Translation efficiency (TE) describes how effectively an mRNA transcript is converted into protein. Ribosomal protein genes are essential for cell function and exhibit coordinated TE patterns, making them a meaningful target for a machine learning prediction task. Predicting TE from genome-wide expression data presents several challenges, including high dimensionality, correlated features, and nonlinear biological relationships.

This project will build an end-to-end machine learning system that predicts TE for ribosomal protein genes using paired RNA expression and TE measurements. The data comes from Dr. Can Cenik's lab, using high-quality ("QC-pass") HEK293T cell samples. The goal is to compare classical models with advanced nonlinear and representation-learning methods to determine which techniques best capture structure in TE data.

This project fits the assignment's example project categories: it applies advanced classical algorithms (such as XGBoost and Support Vector Regression) to a tabular dataset and also incorporates neural network modeling and dimensionality reduction, aligning with the example encouraging exploration of neural architectures and modern deep learning approaches.

2. Data Sources

The dataset comes from Dr. Cenik's lab and consists of two matched tables:

RNA Expression Table (RNA_HEK293T.csv)

- 8,433 genes (rows)
  96 HEK293T samples (columns; sample IDs like GSM3323389)

Translation Efficiency Table (TE_HEK293T.csv)

- 8,433 genes × 96 samples
- CLR-normalized TE values
- TE reflects ribosome occupancy normalized by expression

Because the tables share identical dimensions and gene ordering, they can be cleanly aligned for supervised learning, using RNA expression features to predict TE values.

Planned preprocessing

All work will be performed inside the project:

- Importing both CSVs and aligning gene names
- Checking for missing or low-quality values
- Filtering if necessary

- Normalizing or scaling features for certain models
- Constructing feature and target matrices
- Creating training, validation, and test splits

3. High-Level Methods, Techniques, and Technologies

A range of machine learning approaches will be used to evaluate how well TE can be predicted.

Classical Regression Models

- Linear Regression, Ridge and Lasso

Advanced Classical / Nonlinear Models

- Random Forest, Gradient Boosting (XGBoost)

Dimensionality Reduction and Representation Learning

- Principal Component Analysis

Neural Network Approaches

- Fully connected networks
- Multi-task regression for predicting multiple TE values jointly

Tools & Evaluation Metrics

- Python, NumPy, pandas, scikit-learn, XGBoost, PyTorch/Keras
- Metrics: root mean squared error, R^2, predicted vs. actual TE scatterplots, feature importance charts, latent-space visualizations

4. Products to Be Delivered

A. Code & Workflow

Jupyter notebooks for:

- Data preprocessing, Exploratory analysis, Model training and tuning, Dimensionality reduction, Performance comparison across models

B. Visual Outputs

- Predicted vs. actual TE plots, PCA visualizations