# CytoAutoCluster Project Documentation

## Project Process Overview

### Dimensionality Reduction Techniques: PCA and t-SNE

Purpose: PCA and t-SNE were employed for reducing data dimensions, allowing better visualization of clusters while preserving key patterns. PCA captures linear variance, while t-SNE provides a nonlinear view, clustering similar data points.  
Challenges: Selecting optimal components for PCA and tuning t-SNE’s perplexity and iterations were complex, as misconfiguration could distort cluster formations.  
Outcomes: PCA highlighted primary variance directions, and t-SNE visualized distinct clusters, setting the foundation for further dimensionality reduction with Autoencoders.

### Autoencoders for Feature Extraction

Purpose: Autoencoders compressed the high-dimensional data into latent features, improving representation for clustering. The aim was to balance efficient encoding with minimal information loss.  
Challenges: Hyperparameter tuning to prevent overfitting and deciding the bottleneck layer size were key challenges.  
Outcomes: Autoencoders successfully captured core features, enhancing data representation and paving the way for semi-supervised learning on imbalanced data.

### Semi-Supervised Learning (SSL)

Purpose: SSL was applied to leverage both labeled and unlabeled data, utilizing techniques like entropy minimization and consistency regularization to maximize the model’s learning potential.  
Challenges: Implementing regularization methods to generalize the model effectively was challenging, requiring extensive experimentation with SSL parameters.  
Outcomes: SSL enhanced model performance by effectively learning from unlabeled data, supporting data corruption techniques in the next phase.

### Data Corruption Techniques

Purpose: Binary masking and data shuffling introduced controlled noise, encouraging model robustness by simulating data variability.  
Challenges: Balancing data masking levels to obscure features without distorting patterns was essential.  
Outcomes: Data corruption techniques improved the model’s adaptability, enabling it to identify key patterns in noisy or partial data. This prepared the data for initial model testing.

### Model Development and Evaluation

Logistic Regression and XGBoost: These models provided a baseline (Logistic Regression) and high performance (XGBoost) with encoded data. XGBoost’s ability to handle complex structures complemented Logistic Regression’s simplicity.  
Challenges: Overfitting prevention through feature scaling and loss value adjustments were necessary to optimize performance.  
Outcomes: Both models leveraged encoded features effectively. XGBoost, in particular, showed strong accuracy on complex data, leading to Encoder model exploration.

### Encoder Model Development

Purpose: The Encoder model refined latent features, generating high-quality representations for downstream training.  
Challenges: Dimension balancing and training speed optimization required careful adjustments.  
Outcomes: Encoder-generated features improved model accuracy when used in supervised classifiers, building the foundation for further training on latent data.

## Final Timeline

### October 17, 2024

Agenda: Complete Data Exploration, implement PCA and t-SNE

Action Items: Gain deeper understanding of PCA and t-SNE

### October 18, 2024

Agenda: Review PCA and t-SNE outputs

Action Items: Finalize t-SNE and compare with other results

### October 22, 2024

Agenda: Begin work on Autoencoders

Action Items: Basic understanding of Autoencoders

### October 23, 2024

Agenda: Identify Autoencoder models for tabular data

Action Items: Understand Autoencoders for feature extraction

### October 25, 2024

Agenda: Learn Semi-Supervised Learning basics

Action Items: Study SSL concepts

### October 28, 2024

Agenda: Discuss entropy minimization and consistency regularization

Action Items: Implement SSL techniques

### October 29, 2024

Agenda: Review binary mask implementation

Action Items: Evaluate binary mask outputs

### October 30, 2024

Agenda: Data corruption methods

Action Items: Evaluate corrupted data for training

### November 1, 2024

Agenda: Data splitting and train-test setup

Action Items: Complete model preparation

### November 5, 2024

Agenda: Implement Logistic Regression

Action Items: Finalize model parameters

### November 7, 2024

Agenda: Develop Encoder model

Action Items: Initiate latent feature training

### November 8, 2024

Agenda: Train on Logistic Regression and XGBoost

Action Items: Troubleshoot Encoder-based training

### November 12, 2024

Agenda: Begin Semi-Supervised Learning model

Action Items: Implement SSL on encoded data