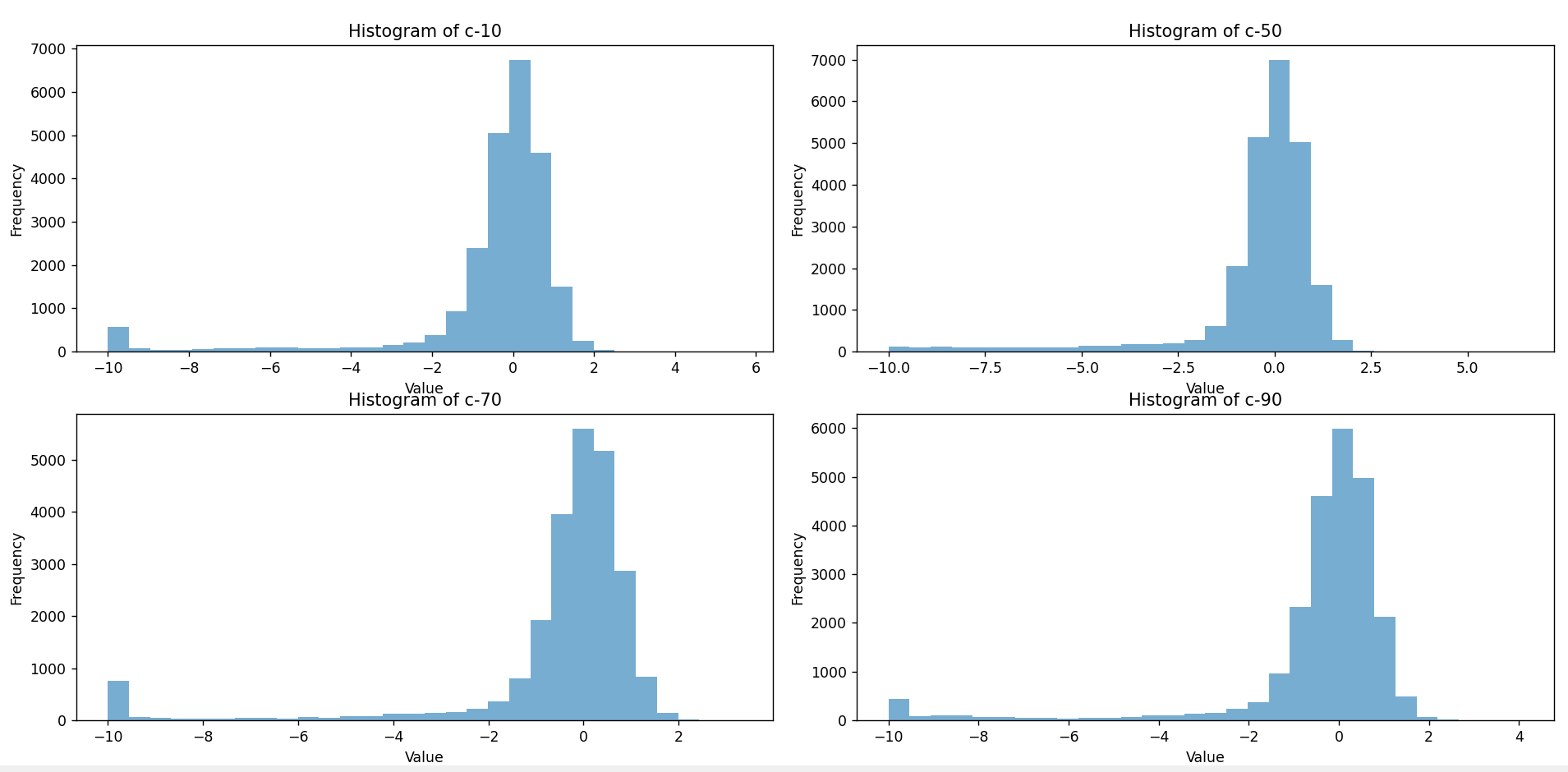
# We obtained our data from the LISH-MoA competition on Kaggle, which contains more than 23,000 compounds. For each compound, we have gene expression and cell viability data available. The MoA annotations in the dataset are binary, meaning they are either present or absent. Our dataset has two types of features: gene expression (represented by columns g-0 to g-771) and cell viability (represented by columns c-0 to c-99)

# EDA:



The distributions of the selected c- features appear to be approximately normal (bell-shaped) with some slight deviations. This suggests that cell viabilities in these features follow a common pattern, and extreme values are less frequent. The histograms show that the peak (mode) for each feature is around 0, which could be a result of the data normalization process applied to the dataset. There is a noticeable range in the x-axis values for these features, spanning from around -10 to 6, high negative values indicate a high number of dead cells, while high positive values indicate a high number of living cells. Each feature has a slightly different spread, as seen in the width of the histograms. This may indicate varying levels of sensitivity or response to the treatments across different cell viabilities.

Chart, line chart, histogram

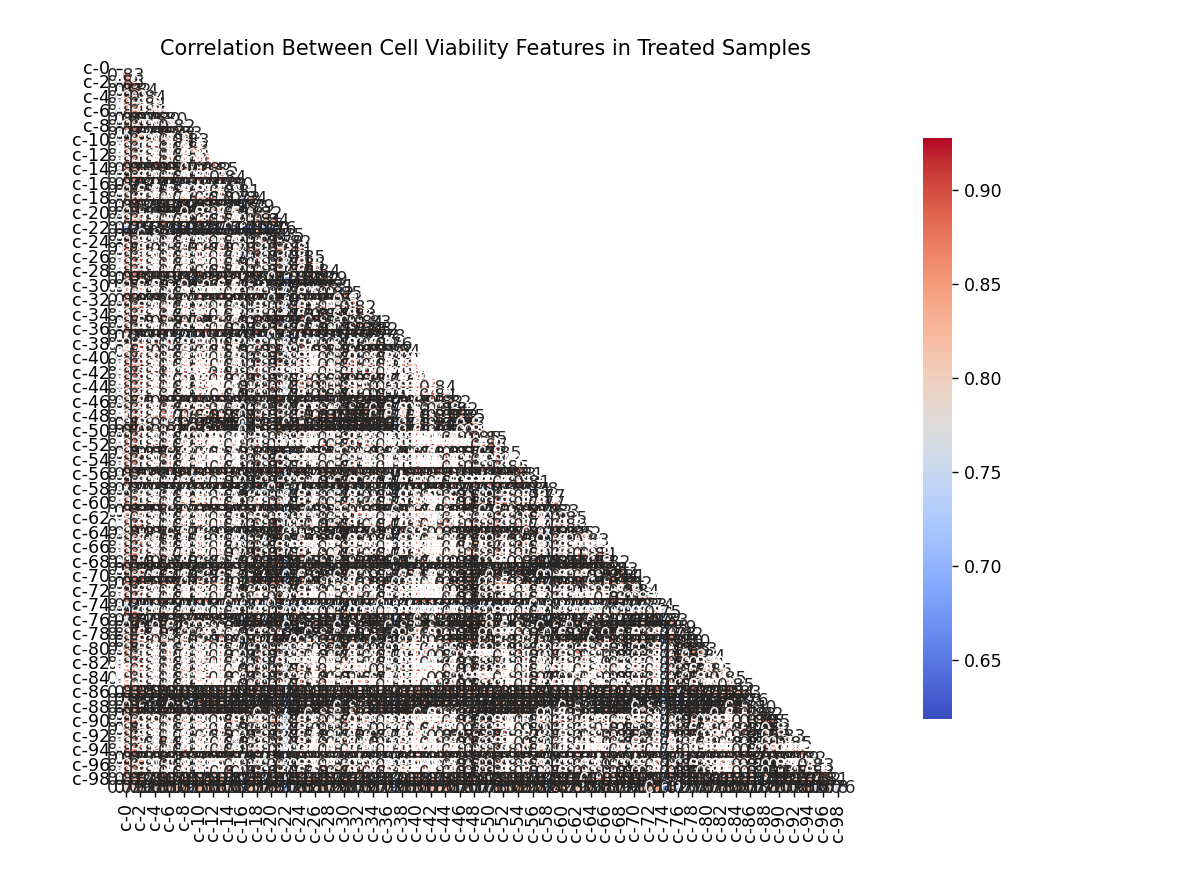
Description automatically generated

The above KDE plot helps visualize the impact of treatments on cell viability for a specific feature (c-30) by comparing it to control samples. The differences in the distributions provide insights into how various treatments may affect cell viability across samples. The treated samples exhibit a broader and slightly more evenly distributed range of cell viability values compared to the control samples. This indicates that the treatment has a more diverse effect on cell viability in various samples. The control samples show a sharp peak around 0, suggesting that in the absence of treatment, cell viability tends to remain stable and less variable. The treated samples' distribution reveals a wider spread, with more extreme values on both the positive and negative sides. This suggests that some treatments might have a significant impact on cell viability, causing either an increase in the number of living cells or a decrease leading to cell death.

Chart, line chart

Description automatically generated

The distribution of cell viability values varies across different treatment durations. This suggests that treatment time has a noticeable effect on cell viability for the selected feature. The plot illustrating the impact of treatment time on cell viability for a specific feature reveals that treatment duration has a considerable influence on cell viability. The observed variability in cell viability values across different treatment times emphasizes the need to account for treatment duration when analyzing the effects of treatments and developing predictive models.



The heatmap plot of the correlation between cell viability features in treated samples reveals a wide range of relationships, from strong positive to negative correlations. These correlations provide valuable insights into how cell viability features interact and respond to various treatments, which can help guide future analysis and model development.

Graphical user interface, application

Description automatically generated

Chart, histogram

Description automatically generated

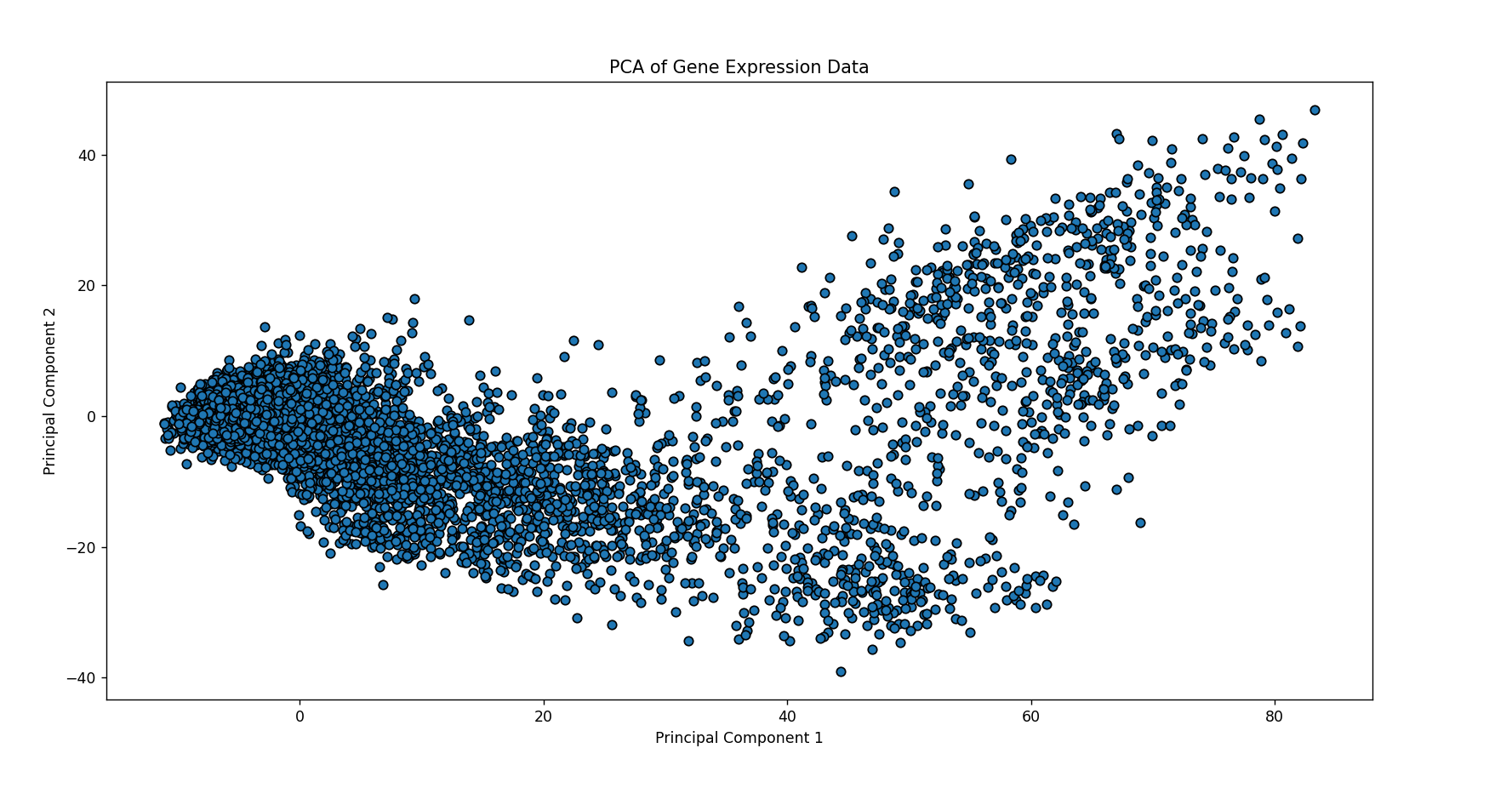
The range of gene expression values is quite wide, spanning from significantly negative to positive values. This demonstrates that the dataset contains genes with varying degrees of activation or inhibition. The gene expression values seem to have a roughly symmetrical distribution around zero, suggesting that many genes have balanced expression levels, with some being upregulated (positive values) and others being downregulated (negative values). Certain genes exhibit a higher concentration of expression values close to zero, which could mean that they have a lower influence or less variation in expression across the samples. On the other hand, some genes have wider distributions, indicating a greater variability in their expression levels across different samples. These genes might be more sensitive to distinct treatments or experimental conditions.

A picture containing diagram

Description automatically generated

Graphical user interface, application

Description automatically generated



PCA effectively reduces the dimensionality of the gene expression dataset, allowing us to visualize the samples in a lower-dimensional space. The first few principal components account for a significant proportion of the total variance in the dataset, indicating that they capture the most prominent patterns and trends in gene expression data. By plotting the samples according to their principal component scores, we can observe the separation between different experimental conditions or treatment groups. This suggests that PCA can help identify relationships and distinctions between various samples. Despite, the dimensionality reduction, some overlap between groups might still be present, indicating that the first few principal components do not fully capture all the differences between samples. However, the separation achieved is still informative for a preliminary analysis.

**2.4 Data Pre-processing Steps**

Before training the machine learning models, the following pre-processing steps were performed on the dataset:

1. For classic ML algorithms.

Merging the data: The train\_features and train\_target\_scored datasets were merged on the "sig\_id" column.

Separating categorical columns: The categorical columns 'cp\_type', 'cp\_time', and 'cp\_dose' were separated from the numerical columns for preprocessing.

Standardizing the numerical data: The gene expression data was standardized using the RobustScaler. This ensures that all gene expression features are on a similar scale and prevents any one feature from dominating the model's predictions.

Dimensionality reduction using PCA: Principal Component Analysis (PCA) was applied to the gene expression data, reducing the dimensionality while retaining 95% of the variance in the data. This step helps visualize the dataset and potentially speeds up the machine learning algorithms.

Merging the preprocessed data: The scaled gene expression data, cell viability data, and one-hot encoded categorical data were merged into a single dataset.

Splitting the data: The data was split into training and validation sets using an iterative stratification approach with an 80-20 ratio, which ensures that the distribution of MoA annotations in both sets is similar.

Scaling and dimensionality reduction on the entire dataset: The merged dataset was then scaled using the RobustScaler, and PCA was applied to reduce dimensionality while retaining 95% of the variance in the data.

**Performance Metric**

The performance of the machine learning models is evaluated using the log loss metric. Log loss, also known as logarithmic loss or cross-entropy loss, measures the performance of a classification model where the predicted output is a probability value between 0 and 1. The log loss for each model is calculated using the log\_loss function from the sklearn.metrics library.

**Comparison of the Results Across Different Models**

|  |  |
| --- | --- |
| Model | Log Loss |
| Logistic Regression | 4.750906931715915 |
| Random Forest | 0.64373824828473726 |

# Summary:

This report presents an analysis of the Mechanism of Action (MoA) dataset from the Lish-MoA Kaggle competition. The dataset contains gene expression and cell viability data for thousands of compounds tested in various conditions, along with information about their MoA. The report includes an exploratory data analysis of the dataset, using visualizations and statistical summaries to gain insights into the relationships between the features and the MoA labels. The analysis also includes a Principal Component Analysis (PCA) to reduce the dimensionality of the gene expression data and visualize it in two dimensions. The report then presents an ensemble model that combines Logistic Regression, Support Vector Machines (SVM), Random Forest, and XGBoost classifiers to predict the MoA labels of the compounds. The performance of the model is evaluated using accuracy scores on a validation set. Overall, the report demonstrates the potential of machine learning models to predict the MoA of compounds based on gene expression and cell viability data, which could have important applications in drug discovery and development.

Code % = 100 \* (30 – 12) / (30 + 17)= 38%

References:

<https://www.kaggle.com/competitions/lish-moa/discussion/184005>

<https://www.kaggle.com/code/headsortails/explorations-of-action-moa-eda>

<https://www.kaggle.com/code/isaienkov/mechanisms-of-action-moa-prediction-eda>

<https://www.kaggle.com/code/gsdeepakkumar/moa-basic-eda-clustering>