# Assessing and Enhancing Model Performance for Gene Expression Prediction using DNA Methylation and Copy Number Variants

CS 309 Final Project

Zhujun Yao, Qixuan Wang

May 5, 2024

**Abstract**

This study explores efficient computational models for predicting gene expression levels from DNA methylation and copy number variation (CNV) data. Amidst the increased accumulation of multi-omics data in biomedical research, the objective is to construct and validate predictive models surpassing the performance of current Denoising Autoencoders (DAEs) applied in gene expression prediction. Our approach involved rigorously evaluating deep learning autoencoder techniques against conventional machine learning methods through cross-validation, error analysis, and assessment of computational efficiency. Inspired by the architecture of Variational Autoencoders (VAEs), we have originally proposed a denoising probabilistic autoencoder model (DPAE), and it outperforms existing DAE model. We also reveal that ridge regression to be superior in terms of lower mean squared error (MSE), higher R-squared values, and reduced computational time. Our results challenge the contemporary trend of applying DL to multi-omics data interpretation by highlighting the effectiveness and efficiency of ridge regression in predicting gene expression. These insights are pivotal to advancing cancer diagnostic modalities, offering a cost-effective and faster alternative to conventional RNA-seq diagnostics.

**Keywords**: Gene Expression, DNA Methylation, Copy Number Variation, Ridge Regression, Deep Learning, Autoencoder

# Contents

# 1 Introduction

Multiple factors contribute the gene expression changes. One of the important factors is DNA methylation, which is the addition of a methyl group to a DNA base. DNA methylation plays a role in gene regulation, such as X-chromosome inactivation and transposable elements repression (Razin & Cedar, 1991). It is generally believed that DNA methylation blocks gene expression by preventing the interaction between DNA and chromatin proteins or specific transcription factors. Nevertheless, researchers have identified multiple exceptions where hypermethylation of DNA does not always lead to downregulated gene expression, and a more quantitative description of their correlation is demanded.

Copy number variant (CNV) is another factor that modulates gene expression changes. CNV is defined by the insertion and deletion of chromosomal fragments of more than 1 kilobases. These variations have a significant impact on gene expression levels, particularly in the development and progression of cancers. Studies such as the Broad-Novartis Cancer Cell Line Encyclopedia, NCI-60, and the Cancer Genome Atlas reveal a strong correlation between CNVs and the differential gene expression of oncogenes and tumor suppressor genes, suggesting that CNVs may serve as a crucial intermediary mechanism affecting gene expression and thus influencing cancer phenotypes (Brewer et al., 2023; Mohanty et al., 2021; Shao et al., 2019). Understanding the linkage between CNVs and gene expression is critical for advancing cancer prevention, diagnosis, and treatment strategies.

As multi-omics data accumulates, predicting gene expression via other multi-omics data with high confidence becomes possible (Hira et al., 2021; Odenkirk et al., 2021). We hope to build a good model that predicts gene expression level based on DNA methylation, CNV and gene expression. On one hand, deciphering the relationship among multi-omics data is within the natural curiosity of biomedical scientists, and such model can be useful for cancer transcriptome profiling data simulations when faced with limited access or scarcity of available datasets, similar to the purpose of methylome simulation, methCancer-gen, by Choi & Chae (2020). On the other hand, a good model in gene expression prediction can potentially eliminate RNA-seq clinical diagnostics, which cost around $500 per sample for a 150bp paired-end sequencing (Byron et al., 2016). Therefore, our models have a practical purpose in providing cost-effective and efficient clinic diagnostic. After literature review, we have decided to focus on deep learning Denoising Autoencoder and Variational Autoencoder models, as explained in the related work section.

# 2 Related work

To quantitatively describe DNA methylation's effect on gene transcription, Zhong et al. (2019) have generated predictive models of gene transcriptional level based on DNA methylation status. Yet, they failed to find the correlation between gene expression levels by using solely DNA methylation on CpG island. Among single linear regression, multinomial linear regression, and LASSO regression, the best model, LASSO regression model, only 30 and 42 genes were found to have cross-validation R2 greater than 0.3 (Zhong et al., 2019). To give a better prediction, researchers later have incorporated more features other than CpG island methylation level to make more accurate predictions, such as methylation sites' distance to transcription start sites (TSS), miRNA expression, and

copy number variations (CNV) (Hira et al., 2021; Kim et al., 2020).

Among these papers incorporating multi-omics data, the deep learning model autoencoder has gained prevalence, as it is proven to be an ideal model in gene expression level prediction and cancer subtypes from high dimensional multi-omics data (Franco et al., 2021a; Seal et al., 2020; Tsimenidis et al., 2022; Yassi et al., 2023). There are mainly 4 types of autoencoders: the Sparse Autoencoder (Franco et al., 2021a), Contractive Autoencoder (CAE) (Diallo et al., 2021) Denoising Autoencoder (DAE) (Vincent et al., 2008), and Variational Autoencoder (VAE) (Titus et al., 2018). Among them, DAE and VAE are the most popular two algorithms (Franco et al., 2021b; Hira et al., 2021; Yassi et al., 2023).

Variational Autoencoders (VAEs) is designed primarily as a method for the approximation of inference in the context of latent variable modeling. The motivation behind VAEs is to address scenarios where each piece of data is correlated with an underlying latent vector, making them particularly suitable for dealing with gene expression level prediction because it is regulated by complicated transcriptional factor regulation upon DNA methylation and CNV alternations. VAEs have been applied to a variety of genomic problems such as predicting gene expression patterns or DNA methylation levels among cancer-related genes. A conditional variational autoencoder model was established to generate cancer type-specific DNA methylome datasets, allowing researchers to simulate epigenetic data tailored to different cancers (Choi & Chae, 2020). In another paper, a variational autoencoder is used to compress high-dimensional single-cell genomic data into a feature space that effectively distinguishes hidden tumor subpopulations while also revealing cell lineages and differentiation trajectories (Rashid et al., 2021). An MMD-VAE model for cancer subtype classification is shown to range from 93.2 to 95.5% (Hira et al., 2021). Therefore, VAE is one the most suitable models to link phenotypical data and gene expression with upstream omics data like DNA methylation.

Denoising autoencoders are also popular due to their application in high predictive power in gene expression level prediction. Seal et al. (2020) have proposed a DAE-MLP model to predict gene expression from DNA methylation and CNV. The technique involves a multi-step process. To begin, a preprocessing step is implemented, incorporating the computation of the mean methylation levels in proximity to the TSS, pinpointing shared samples and genes across all three omics datasets, filling in any gaps in the data, and standardizing the datasets using tools from the Scikit-learn library. After this, a deep learning (DL) regression framework is developed, employing a deep denoising autoencoder (DDAE) for extracting features and minimizing data complexity, which is integrated with a multilayer perceptron (MLP) for multi-target regression that forecasts gene expression (GE) levels. The culmination of the process comes with the DL model's capacity to distinguish features that separate tumor samples from normal ones, thereby confirming the effectiveness of the model's design. The model has demonstrated exceptional outcomes in the regression task, evidenced by performance metrics including a negative log-scaled root mean square error (RMSE) of 1.33, an R-squared value of 0.96, and a Pearson correlation coefficient of 0.68.

There are several drawbacks to Seal's work. First, the authors did not examine the data distribution and correlations among the variables, which should otherwise be reported (Hira et al., 2021; Taguchi et al., 2023). Second, the paper does not use cross-validation. Third, the paper does not report the running time for the DDAE-MLP model against conventional machine learning methods. Fourth, only one type of cancer dataset was included, which is not sufficient to prove the validity of such a model among similar

types of DNA methylation and CNV data.

In this project, we hope to tackle the issues in Seal et al. (2020) by providing more stringent model evaluations. We also increased the R-squared value of autoencoder DL models to predict gene expression levels based on DNA methylation CNV data, mainly DAE and VAE. Interestingly, we discovered that conventional machine learning methods have their advantages in running time as well as high R-squared value. We believe the current trend of using DL for gene expression prediction is not necessary, as Ridge regression can reach an R-squared stably above 98%.

# 3 Dataset and features

The datasets include two types of cancer: LIHC and OV. The LIHC datasets were used in Seal et al. (2020), which was uploaded at https://zenodo.org/records/3712496. The OV data were used in Hira et al. (2021), and was directly downloaded from UCSC xenabrower: https://xenabrowser.net/. The LIHC data's DNA methylation set was processed by the authors by annotating the methylated CpG islands within 1500bp. The OV data's DNA methylation was processed by us using GenomicRanges and IRanges in R. UCSC ID was converted to gene symbols using UCSC API. The methodology for data harvesting and data structure are listed below in Table 1 and Table 2.

***Table 2.*** *The Multi-Omics Data Information of OV datasets at UCSC*

| ***Omics type*** | **#Samples** | **#Genes** | **Assay** | **Platform** |
|---|---|---|---|---|
| *DNA methylation* | 555 | 14995 | Infinium HumanMethylation27 BeadChip | methylation_27 |
| *CNV* | 555 | 23700 | Affymetrix SNP Array 6.0 | cna_cnv.hg19 |
| *RNA-Seq* | 555 | 9031 | Illumina HiSeq | gene.normalized_RNA-seq |

After downloading RNA, DNA methylation, and CNV data, common samples were extracted, and outliers of RNA-seq data was filtered based on the previous reported parameters (Seal et al., 2020). Rows with more than 20% of NAs were filtered. All numerical values were normalized among biological samples.

| | TCGA.61.1910.01 | TCGA.61.1720.01 | TCGA.31.1953.01 | TCGA.13.1819.01 | TCGA.13.0764.01 | TCGA.31.1951.01 | TCGA.24.1552.01 | TCGA.13.1477.01 | TCGA.30.1862.01 | TCGA.29.1698.01 | ... | TCGA.23.2079.01 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5S_RRNA | 0.010800 | 0.01250 | 0.010600 | 0.011900 | 0.02030 | 0.015750 | 0.00875 | 0.018050 | 0.01040 | 0.00905 | ... | 0.00925 |
| A1BG | 0.964900 | 0.95930 | 0.971800 | 0.976400 | 0.98050 | 0.975400 | 0.98840 | 0.971600 | 0.97470 | 0.98900 | ... | 0.98490 |
| A1CF | 0.280800 | 0.30470 | 0.354500 | 0.404300 | 0.33990 | 0.732500 | 0.31390 | 0.454700 | 0.51810 | 0.46790 | ... | 0.90360 |
| A2M | 0.569500 | 0.74410 | 0.479600 | 0.786300 | 0.24940 | 0.668500 | 0.69810 | 0.689300 | 0.80230 | 0.85500 | ... | 0.86610 |
| A2ML1 | 0.569550 | 0.69225 | 0.633650 | 0.355800 | 0.83355 | 0.688250 | 0.20240 | 0.858950 | 0.16815 | 0.56705 | ... | 0.84505 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| ZWINT | 0.011500 | 0.01695 | 0.013600 | 0.019500 | 0.01810 | 0.017450 | 0.01160 | 0.024150 | 0.01955 | 0.01130 | ... | 0.01190 |
| ZZEF1 | 0.014125 | 0.01895 | 0.017925 | 0.025225 | 0.01515 | 0.023725 | 0.00980 | 0.028475 | 0.02265 | 0.01560 | ... | 0.01410 |
| ZZZ3 | 0.015050 | 0.02170 | 0.016150 | 0.019100 | 0.01780 | 0.021150 | 0.01190 | 0.054850 | 0.01650 | 0.01250 | ... | 0.01380 |

15493 rows × 555 columns

Figure 1: Sample data structure for single-omics dataset. This figure is an example for DNA methylation data, where the methylation at gene level is calculated by taking the average of all CpG island methylation level within 1500bp distance to the transcription starting sites (TSS). The gene expression level and copy number variants data are all preprocessed to this format.

The data distribution of dependent and independent variables is examined (Figure 2). For the gene expression data, it follows the bimodal gene expression reported in Figure 5 by Ertel & Tozeren (2008). The DNA methylation data demonstrate that many genes do

not have CG methylation within 1500bp of TSS, and the result is consistent with previous findings (Singhal et al., 2015). The normalized CNV data looks relatively normally distributed, which also matches previous findings (Pokrovac et al., 2023). The heatmaps allow us to observe the heterogeneity within the datasets. In CNV and DNA methylation heatmaps, evident subclusters existed, indicating that these patients or healthy individuals can be separated via these variables. Interestingly, the gene expression heatmap does not have any pattern, which we hypothesized can be deciphered via CNV and DNA methylation data.

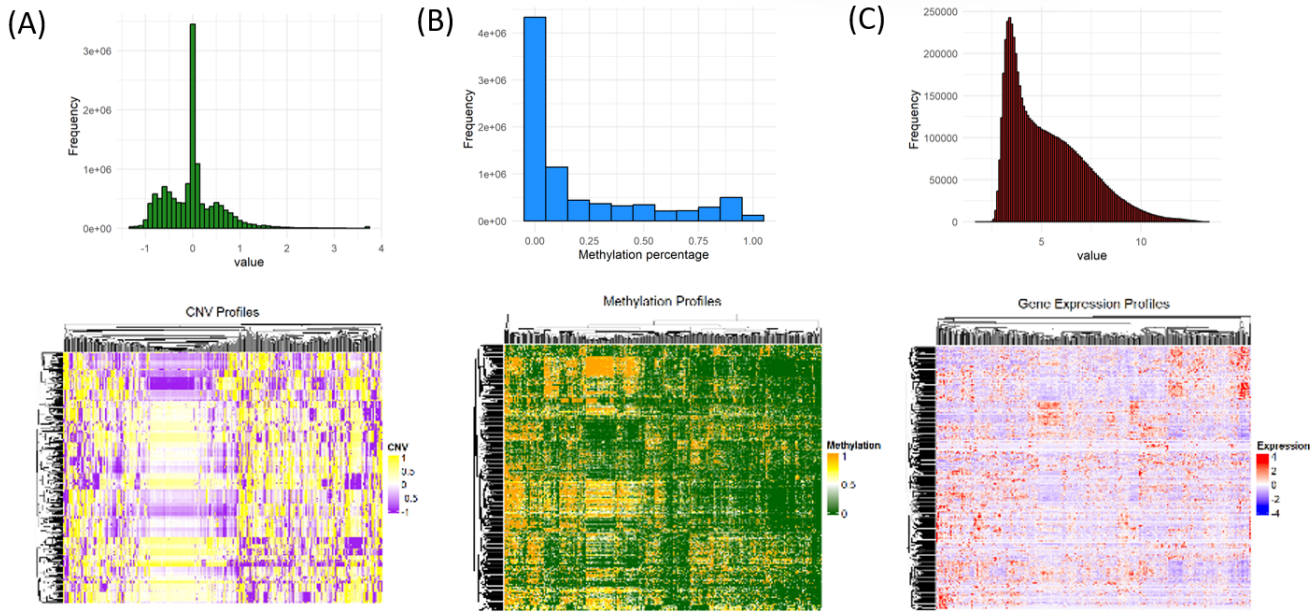


Figure 2: The data distribution of three variables.The figure demonstrates histograms and heatmaps of CNV, DNA methylation, and gene expression respectively. The heatmaps are plotted with hierarchical clustering by rows and columns.

The correlations between the three variables are analyzed in Figure 3. Figure 3A, 3B, and 3C are the data presented in our work. It is showned that gene expression decrease in general as DNA methylation increases or CNV number decreases. Unsurprisingly, CNV and DNA are negatively correlated. Such data correlations are canonical and can be explained in biology (Hira et al., 2021). Figure 3D, 3E, and 3F are referred from Hira et al. (2021). (D) is the relationship between DNA methylation (cp25247520) and its nearby gene PVT1. Me1, Me2, and Me3 indicate the number of methyl groups, demonstrating that more methyl groups lead to less gene expression. (E) is the relationship between CNV of PVT1 and PVT1 gene expression. CN1, CN2, CN3 indicate the copy number of PVT1, the increase of copy number will lead to the increase in gene expression. (F) is the relationship between PVT1 CNV and DNA methylation at nearby CpG island ( (cp25247520)). In general, the correlation trends among the variables are consistent with the canonical biological patterns reported in Hira et al.(2021).

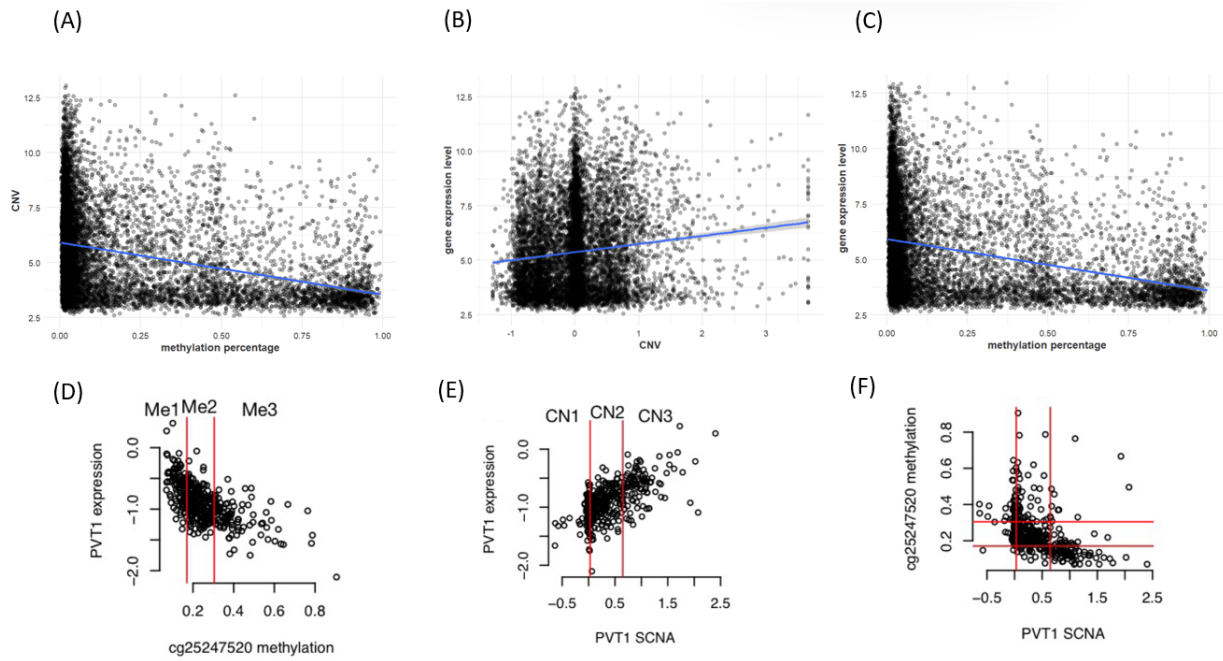


Figure 3: The correlation between variables. (A) DNA methylation level is negatively correlated with gene expression, and (B) CNV is positive correlated with gene expression. (C) Unsurprisingly, CNV is negatively correlated with DNA methylation. Figure (D),(E),(F) are canonical relationships between multi-omics data in Hira et al.(2021).

# 4 Methods and models

In order to perform the task of estimating gene expression, Seal and his co-authors (2020) have developed a deep learning regression model based on multi-omics integration. They have used deep denoising autoencoder (DDAE) for dimension reduction and feature extraction and multi-layer perceptron (MLP) for regression. In machine learning, dimensionality reduction is the process of reducing the number of features that describe some data. This reduction can be achieved either by selection, where only certain existing features are retained, or by extraction, where a smaller set of new features is derived from the original ones. This process is useful in various scenarios that require lower-dimensional data, such as data visualization, data storage, and situations involving intensive computations.

To evaluate and compare the performance of different models, Seal et al. have used the MSE and $R^2$ statistic (also known as coefficient of determination). Recall that

$$MSE = \frac{1}{n}\sum_{i=1}^{\infty}(y_i - \hat{y}_i)^2,\ RSS = \sum_{i=1}^{\infty}(y_i - \hat{y}_i)^2$$

$$TSS = \sum_{i=1}^{\infty}(y_i - \bar{y})^2,\ R^2 = 1 - \frac{RSS}{TSS}.$$

In this section, we are going to improve the original models developed by Seal et al. and compare the MSE and $R^2$ statistic.

## 4.1 Denoising autoencoder

A denoising autoencoder (DAE) is a type of artificial neural network used to learn efficient codings by training to ignore "noise" in the input data. It's a variation of the more general autoencoder, which is designed to encode input data into a smaller, dense representation and then reconstruct the input data from this representation as accurately as possible. Specifically, a DAE is a neural network model that removes noise from noisy data by learning to reconstruct the original data from the noisy version. It is trained to minimize the difference between the original and reconstructed data. A DAE consists of an encoder, which maps the input data to a lower-dimensional representation, or encoding, and a decoder, which maps the encoding back to the original data space. DAEs have a lot of applications in speech processing, computer vision, and natural language processing. For example, DAEs can be applied to image denoising, fraud detection, and data compression.

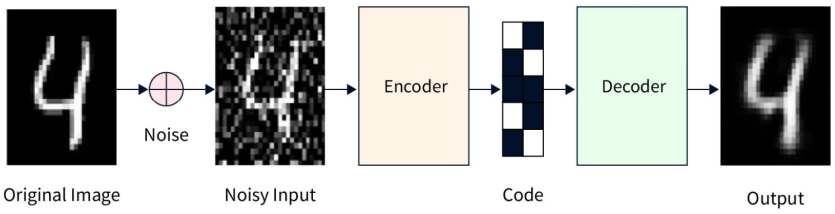


Figure 4: The architecture of a denoising autoencoder

Figure 4 shows the architecture of a denoising autoencoder. We first introduce random noise to the input. The noise can be anything from simple Gaussian noise added to the input data, to more complex forms. Encoder is a neural network with one or more hidden layers. It accepts noisy data as input and produces a lower-dimensional encoding, effectively compressing the data. On the other hand, the decoder operates as an expansion function, reconstructing the original data from this compressed encoding. The input for the decoder is the encoding generated by the encoder, and its output is a reconstruction of the original data. Like the encoder, the decoder is also a neural network with one or more hidden layers. By using DAE for dimension reduction and multi-layer perceptron for regression, Seal et al. have computed the MSE and $R^2$ statistic of their deep learning models. The table below shows the The MSE and $R^2$ statistic of the original models developed by Seal et al.

| MSE | $R^2$ statistic |
|---|---|
| 0.01565 | 0.96582 |

Table 1: The MSE and $R^2$ statistic of the original models

We have improved the performance of the original DAE by introducing the following changes:

- Use a Different Optimizer: Replace the Stochastic Gradient Descent (SGD) optimizer with Adam (Adaptive Moment Estimation), which may perform better due to its adaptive learning rate capabilities
- Add Batch Normalization layers, which can help in stabilizing the learning process by normalizing the inputs of each layer. Batch normalization works by normalizing

the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation.

- Dynamic Learning Rate: Implement learning rate scheduling to decrease the learning rate gradually during training
- Use callbacks like EarlyStopping to stop training when the validation loss stops improving.

The MSE and $R^2$ statistic of our improved DAE and the same MLP is shown in the table below. The $R^2$ value has increased by 0.01405 compared with the original DAE.

| MSE | $R^2$ statistic |
|---|---|
| 0.01679 | 0.97987 |

Table 2: The MSE and $R^2$ statistic of the improved DAE and the same MLP

## 4.2 Variational autoencoder

A variational autoencoder (VAE) is a type of autoencoder that uses techniques from Bayesian inference to generate complex models that can be sampled and interpreted. It's a powerful tool in the field of machine learning, particularly for the tasks of unsupervised learning, dimensionality reduction, and generative modeling. A VAE offers a probabilistic way to represent an observation in latent space. Instead of an encoder that outputs a single value for each latent state attribute, a VAE designs its encoder to define a probability distribution for each latent attribute. This approach has many applications, including data compression, the generation of synthetic data, and more. A variational autoencoder differs from a traditional autoencoder in that it describes dataset samples in latent space through a statistical approach. In a variational autoencoder, the encoder produces a probability distribution at the bottleneck layer rather than a single output value.

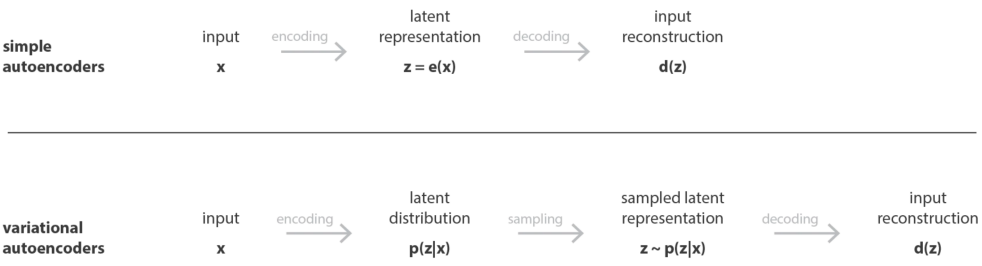


Figure 5: The architectures of a standard AE and VAE

Figure 5 shows the differences of the architectures of a standard autoencoder and variational autoencoder. The encoder-decoder architecture of variational autoencoders distinguishes them from traditional autoencoders. The encoder part of a VAE takes the input data and transforms it into a distribution in a latent (hidden) space. This distribution is typically parameterized by means ($\boldsymbol{\mu}$) and variances ($\boldsymbol{\sigma}$), which define a Gaussian probability distribution. The latent code generated by the encoder in a VAE is a probabilistic encoding, allowing the VAE to represent not just a single point in latent

space, but a distribution of possible representations. The decoder takes a sample from the latent space distribution (generated by the encoder) and attempts to reconstruct the original input data. The process of sampling and reconstructing adds a generative aspect to the VAE, allowing it to generate new data points similar to the training data.

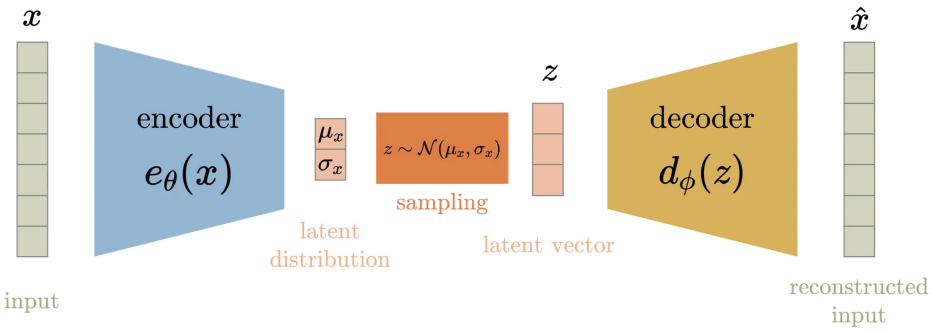


Figure 6: The architectures of a VAE

Figure 6 shows the architectures of a VAE vividly. The loss function in VAEs is composed of two parts. Reconstruction loss encourages the decoded samples to match the original inputs, thus ensuring that the VAE learns to accurately reconstruct the data. KL divergence is a regularizer that measures how much information is lost when using the approximated distribution (the one produced by the encoder) to represent the true underlying distribution. It helps in keeping the latent space distributions well-formed and continuous, which is essential for generating new data points. Unlike standard autoencoders, VAEs are generative models. They can generate new data points that are similar to the training data. This makes them useful for tasks like image generation, style transfer, and more.

## 4.3 Denoising probabilistic autoencoder

Inspired by the architecture of a variational autoencoder, we propose a probabilistic version of the denoising autoencoder and we call it denoising probabilistic autoencoder (DPAE).

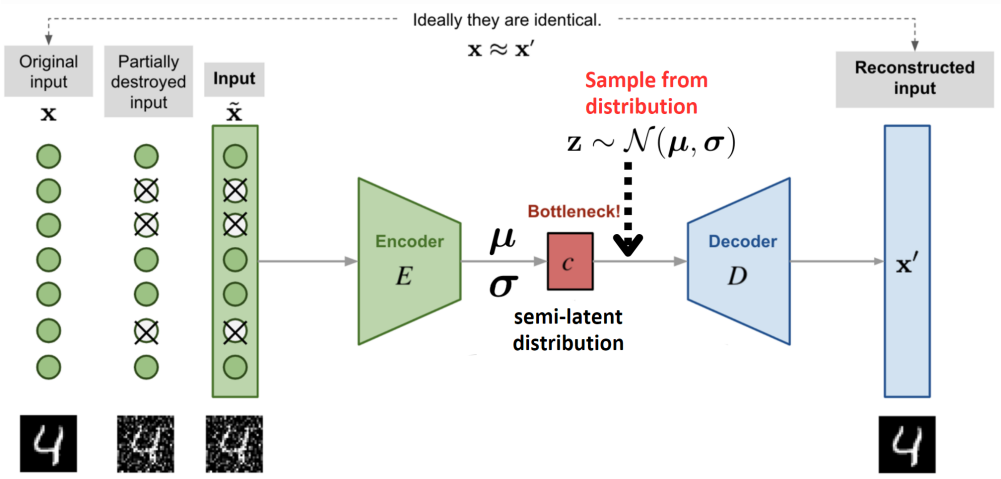


Figure 7: The architectures of a denoising probabilistic autoencoder

As shown by Figure 7, the architecture of DPAE is similar to that of DAE. The main difference is that the encoder in DPAE outputs the encoded representation (the vector $\boldsymbol{\mu}$) and the variance $\boldsymbol{\sigma}$, which is a hyperparameter. Then we sample the vector $\mathbf{z}$ from the semi-latent distribution, i.e., $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$. We call it semi-latent distribution because we want to distinguish it from the latent distribution in a variational autoencoder. In VAE, the encoder outputs the mean $\boldsymbol{\mu}$ and the variance $\boldsymbol{\sigma}$ and neither $\boldsymbol{\mu}$ nor $\boldsymbol{\sigma}$ is a hyperparameter. However, in DPAE, the variance $\boldsymbol{\sigma}$ is a hyperparameter. Next, the vector $\mathbf{z}$ is passed to the decoder and the decoder outputs the reconstructed version of the input. What we want to emphasize is that the reparameterization trick plays an important role in both VAE and DPAE. Let's say $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$. Then we have

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma}\boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The reparameterization trick is a powerful engineering trick and we have implemented it for our DPAE in Python.

```
def sampling(args):
    """Reparameterization trick by sampling from an isotropic unit Gaussian
    # Arguments:
        args (tensor): mean and log of variance of Q(z|X)
    # Returns:
        z (tensor): sampled latent vector
    """
    z_mean, z_log_var = args
    batch = K.shape(z_mean)[0]
    dim = K.shape(z_mean)[1]
    epsilon = K.random_normal(shape=(batch, dim))
    return z_mean + K.exp(0.5 * z_log_var) * epsilon
```

```
z = Lambda(sampling, output_shape=(encoding_dim2,), name='z')([z_mean,
                                        z_log_var])
```

The above code also shows how we sample the vector $\mathbf{z}$ from the Gaussian distribution. By implementing DPAE and the same MLP, we find that the $R^2$ value has increased by 0.01428 compared with the original DAE. The $R^2$ value is now greater than 98%.

| MSE | $R^2$ statistic |
|---|---|
| 0.01679 | 0.98010 |

Table 3: The MSE and $R^2$ statistic of the DPAE and the same MLP

The result above shows that by introducing probabilistic method to the algorithm we are able to further improve the performance of the current DAE.

## 4.4 MLP model optimization

The Multilayer Perceptron (MLP) models were optimized downstream of DAE and DPAE model respectively. We have considered changing activation function, batch size, and number of hidden layers and neurons. After optimization, the MLP model for DAE has a batch size of 3, an activation function of ReLU, and 600 hidden neurons at the last encoder layers. The MLP model for DPAE has a batch size of 2, an activation function of ReLU, and 400 hidden neurons at the last encoder layer. The process of testing optimization are in Figure 7.

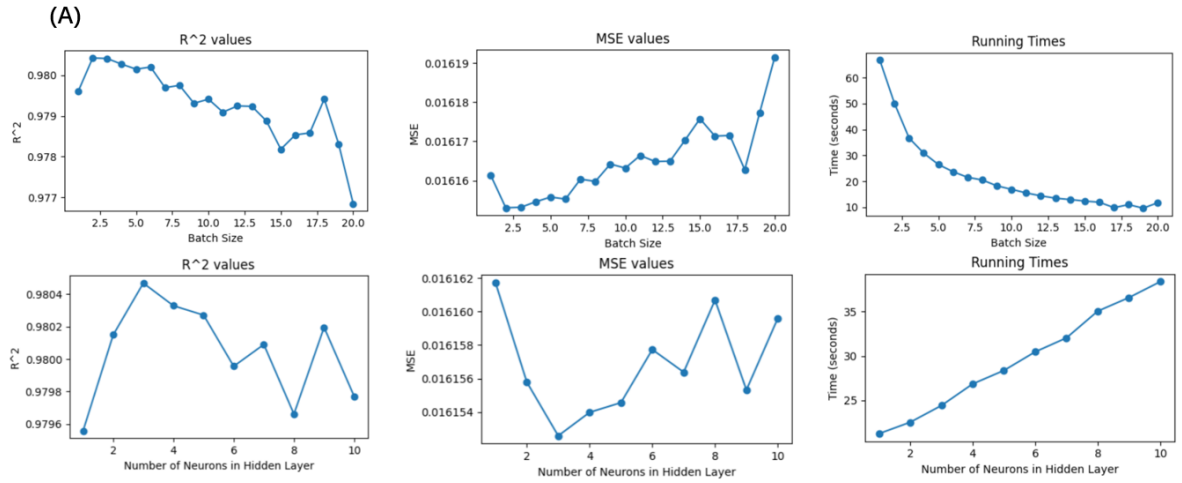


Figure 8: The optimization for DPAE model.

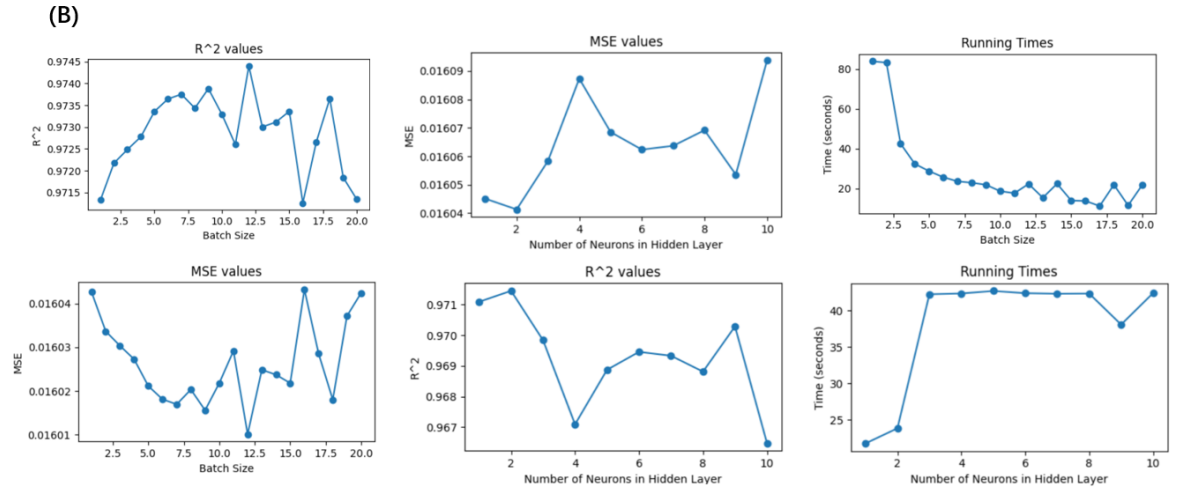


Figure 9: The optimization for DAE model.

# 5 Result

Please note that in some plots we use the term DVAE and by DVAE we are referring to denoising probabilistic autoencoder (DPAE) that we proposed before.

## 5.1 The DPAE-MLP model exceeds the current DAE-MLP and VAE models in gene expression level prediction

Our DVAE-MLP model is shown to have an average R-squared of 0.98, which is the highest among contemporary DL models, the DDAE-MLP model proposed by Seal et al. (2020). Upon comparison, DPAE outperformed the benchmark models across several key performance indicators. Specifically, the mean squared error (MSE) for DPAE stood at a notably lower value compared to the DAE, which recorded MSEs of 0.0172. This denotes an improvement in the predictive accuracy of gene expression levels.

We also tested the DPAE-MLP model against the ovarian cancer datasets. Our model demonstrated an R-squared of 0.9803, confirming that our model is likely to have a good performance in other cancer DNA methylome and CNV datasets.

We are interested in if DPAE is also better than DDAE in cancer classification. Therefore, DPAE-MLP and DDAE-MLP models using sigmoid as the last activation function to do a binary classification on cancer. The result shows that the DPAE-MLP model has a 90.4% training accuracy and 90.1% testing accuracy, while the DDAE-MLP model has 91.3% training accuracy and 87.7% testing accuracy. Therefore, it is likely that DPAE is a more accurate dimension reduction algorithm compared to DDAE not only in gene expression prediction but also in phenotypical classification.

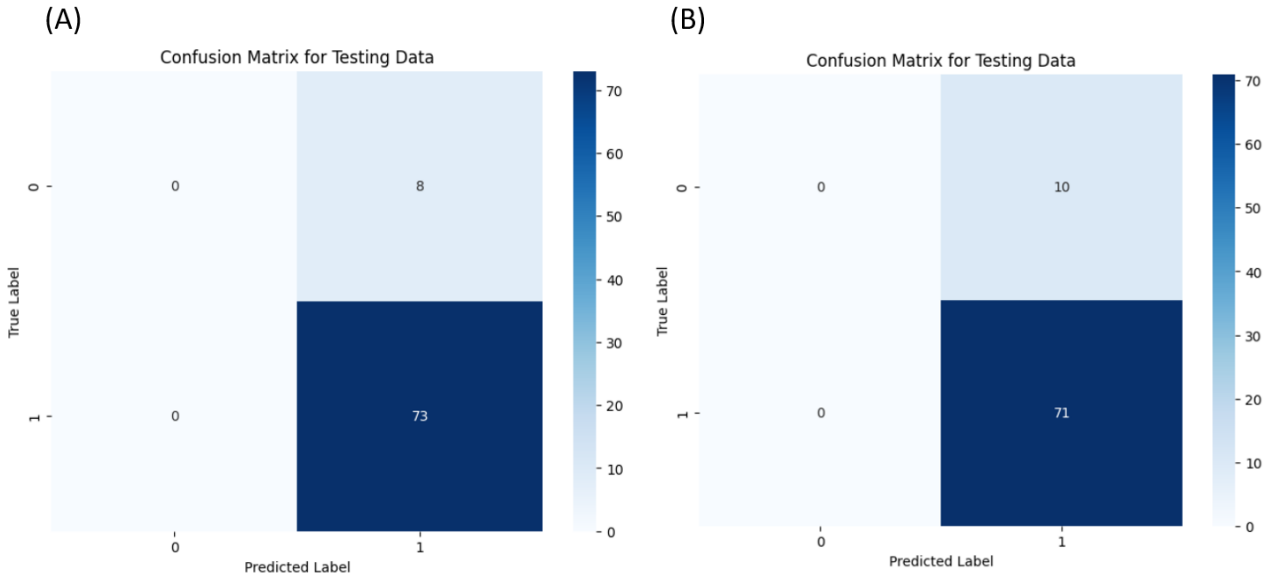


Figure 10: The confusion matrix of DPAE-MLP model and DDAE-MLP model in cancer classification

## 5.2 Ridge regression model better predicts gene expression level compared to DL models

The results of the comparative analysis between Ridge Regression and various deep learning algorithms provided compelling insights into the predictive modeling of gene expression. Ridge Regression emerged as the more effective technique when evaluated against all other algorithms, including DDAE-MLP, and DPAE-MLP. The evaluation metrics were mean squared error (MSE), R-squared ($R^2$), and computational efficiency.

For MSE, Ridge Regression reported a lower score of 0.0124, whereas the DPAE-MLP and DDAE-MLP recorded higher errors of 0.0163, and 0.0171 respectively. The $R^2$ values further consolidated Ridge Regression's superiority, with a score of 0.9821, in contrast to DPAE-MLP's 0.9801 and DDAE-MLP's 0.9777. These metrics clearly demonstrate the enhanced accuracy and consistency of Ridge Regression in capturing the variance in gene expression data.

In terms of computational efficiency, Ridge Regression required significantly less time to both train and predict, clocking in 5 seconds for the entire dataset, while the least efficient DDAE-MLP took upwards of 27 seconds, followed by DPAE-MLP at 54 seconds. This acceleration in performance underlines the practical utility of Ridge Regression for

large-scale applications. Notably, the efficient handling of multicollinearity by Ridge Regression contributed to its robustness against overfitting—a common issue in complex biological datasets—further substantiating its superiority in precision and computational resource utilization. We also think that the seemingly linear relationship between DNA methylation, CNV, and gene expression might contribute to the good performance of Ridge regression model.

Conclusively, the empirical evidence clearly favored Ridge Regression, reestablishing it as a potent and efficient predictor of gene expression, outstripping its deep learning counterparts in both accuracy and computational economy.

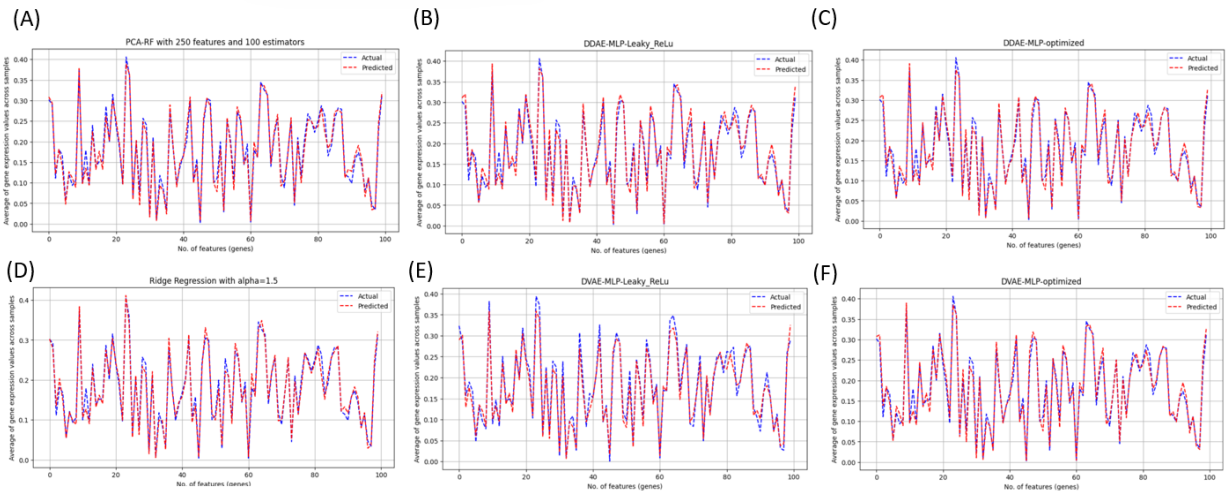


Figure 11: The curves of predicted and actual gene expression value of multiple machine learning model

***Table 3**. A summary model performance of the tested DL and conventional ML models.*

| *Algorithm* | Mean_MSE | Min_MSE | Mean_R_squared | Max_R_squared | Mean_Time | Min_Time |
|---|---|---|---|---|---|---|
| DDAE-MLP | 0.0162 | 0.0151 | 0.9688 | 0.9758 | 38.278 | 37.62 |
| *DDAE-MLP_optimized* | 0.0171 | 0.0164 | 0.9777 | 0.9807 | 26.6156 | 23.09 |
| *DDAE_MLP_Leaky_ReLU* | 0.0172 | 0.0168 | 0.9714 | 0.9788 | 33.428 | 30.7 |
| *DVAE-MLP_Leaky_ReLU* | 0.0164 | 0.0159 | 0.9786 | 0.9808 | 53.2325 | 51.09 |
| *DVAE-MLP_optimized* | 0.0163 | 0.0157 | 0.9801 | 0.9804 | 53.6555 | 48.64 |
| *KNN_10* | 0.0148 | 0.014 | 0.9433 | 0.9541 | 0.0879 | 0.0853 |
| *KNN_15* | 0.0149 | 0.0141 | 0.9436 | 0.9577 | 0.0866 | 0.0846 |
| *KNN_20* | 0.0149 | 0.014 | 0.9426 | 0.9588 | 0.0871 | 0.0848 |
| *KNN_25* | 0.015 | 0.0141 | 0.9413 | 0.9578 | 0.087 | 0.0851 |
| *KNN_5* | 0.0153 | 0.0146 | 0.9451 | 0.9575 | 0.0871 | 0.0853 |
| *Linear regression* | 0.0132 | 0.0125 | 0.982 | 0.9845 | 36.9867 | 35.51 |
| *PCA_MLP* | 0.0142 | 0.0128 | 0.9653 | 0.9714 | 13.4899 | 13.1695 |
| *PCA_Random Forest_estimator_10* | 0.0154 | 0.0147 | 0.9635 | 0.9789 | 36.0822 | 35.25 |
| *PCA_Random Forest_estimator_100* | 0.0145 | 0.0142 | 0.9684 | 0.9787 | 252.3851 | 239.26 |
| *PCA_Random Forest_estimator_150* | 0.0145 | 0.0142 | 0.9684 | 0.9782 | 373.584 | 353.73 |
| *PCA_Random Forest_estimator_200* | 0.0145 | 0.014 | 0.9679 | 0.9781 | 502.8862 | 475.46 |
| *PCA_Random Forest_estimator_50* | 0.0146 | 0.0143 | 0.968 | 0.9787 | 132.7146 | 130.04 |
| *Ridge_alpha_0.01* | 0.0124 | 0.0112 | 0.9821 | 0.9845 | 5.5255 | 5.09 |
| *Ridge_alpha_0.1* | 0.0124 | 0.0112 | 0.9821 | 0.9845 | 5.2281 | 4.83 |
| *Ridge_alpha_0.5* | 0.0124 | 0.0112 | 0.9821 | 0.9845 | 5.469 | 4.98 |
| *Ridge_alpha_1* | 0.0124 | 0.0112 | 0.9821 | 0.9845 | 5.4808 | 4.91 |
| *Ridge_alpha_1.5* | 0.0124 | 0.0112 | 0.9821 | 0.9845 | 5.0555 | 4.9065 |

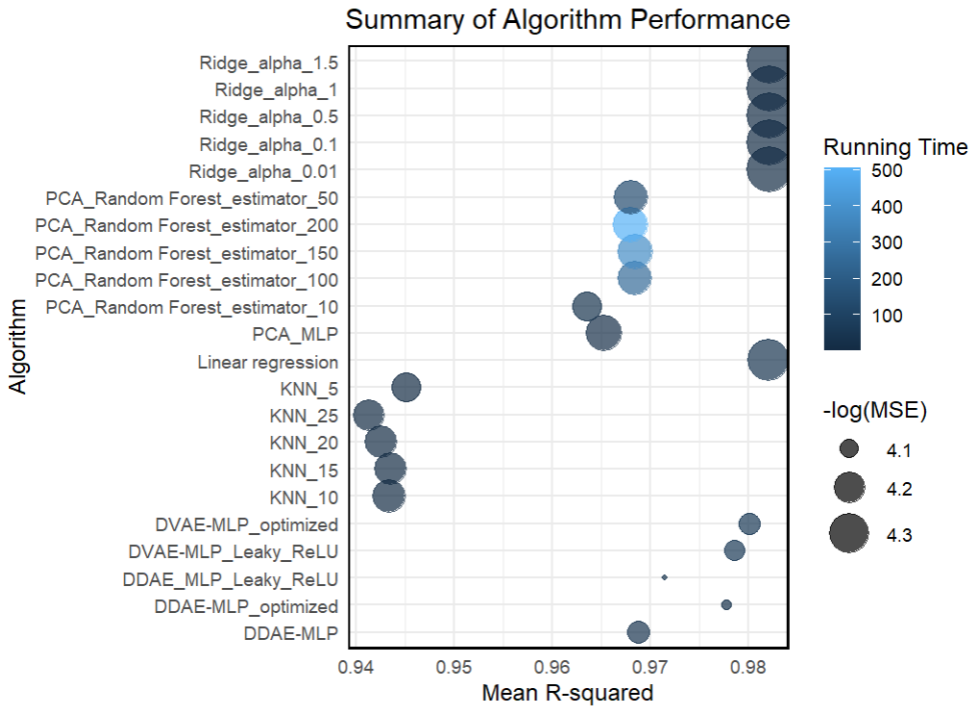


Figure 12: A summary of algorithm performance based on MSE, R-squared, and running time

# 6 Conclusion and discussion

Our project investigates effective computational models for predicting gene expression levels using DNA methylation and copy number variation (CNV) data. With the growing accumulation of multi-omics data in biomedical research, our goal is to develop and validate models that outperform the existing Denoising Autoencoders (DAEs) in gene expression prediction. We have provided a detailed evaluation of deep learning autoencoder methods against traditional machine learning approaches through cross-validation, error analysis, and computational efficiency checks. Drawing inspiration from Variational Autoencoders (VAEs), we proposed an original model called the denoising probabilistic autoencoder (DPAE), which surpasses the performance of the current DAE models. In the new DPAE, we introduce randomness to the encoded representations by sampling the vector $\mathbf{z}$ from the semi-latent distribution and thus we have improved the overall performance. Additionally, we found that ridge regression offers advantages such as lower mean squared error (MSE), higher $R^2$ values, and shorter computational times. These findings contest the prevalent application of deep learning in multi-omics data analysis by underscoring the effectiveness and efficiency of ridge regression in predicting gene expression. These insights are crucial for advancing cancer diagnostic techniques, providing a more cost-effective and faster alternative to traditional RNA-seq diagnostics. We also found that some linear regression models with regularizations can outperform deep learning models in terms of the $R^2$ value, MSE, and running time.

Furthermore, to make sure our refined model and other conventional machine learning model also works on other types of cancer datasets, we downloaded and processed the same data from Hira et al.(2021), which works in autoencoder model on ovarian cancer subtype classification. The data was downloaded directly from TCGA websites and preprocessed in R using hg19 annotation and GenomicRanges (1.52.1). The $R^2$ are around 98% for the linear regression model and our optimized model.

The future work includes the following tasks:

- Further improving the performance of denoising probabilistic autoencoder (DPAE) by learning the hyperparameter $\boldsymbol{\sigma}$ instead of setting $\boldsymbol{\sigma}$ a constant.
- Further improving the performance of the multilayer perceptron (MLP) or replacing MLP by other regression models.
- Applying our DPAE and the conventional machine learning models to other problems in biology and even some different field.

# 7 Contribution Statements

## 7.1 Zhujun Yao

Zhujun proposed the project topic and did literature research, proposed AE models, preprocessed LIHC and OV data, optimized MLP models, collected and analyzed the results, and plotted the table & graphs.

## 7.2 Qixuan Wang

Qixuan investigated the original DAE, proposed the improved DAE with its code. Inspired by VAE, he also originally proposed the denoising probabilistic autoencoder (DPAE),

which increased the $R^2$ value to over 0.98, and implemented the code.

## References


Brewer, T., Yehia, L., Bazeley, P., & Eng, C. (2023). Integrating somatic CNV and gene expression in breast cancers from women with PTEN hamartoma tumor syndrome. *Npj Genomic Medicine*, *8*(1), 1–10. https://doi.org/10.1038/s41525-023-00361-0

Byron, S. A., Van Keuren-Jensen, K. R., Engelthaler, D. M., Carpten, J. D., & Craig, D. W. (2016). Translating RNA sequencing into clinical diagnostics: Opportunities and challenges. *Nature Reviews Genetics*, *17*(5), 257–271. https://doi.org/10.1038/nrg.2016.10

Choi, J., & Chae, H. (2020). methCancer-gen: A DNA methylome dataset generator for user-specified cancer type based on conditional variational autoencoder. *BMC Bioinformatics*, *21*(1), Article 1. https://doi.org/10.1186/s12859-020-3516-8

Diallo, B., Hu, J., Li, T., Khan, G. A., Liang, X., & Zhao, Y. (2021). Deep embedding clustering based on contractive autoencoder. *Neurocomputing*, *433*, 96–107. https://doi.org/10.1016/j.neucom.2020.12.094

Franco, E. F., Rana, P., Cruz, A., Calderón, V. V., Azevedo, V., Ramos, R. T. J., & Ghosh, P. (2021a). Performance Comparison of Deep Learning Autoencoders for Cancer Subtype Detection Using Multi-Omics Data. *Cancers*, *13*(9), Article 9. https://doi.org/10.3390/cancers13092013

Franco, E. F., Rana, P., Cruz, A., Calderón, V. V., Azevedo, V., Ramos, R. T. J., & Ghosh, P. (2021b). Performance Comparison of Deep Learning Autoencoders for Cancer Subtype Detection Using Multi-Omics Data. *Cancers*, *13*(9), Article 9.

https://doi.org/10.3390/cancers13092013

Hira, M. T., Razzaque, M. A., Angione, C., Scrivens, J., Sawan, S., & Sarker, M. (2021). Integrated multi-omics analysis of ovarian cancer using variational autoencoders. *Scientific Reports*, *11*(1), 6265. https://doi.org/10.1038/s41598-021-85285-4

Kim, S., Park, H. J., Cui, X., & Zhi, D. (2020). Collective effects of long-range DNA methylations predict gene expressions and estimate phenotypes in cancer. *Scientific Reports*, *10*(1), 3920. https://doi.org/10.1038/s41598-020-60845-2

Mohanty, V., Wang, F., Mills, G. B., & Chen, K. (2021). Uncoupling of gene expression from copy number presents therapeutic opportunities in aneuploid cancers. *Cell Reports Medicine*, *2*(7), 100349. https://doi.org/10.1016/j.xcrm.2021.100349

Odenkirk, M. T., Reif, D. M., & Baker, E. S. (2021). Multiomic Big Data Analysis Challenges: Increasing Confidence in the Interpretation of Artificial Intelligence Assessments. *Analytical Chemistry*, *93*(22), 7763–7773. https://doi.org/10.1021/acs.analchem.0c04850

Pokrovac, I., Rohner, N., & Pezer, Ž. (2023). *The prevalence of copy number increase at multiallelic CNVs associated with cave colonization*. https://doi.org/10.1101/2023.11.10.566513

Rashid, S., Shah, S., Bar-Joseph, Z., & Pandya, R. (2021). Dhaka: Variational autoencoder for unmasking tumor heterogeneity from single cell genomic data. *Bioinformatics (Oxford, England)*, *37*(11), 1535–1543. https://doi.org/10.1093/bioinformatics/btz095

Razin, A., & Cedar, H. (1991). DNA methylation and gene expression. *Microbiological Reviews*, *55*(3), 451–458.

Seal, D. B., Das, V., Goswami, S., & De, R. K. (2020). Estimating gene expression from DNA methylation and copy number variation: A deep learning regression model for multi-omics integration. *Genomics*, *112*(4), 2833–2841. https://doi.org/10.1016/j.ygeno.2020.03.021

Shao, X., Lv, N., Liao, J., Long, J., Xue, R., Ai, N., Xu, D., & Fan, X. (2019). Copy number variation is highly correlated with differential gene expression: A pan-cancer study. *BMC Medical Genetics*, *20*(1), 175. https://doi.org/10.1186/s12881-019-0909-5

Singhal, S. K., Usmani, N., Michiels, S., Metzger-Filho, O., Saini, K. S., Kovalchuk, O., & Parliament, M. (2015). Towards understanding the breast cancer epigenome: A comparison of genome-wide DNA methylation and gene expression data. *Oncotarget*, *7*(3), 3002–3017. https://doi.org/10.18632/oncotarget.6503

Taguchi, Y. -h, Komaki, S., Sutoh, Y., Ohmomo, H., Otsuka-Yamasaki, Y., & Shimizu, A. (2023). Integrated analysis of human DNA methylation, gene expression, and genomic variation in iMETHYL database using kernel tensor decomposition-based unsupervised feature extraction. *PLOS ONE*, *18*(8), e0289029. https://doi.org/10.1371/journal.pone.0289029

Titus, A., Wilkins, O., Bobak, C., & Chrisetensen, B. (2018). *An unsupervised deep learning framework with variational autoencoders for genome-wide DNA*

*methylation analysis and biologic feature extraction applied to breast cancer*. https://doi.org/10.1101/433763

Tsimenidis, S., Vrochidou, E., & Papakostas, G. A. (2022). Omics Data and Data Representations for Deep Learning-Based Predictive Modeling. *International Journal of Molecular Sciences*, *23*(20), Article 20. https://doi.org/10.3390/ijms232012272

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. *Proceedings of the 25th International Conference on Machine Learning*, 1096–1103. https://doi.org/10.1145/1390156.1390294

Yassi, M., Chatterjee, A., & Parry, M. (2023). Application of deep learning in cancer epigenetics through DNA methylation analysis. *Briefings in Bioinformatics*, *24*(6). https://doi.org/10.1093/bib/bbad411

Zhong, H., Kim, S., Zhi, D., & Cui, X. (2019). Predicting gene expression using DNA methylation in three human populations. *PeerJ*, *7*, e6757. https://doi.org/10.7717/peerj.6757