast carcinomas distinguish tumor subclasses with clinical implications," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 98, pp. 10869–10874, Sept. 2001.

[28] A. Sathyanarayanan, R. Gupta, E. W. Thompson, D. R. Nyholt, D. C. Bauer, and S. H. Nagaraj, "A comparative study of multi-omics integration tools for cancer driver gene identification and tumour subtyping," *Briefings in bioinformatics*, vol. 21, no. 6, pp. 1920–1936, 2020.

[29] A. Alkhateeb, A. A. Tabl, and L. Rueda, "Deep learning in multi-omics data integration in cancer diagnostic," in *Deep learning for biomedical data analysis: Techniques, approaches, and applications*, pp. 255–271, Springer, 2021.

[30] J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, and J. M. Stuart, "The cancer genome atlas pan-cancer analysis project," *Nature genetics*, vol. 45, no. 10, pp. 1113–1120, 2013.

[31] R. M. Samstein, C.-H. Lee, A. N. Shoushtari, M. D. Hellmann, R. Shen, Y. Y. Janjigian, D. A. Barron, A. Zehir, E. J. Jordan, A. Omuro, *et al.*, "Tumor mutational load predicts survival after immunotherapy across multiple cancer types," *Nature genetics*, vol. 51, no. 2, pp. 202–206, 2019.

[32] K. A. Hoadley, C. Yau, T. Hinoue, D. M. Wolf, A. J. Lazar, E. Drill, R. Shen, A. M. Taylor, A. D. Cherniack, V. Thorsson, R. Akbani, R. Bowlby, C. K. Wong, M. Wiznerowicz, F. Sanchez-Vega, A. G. Robertson, B. G. Schneider, M. S. Lawrence, H. Noushmehr, T. M. Malta, Cancer Genome Atlas Network, J. M. Stuart, C. C. Benz, and P. W. Laird, "Cell-of-origin patterns dominate the molecular classification of 10,000 tumors from 33 types of cancer," *Cell*, vol. 173, pp. 291–304.e6, Apr. 2018.

[33] D. Hanahan and R. A. Weinberg, "Hallmarks of cancer: the next generation," *cell*, vol. 144, no. 5, pp. 646–674, 2011.

[34] Mayo Clinic and National Cancer Institute, "What does cancer look like?," 2023. Credit: Cecil Fox, National Cancer Institute, NIH.

[35] B. Vogelstein, N. Papadopoulos, V. E. Velculescu, S. Zhou, L. A. Diaz Jr, and K. W. Kinzler, "Cancer genome landscapes," *science*, vol. 339, no. 6127, pp. 1546–1558, 2013.

[36] Y. Hasin, M. Seldin, and A. Lusis, "Multi-omics approaches to disease," *Genome Biology*, vol. 18, no. 1, p. 83, 2017.

[37] M. D. Ritchie, E. R. Holzinger, R. Li, S. A. Pendergrass, and D. Kim, "Methods of integrating data to uncover genotype–phenotype interactions," *Nature Reviews Genetics*, vol. 16, no. 2, pp. 85–97, 2015.

[38] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal, and F. Bray, "Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide," *CA: A Cancer Journal for Clinicians*, vol. 71, no. 3, pp. 209–249, 2021.

[39] A. G. Rivenbark, S. M. O'Connor, and W. B. Coleman, "Molecular and cellular heterogeneity in breast cancer: Challenges for personalized medicine," *The American Journal of Pathology*, vol. 183, no. 4, pp. 1113–1124, 2013.

[40] J. Poorolajal, F. Heidarimoghis, M. Karami, Z. Cheraghi, F. Gohari-Ensaf, F. Shahbazi, B. Zareie, P. Ameri, and F. Sahraee, "Factors for the primary prevention of breast cancer: A meta-analysis of prospective cohort studies," *Journal of Research in Health Sciences*, vol. 21, pp. e00520 – e00520, 2021.

[41] E. Rakha, G. Tse, and C. Quinn, "An update on the pathological classification of breast cancer," *Histopathology*, vol. 82, pp. 5 – 16, 2022.

[42] C. K. Anders, R. Johnson, J. Litton, M. Phillips, and A. Bleyer, "Breast cancer before age 40 years," in *Seminars in oncology*, vol. 36, pp. 237–249, Elsevier, 2009.

[43] A. Vincent, "Management of menopause in women with breast cancer," *Climacteric*, vol. 18, no. 5, pp. 690–701, 2015.

[44] H. A. Azim and A. H. Partridge, "Biology of breast cancer in young women," *Breast cancer research*, vol. 16, no. 4, p. 427, 2014.

[45] A. Goldhirsch, E. P. Winer, A. Coates, R. Gelber, M. Piccart-Gebhart, B. Thürlimann, H.-J. Senn, K. S. Albain, F. André, J. Bergh, *et al.*, "Personalizing the treatment of women with early breast cancer: highlights of the st gallen international expert consensus on the primary therapy of early breast cancer 2013," *Annals of oncology*, vol. 24, no. 9, pp. 2206–2223, 2013.

[46] A. T. Lenis, P. M. Lec, K. Chamie, and M. Mshs, "Bladder cancer: a review," *Jama*, vol. 324, no. 19, pp. 1980–1991, 2020.

[47] A. M. Kamat, N. M. Hahn, J. A. Efstathiou, S. P. Lerner, P.-U. Malmström, W. Choi, C. C. Guo, Y. Lotan, and W. Kassouf, "Bladder cancer," *The Lancet*, vol. 388, no. 10061, pp. 2796–2810, 2016.

[48] S. Lindskrog, F. Prip, P. Lamy, A. Taber, C. Groeneveld, K. Birkenkamp-Demtröder, J. B. Jensen, T. Strandgaard, I. Nordentoft, E. Christensen, M. Sokač, N. Birkbak, L. Maretty, G. Hermann, A. Petersen, V. Weyerer, M. Grimm, M. Horstmann, G. Sjödahl, M. Höglund, T. Steiniche, K. Mogensen, A. de Reyniès, R. Nawroth, B. Jordan, X. Lin, D. Dragičević, D. Ward, A. Goel, C. Hurst, J. Raman, J. Warrick, U. Segersten, D. Sikic, K. V. van Kessel, T. Maurer, J. Meeks, D. DeGraff, R. Bryan, M. Knowles, T. Simić, A. Hartmann, E. Zwarthoff, P. Malmström, N. Malats, F. Real, and L. Dyrskjøt, "An integrated multi-omics analysis identifies prognostic molecular subtypes of non-muscle-invasive bladder cancer," *Nature Communications*, vol. 12, 2021.

[49] T. A. Chan, M. Yarchoan, E. Jaffee, C. Swanton, S. A. Quezada, A. Stenzinger, and S. Peters, "Development of tumor mutation burden as an immunotherapy biomarker: utility for the oncology clinic," *Annals of oncology*, vol. 30, no. 1, pp. 44–56, 2019.

[50] S. Mariathasan, S. J. Turley, D. Nickles, A. Castiglioni, K. Yuen, Y. Wang, E. E. Kadel Iii, H. Koeppen, J. L. Astarita, R. Cubas, *et al.*, "Tgfβ attenuates tumour response to pd-l1 blockade by contributing to exclusion of t cells," *Nature*, vol. 554, no. 7693, pp. 544–548, 2018.

[51] R. Cristescu, R. Mogg, M. Ayers, A. Albright, E. Murphy, J. Yearley, X. Sher, X. Q. Liu, H. Lu, M. Nebozhyn, *et al.*, "Pan-tumor genomic biomarkers for pd-1 checkpoint blockade–based immunotherapy," *Science*, vol. 362, no. 6411, p. eaar3593, 2018.

[52] A. E. Mohr, C. P. Ortega-Santos, C. M. Whisner, J. Klein-Seetharaman, and P. Jasbi, "Navigating challenges and opportunities in multi-omics integration for personalized healthcare," *Biomedicines*, vol. 12, no. 7, p. 1496, 2024.

[53] P. S. Reel, S. Reel, E. Pearson, E. Trucco, and E. Jefferson, "Using machine learning approaches for multi-omics data analysis: A review," *Biotechnology Advances*, vol. 49, p. 107739, 2021.

[54] W. Zhang, F. Li, and L. Nie, "Integrating multiple 'omics' analysis for microbial biology: application and methodologies," *Microbiology*, vol. 156, no. 2, pp. 287–301, 2010.

[55] E. Tasci, Y. Zhuge, K. Camphausen, and A. V. Krauze, "Bias and class imbalance in oncologic data—towards inclusive and transferrable ai in large scale oncology data sets," *Cancers*, vol. 14, no. 12, p. 2897, 2022.

[56] H. Yu, C. Sun, W. Yang, S. Xu, and Y. Dan, "A review of class imbalance learning methods in bioinformatics," *Current Bioinformatics*, vol. 10, no. 4, pp. 360–369, 2015.

[57] D. Acharya and A. Mukhopadhyay, "A comprehensive review of machine learning techniques for multi-omics data integration: challenges and applications in precision oncology," *Briefings in functional genomics*, vol. 23, no. 5, pp. 549–560, 2024.

[58] Madhumita and S. Paul, "Autoencoder assisted cancer subtyping by integrating multi-omics data," in *International Conference on Pattern Recognition and Machine Intelligence*, pp. 127–136, Springer, 2021.

[59] R. Correa-Aguila, N. Alonso-Pupo, and E. W. Hernandez-Rodriguez, "Multi-omics data integration approaches for precision oncology," *Molecular Omics*, vol. 18, no. 6, pp. 469–479, 2022.

[60] F. S. Bidabadi, M. Fahmy, and M. Hemati, "Machine learning, artificial intelligence and policy challenges in economic and health sciences," *Scientific Hypotheses*, 2025.

[61] M. Picard, M.-P. Scott-Boyer, A. Bodein, O. Périn, and A. Droit, "Integration strategies of multi-omics data for machine learning analysis," *Computational and Structural Biotechnology Journal*, vol. 19, pp. 3735–3746, 2021.

[62] M. W. Libbrecht and W. S. Noble, "Machine learning applications in genetics and genomics," *Nature Reviews Genetics*, vol. 16, no. 6, pp. 321–332, 2015.

[63] P. Larranaga, B. Calvo, R. Santana, C. Bielza, J. Galdiano, I. Inza, J. A. Lozano, R. Armananzas, G. Santafé, A. Pérez, *et al.*, "Machine learning in bioinformatics," *Briefings in bioinformatics*, vol. 7, no. 1, pp. 86–112, 2006.

[64] K. T. Ahmed, J. Sun, S. Cheng, J. Yong, and W. Zhang, "Multi-omics data integration by generative adversarial network," *Bioinformatics*, vol. 38, no. 1, pp. 179–186, 2022.

[65] O. Habibi, M. Chemmakha, and M. Lazaar, "Imbalanced tabular data modelization using ctgan and machine learning to improve iot botnet attacks detection," *Engineering Applications of Artificial Intelligence*, vol. 118, p. 105669, 2023.

[66] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, pp. 214–223, PMLR, 2017.

[67] X. Zhang, J. Zhang, K. Sun, X. Yang, C. Dai, and Y. Guo, "Integrated multi-omics analysis using variational autoencoders: application to pan-cancer classification," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 765–769, IEEE, 2019.

[68] Broad Institute, "Deep dive: Multi-omics variational autoencoders," 2025. Accessed: Jan. 2025.

[69] Y. Choi, R. Li, and G. Quon, "sivae: interpretable deep generative models for single-cell transcriptomes," *Genome biology*, vol. 24, no. 1, p. 29, 2023.

[70] H.-S. Lee, S.-H. Hong, G.-H. Kim, H.-J. You, E.-Y. Lee, J.-H. Jeong, J.-W. Ahn, and J.-H. Kim, "Generative models utilizing padding can efficiently integrate and generate multi-omics data," *AI*, vol. 5, no. 3, pp. 1614–1632, 2024.

[71] L. Xin, C. Huang, H. Li, S. Huang, Y. Feng, Z. Kong, Z. Liu, S. Li, C. Yu, F. Shen, *et al.*, "Artificial intelligence for central dogma-centric multi-omics: challenges and breakthroughs," *arXiv preprint arXiv:2412.12668*, 2024.

[72] J. Kim and J. Seok, "ctgan: combined transformation of gene expression and survival data with generative adversarial network," *Briefings in Bioinformatics*, vol. 25, no. 4, 2024.

[73] F. Alharbi, A. Vakanski, M. K. Elbashir, and M. Mohammed, "Lasso–mogat: a multi-omics graph attention framework for cancer classification," *Academia Biology*, vol. 2, no. 3, 2024.

[74] J. Nußberger, F. Boesel, S. Lenz, H. Binder, and M. Hess, "Synthetic observations from deep generative models and binary omics data with limited sample size," *Briefings in Bioinformatics*, vol. 22, no. 4, p. bbaa226, 2021.

[75] D. Sidorenko, S. Pushkov, A. Sakip, G. H. D. Leung, S. W. Y. Lok, A. Urban, D. Zagirova, A. Veviorskiy, N. Tihonova, A. Kalashnikov, *et al.*, "Precious2gpt: the combination of multiomics pretrained transformer and conditional diffusion for artificial multi-omics multi-species multi-tissue sample generation," *npj Aging*, vol. 10, no. 1, p. 37, 2024.

[76] P. A. Apellaniz, B. A. Galende, A. Jim, J. Parras, S. Zazo, *et al.*, "Advancing cancer research with synthetic data generation in low-data scenarios," *IEEE Journal of Biomedical and Health Informatics*, 2025.

[77] H. A. Ahmed, J. A. Nepomuceno, B. Vega-Márquez, and I. A. Nepomuceno-Chamorro, "Synthetic data generation for healthcare: exploring generative adversarial networks variants for medical tabular data," *International Journal of Data Science and Analytics*, vol. 20, no. 6, pp. 5739–5754, 2025.

[78] J. N. Weinstein *et al.*, "The cancer genome atlas pan-cancer analysis project," *Nature Genetics*, vol. 45, no. 10, pp. 1113–1120, 2013.

[79] N. Rappoport and R. Shamir, "Multi-omics data integration methods," *Bioinformatics*, vol. 34, no. 12, pp. i385–i393, 2018.

[80] R. Bellman, *Adaptive Control Processes*. Princeton University Press, 1961.

[81] A. K. Jain *et al.*, "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.

[82] E. A. Bruford *et al.*, "Guidelines for human gene nomenclature," *Nature Genetics*, vol. 52, pp. 754–758, 2020.

[83] M. S. Lawrence *et al.*, "Mutational heterogeneity in cancer and the search for new cancer-associated genes," *Nature*, vol. 499, no. 7457, pp. 214–218, 2013.

[84] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[85] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[86] E. Athieniti and G. M. Spyrou, "A guide to multi-omics data collection and integration for translational medicine," *Computational and structural biotechnology journal*, vol. 21, pp. 134–149, 2023.

[87] K. An, "Sulla determinazione empirica di una legge didistribuzione," *Giorn Dell'inst Ital Degli Att*, vol. 4, pp. 89–91, 1933.

[88] T. Bayes, "An essay towards solving a problem in the doctrine of chances," *Biometrika*, vol. 45, no. 3/4, pp. 296–315, 1958.

[89] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[90] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[91] J. Wang, Q. Chen, and Y. Chen, "Rbf kernel based support vector machine with universal approximation and its application," in *International symposium on neural networks*, pp. 512–517, Springer, 2004.

[92] L. S. Shapley, "Notes on the n-person game—ii: The value of an n-person game," Tech. Rep. RM-670, RAND Corporation, 1951.

[93] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

[94] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent individualized feature attribution for tree ensembles," *arXiv preprint arXiv:1802.03888*, 2018.

[95] E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *Journal of the American Statistical Association*, vol. 53, no. 282, pp. 457–481, 1958.

[96] S. X. Ge, D. Jung, and R. Yao, "Shinygo: a graphical gene-set enrichment tool for animals and plants," *Bioinformatics*, vol. 36, no. 8, pp. 2628–2629, 2020.

[97] M. Kanehisa, M. Furumichi, Y. Sato, M. Ishiguro-Watanabe, and M. Tanabe, "Kegg: integrating viruses and cellular organisms," *Nucleic acids research*, vol. 49, no. D1, pp. D545–D551, 2021.

[98] M. Tian, A. Yang, Q. Lu, X. Zhang, G. Liu, and G. Liu, "Study on the mechanism of baihe dihuang decoction in treating menopausal syndrome based on network pharmacology," *Medicine*, vol. 102, no. 18, p. e33189, 2023.

[99] J. Pei, M. Harakalova, H. den Ruijter, G. Pasterkamp, D. J. Duncker, M. C. Verhaar, F. W. Asselbergs, and C. Cheng, "Cardiorenal disease connection during postmenopause: The protective role of estrogen in uremic toxins induced microvascular dysfunction," *International Journal of Cardiology*, vol. 238, pp. 22–30, 2017.

[100] A. Riggio, *The Role of Runx1 in Genetic Models of Breast Cancer*. PhD thesis, University of Glasgow, Scotland, UK, 2017.

[101] H. Zhang, F. Liang, Z. Jia, S. Song, and Z. Jiang, "Pten mutation, methylation and expression in breast cancer patients," *Oncology Letters*, vol. 6, no. 1, pp. 161–168, 2013.

[102] T. R. Rebbeck, A. DeMichele, T. Tran, S. Panossian, G. Bunin, A. Troxel, and B. Strom, "Hormone-dependent effects of fgfr2 and map3k1 in breast cancer susceptibility in a population-based sample of post-menopausal african-american and european-american women," *Carcinogenesis*, vol. 30, no. 2, pp. 269–274, 2009.

[103] K. Sebova, I. Zmetakova, V. Bella, K. Kajo, I. Stankovicova, V. Kajabova, T. Krivulcik, Z. Lasabova, M. Tomka, S. Galbavy, *et al.*, "Rassf1a and cdh1 hypermethylation as potential epimarkers in breast cancer," *Cancer Biomarkers*, vol. 10, no. 1, pp. 13–26, 2012.

[104] G. Ciriello, M. L. Gatza, A. H. Beck, M. D. Wilkerson, S. K. Rhie, A. Pastore, H. Zhang, M. McLellan, C. Yau, C. Kandoth, *et al.*, "Comprehensive molecular portraits of invasive lobular breast cancer," *Cell*, vol. 163, no. 2, pp. 506–519, 2015.

[105] E. Cerami, J. Gao, U. Dogrusoz, B. E. Gross, S. O. Sumer, B. A. Aksoy, A. Jacobsen, C. J. Byrne, M. L. Heuer, E. Larsson, *et al.*, "The cbio cancer genomics portal: an open platform for exploring multidimensional cancer genomics data," *Cancer discovery*, vol. 2, no. 5, pp. 401–404, 2012.

[106] J. Gao, B. A. Aksoy, U. Dogrusoz, G. Dresdner, B. Gross, S. O. Sumer, Y. Sun, A. Jacobsen, R. Sinha, E. Larsson, *et al.*, "Integrative analysis of complex cancer genomics and clinical profiles using the cbioportal," *Science signaling*, vol. 6, no. 269, pp. pl1–pl1, 2013.

[107] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[108] Z. Mehrbakhsh, R. Hassanzadeh, N. Behnampour, L. Tapak, Z. Zarrin, S. Khazaei, and I. Dinu, "Machine learning-based evaluation of prognostic factors for mortality and relapse in patients with acute lymphoblastic leukemia: a comparative simulation study," *BMC Medical Informatics and Decision Making*, vol. 24, no. 1, p. 261, 2024.

[109] J. Zhou, X. Lu, W. Chang, C. Wan, X. Lu, C. Zhang, and S. Cao, "Plus: Predicting cancer metastasis potential based on positive and unlabeled learning," *PLoS computational biology*, vol. 18, no. 3, p. e1009956, 2022.

[110] I. Abed Al-Ghafer, N. AlAfeshat, L. Alshomali, A. Shaheen, H. Qattous, M. Azzeh, and A. Alkhateeb, "Nmf-guided feature selection and genetic algorithm-driven framework for tumor mutational burden classification in bladder cancer using multi-omics

data," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 13, no. 1, p. 26, 2024.

[111] A. N. Wu, R. Stouffs, and F. Biljecki, "Generative adversarial networks in the built environment: A comprehensive review of the application of gans across data types and scales," *Building and Environment*, vol. 223, p. 109477, 2022.

[112] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. CreateSpace, 2009.

[113] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[114] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, *et al.*, "Array programming with numpy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

[115] W. McKinney *et al.*, "Data structures for statistical computing in python.," *scipy*, vol. 445, no. 1, pp. 51–56, 2010.

[116] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[117] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *Journal of Machine Learning Research*, vol. 18, no. 17, pp. 1–5, 2017.

[118] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[119] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[120] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. E. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. B. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing, "Jupyter notebooks – a publishing format for reproducible computational

workflows," in *Positioning and Power in Academic Publishing: Players, Agents and Agendas* (F. Loizides and B. Schmidt, eds.), pp. 87–90, IOS Press, 2016.

[121] Google, "Google colaboratory," 2026. Accessed: 2026-04-11.