**Predicting Gene Essentiality Using Multi-Omics Features and Deep Learning**

**Abstract (≤100 words)**

This project aimed to predict gene essentiality in human cancer cell lines using multi-omics data from the DepMap 24Q2 release. A variational autoencoder (VAE) learned latent representations from gene expression profiles, and a multilayer perceptron (MLP) model predicted CRISPR gene effect (CERES) scores using expression, copy number variation, and mutation features. Despite modest quantitative performance (R² = 0.02, MAE = 0.24, ρ = 0.14), qualitative trends aligned with biological expectations—core essential genes (e.g., *RPA1*, *PCNA*) showed strong dependency, while oncogenes displayed cell-specific effects. The workflow demonstrates an interpretable, accessible AI pipeline for multi-omics integration and gene essentiality prediction.

**Background and Motivation**

Understanding which genes are essential for cancer cell survival guides target discovery and synthetic-lethal screening.  
The Broad Institute’s **Cancer Dependency Map (DepMap)** provides genome-wide CRISPR knockout screens across >1,000 cell lines, reporting **CERES** scores that quantify loss-of-function impact.

While many approaches model CERES directly from single omics layers (e.g., expression only), this project explores whether **multi-omics features** (gene expression + copy number + mutation) combined with **deep latent embeddings** from a VAE can improve predictive modeling of essentiality.

This task is relevant because:

* It integrates several core topics from the DDLS course: dimensionality reduction, neural network modeling, FAIR data, and accessibility.
* It uses **public datasets** with clear licensing and high biomedical value.
* It enables translation into an **interactive AI-driven tool** for researchers to explore gene dependencies.

**Dataset Summary**

**Data source**

All data were downloaded from the **DepMap Public 24Q2 release**:

* OmicsExpressionProteinCodingGenesTPMLogp1.csv
* CRISPRGeneEffect.csv (CERES scores)
* OmicsCNGene.csv (copy number)
* OmicsSomaticMutationsProfile.csv
* Mapping: OmicsProfiles.csv (ProfileID → ModelID)

**Preprocessing**

* **Mapping & alignment:** ProfileID entries from mutation data were mapped to ACH-... DepMap IDs using OmicsProfiles.csv.
* **Matrices unified** to common cell × gene space (~1,066 cell lines × 17,622 genes).
* **NaN imputation:** per-gene median for expression/CNV; binary fill (0/1) for mutation.
* **Standardization:** z-score scaling per gene for expression & CNV.
* **Targets:** CERES values (negative = essential).
* **Train/test split:** 80 / 20 by cell line (ensures unseen cell generalization).

**Exploratory analysis**

Latent embeddings of cells learned by the VAE were visualized using PCA and UMAP.

|  |
| --- |
| *A blue dots on a white background  AI-generated content may be incorrect.*  *Figure1.* PCA(2) of latent space: Shows structure among cell lines. |  |
| *A blue dots on a white background  AI-generated content may be incorrect.*  *Figure 2.* UMAP of latent space: Preserves local neighborhood; reveals clusters   |  | | --- | |  | |  |

**Method Description**

**Workflow overview**

1. **VAE (Stage A)** — Learns 32-dimensional latent embeddings from log-TPM gene expression.
   * Encoder: [1024→256] layers, mean & log-variance heads.
   * Decoder: reconstructs expression.
   * Loss: reconstruction + KLD (regularization).
2. **MLP (Stage B)** — Predicts CERES per gene from [Z\_cell, expr\_g, cnv\_g, mut\_g].
   * Architecture: [256→64→1], GELU activations, dropout 0.2.
   * Loss: Smooth L1.
   * Optimization: Adam (lr = 1e-3), 25 epochs, block-wise gene batching (400 genes).
3. **Evaluation metrics**
   * Regression: R², MAE, Spearman ρ
   * Classification (CERES < −0.5 = essential): ROC-AUC, PR-AUC
4. **Accessibility:** Wrapped as a Gradio web app allowing user-supplied genes and adjustable threshold.

**Architecture sketch**

Expression ─┐

│ VAE → Latent (32D)

CNV ────────┤

Mutation ───┘

↓ concat

[Z\_cell | expr\_g | cnv\_g | mut\_g]

↓

MLP

↓

Predicted CERES

**Results**

**Quantitative metrics (test cells)**

| **Metric** | **Value** |
| --- | --- |
| R² | **0.020** |
| MAE | **0.244** |
| Spearman ρ | **0.136** |
| ROC-AUC | **0.630** |
| PR-AUC | **0.152** |

→ modest regression fit but non-random correlation with true CERES, confirming biologically meaningful signal capture.

**Figures**

**A graph of a blue object

AI-generated content may be incorrect.**

*Figure 3.* Observed vs Predicted CERES: Slightly negative correlation; spread reflects cross-gene variability

A graph of a number of cells

AI-generated content may be incorrect.

*Figure 4.* Residual distribution: Centered towards 0 with mild skew toward underprediction

**Qualitative biological checks**

* **Core essentials** (e.g., *PCNA*, *RPA1*, *POLR2A*) consistently showed CERES < −1 across lines.
* **Oncogenes** (*KRAS*, *MYC*, *BRAF*) displayed context-dependent essentiality.
* Predictions captured approximate rank order even when magnitudes differed.

**Accessibility wrapper**

The **Gradio app** (https://ef6d0df6cefa38a8b9.gradio.live/) enables:

* Input: ACH cell line + genes (text or CSV)
* Output: predicted CERES, observed CERES (if available), and *Essential/Non-essential* label (default threshold = −0.5).
* Optional slider adjusts threshold; results downloadable as CSV.

**Conclusion & Discussion**

This project demonstrates an end-to-end **AI-assisted multi-omics modeling workflow** for predicting gene essentiality.  
While absolute predictions remain noisy, the model captured **relative dependency structure** between genes and cell lines.

**Key takeaways**

* VAE embeddings successfully compressed expression profiles while preserving cell-type relationships.
* MLP captured partial essentiality signals (ρ ≈ 0.14).
* Performance could improve with gene-wise normalization, deeper architectures, or transfer learning (e.g., scVI/scArches).
* The pipeline is reusable, modular, and transparent—key aspects of FAIR data science.

**Limitations**

* Limited training time (single run, 25 epochs).
* Global model across all genes; per-gene modeling might yield higher accuracy.
* CERES targets inherently noisy; metrics underestimate biological interpretability.

**Future directions**

* Implement variational graph autoencoders for gene-network regularization.
* Fine-tune with cross-cell and cross-gene attention (transformer encoder).
* Integrate functional annotation embeddings to enhance interpretability.

**Data and Code Availability**

* **Datasets:** Broad Institute DepMap 24Q2 Public release (https://depmap.org/portal/data\_page/).
* **Preprocessing scripts, trained models, and web app:**  
  [DDLS Essentiality Repo – Google Drive link: https://drive.google.com/drive/folders/1yrMFdGZH6bbrlNON8Gdjq\_I00WVjv1Tj?usp=sharing  
  the link is publicly accessible]
* **FAIR principles:** all code and outputs versioned, modular, and documented.

**Acknowledgments**

This work was conducted as part of the **DDLS 2025 course**.  
Thanks to course instructors for guidance on project validation, and to the DepMap team for open data access.  
GenAI tool ChatGPT was used for code brainstorming, pipeline debugging, and documentation drafting, as required by the assignment guidelines.  
All analytical choices and interpretations are by the student.

**References**

1. Meyers RM *et al.* (2017). *Computational correction of copy number effect improves specificity of CRISPR–Cas9 essentiality screens in cancer cells.* **Nat. Genet.** 49, 1779–1784.
2. Tsherniak A *et al.* (2017). *Defining a cancer dependency map.* **Cell** 170, 564–576.
3. DepMap Consortium (2024). *DepMap Public 24Q2 Dataset.* Broad Institute. https://depmap.org/portal/data\_page/
4. Kingma DP & Welling M (2013). *Auto-Encoding Variational Bayes.* arXiv:1312.6114.
5. He K *et al.* (2015). *Deep Residual Learning for Image Recognition.* CVPR.

**Appendices**

* **A. AI Deep Research Log:** transcripts of brainstorming with ChatGPT attached: <https://chatgpt.com/share/6902640d-374c-8011-b034-f2dcd1fcaa21>
* **B. Sample Predictions:** predictions\_ACH-001289.csv<file:///Users/sulagnadasgupta/Downloads/predictions_ACH-001289.csv>, predictions\_ACH-002048.csv<file:///Users/sulagnadasgupta/Downloads/predictions_ACH-002048.csv>

**Agent Demo**

A 3–5 minute screen recording demonstrates:

* the overall working of the app
* it shows the user uploading a list of 25 genes and performing predictions for different cell lines
* it also shows predictions with varying CERES thresholds across various cell lines