Fish Toxicity Data Analysis and Modelling

Introduction

This project is focused on the analysis and modelling of a dataset containing chemical toxicity data. The primary goal is to understand the relationships between various chemical descriptors and their toxicity levels, measured as LC50 [-LOG (mol/L)]. By employing a range of data preprocessing techniques, regression, classification, and clustering models, this project aims to derive meaningful insights and patterns from the dataset.

1. Project Overview

The project aims to analyse a dataset to understand its characteristics and derive meaningful insights through preprocessing, modelling, evaluation, and clustering. The dataset includes several features, such as CIC0, SM1_Dz(Z), GATS1i, NdsCH, NdssC, MLOGP, and the response variable LC50 [-LOG (mol/L)].

2. Data Preprocessing

Handling Missing Values:

- Missing values in the dataset were identified and imputed rather than deleted to prevent data loss and ensure that the dataset remains comprehensive. Various imputation methods, such as mean, median, or mode imputation, were considered based on the nature of the missing data

Standardization and Normalization:

- The dataset was standardized to ensure that each feature contributed equally to the analysis. This involved scaling the data to have a mean of zero and a standard deviation of one, which is crucial for algorithms sensitive to the scale of data, like regression and clustering models.

Feature Engineering:

- New features were created, and existing ones were modified to improve the performance of the model. This step included combining features, transforming features, or creating interaction terms to enhance the model's predictive power.

Data Splitting:

- The dataset was split into training and testing sets to evaluate the model's performance. Typically, a common split ratio such as 70-30 or 80-20 was used to ensure that the model was trained on a sufficient amount of data while being tested on a separate subset to validate its performance.

3. Regression Modelling

Multiple Linear Regression (MLR)

Description:

- Multiple Linear Regression (MLR) models the relationship between a dependent variable and multiple independent variables by fitting a linear equation to the observed data.

Performance:

- **Regression Metrics:**
  - **Mean Absolute Error (MAE):** Measures the average magnitude of the errors in a set of predictions, without considering their direction.
  - **Mean Squared Error (MSE):** Measures the average of the squares of the errors, giving a higher weight to large errors.
  - **R2 Score:** Represents the proportion of the variance for the dependent variable that's explained by the independent variables in the model.
- **Evaluation Results:** The MLR model showed reasonable performance with moderate MAE and MSE values, and an R2 score indicating a good fit but with some room for improvement in prediction accuracy.

Ridge Regression

Description:

- Ridge Regression is a type of linear regression that includes a regularization term (L2 penalty) to prevent overfitting by shrinking the coefficients of less important features.

Performance:

- **Regression Metrics:** Similar metrics as MLR but typically with lower variance and better generalization on the test set due to regularization.
- **Evaluation Results:** Ridge Regression performed better than MLR in terms of reducing overfitting, with slightly improved R2 scores and lower MSE values, indicating more reliable predictions.

Lasso Regression

Description:

- Lasso Regression (Least Absolute Shrinkage and Selection Operator) is a linear regression technique that includes an L1 penalty to both prevent overfitting and perform feature selection by shrinking some coefficients to zero.

Performance:

- **Regression Metrics:** Similar to Ridge Regression but with the added benefit of automatic feature selection.
- **Evaluation Results:** Lasso Regression demonstrated effective feature selection and comparable performance to Ridge Regression, with improved interpretability and reduced complexity in the model.

4. Classification Modelling

Naive Bayes (Boosting)

Description:

- Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming independence between predictors. Boosting improves its performance by combining multiple weak learners into a strong learner.

Performance:

- **Classification Metrics:**
  - **Precision:** Proportion of true positive predictions among all positive predictions.
  - **Recall:** Proportion of true positive instances correctly identified by the model out of all actual positive instances.
  - **F1-Score:** Harmonic mean of precision and recall.

- **Evaluation Results:** The Naive Bayes (Boosting) model achieved high precision and recall, with an F1-Score indicating a strong balance between the two metrics, making it effective for this dataset

Voting Classification

Description:

- Voting Classification combines multiple models to improve performance by taking a majority vote for classification tasks.

Performance:

- **Classification Metrics:** Similar to Naive Bayes but with improved robustness and reduced overfitting.
- **Evaluation Results:** The Voting Classification model had the highest F1-Score of 0.7824, showing strong predictive capability and reliability.

5. Clustering

K-means Clustering

Description:

- K-means Clustering partitions data into K clusters by minimizing the variance within each cluster. It assigns each data point to the nearest cluster center.

Performance:

- **Clustering Metrics:**
  - **Within-cluster Sum of Squares (WCSS):** Measures the compactness of the clusters.
  - **Silhouette Score:** Measures how similar an object is to its own cluster compared to other clusters.
- **Evaluation Results:** K-means effectively identified distinct clusters within the dataset, with Cluster 1 containing the majority of highly toxic data points.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

Description:

- DBSCAN groups closely packed points and marks points in low-density regions as outliers.

Performance:

- **Clustering Metrics:** Silhouette Score and evaluation of noise points.
- **Evaluation Results:** DBSCAN was effective for identifying clusters of varying shapes and handling noise in the dataset, revealing patterns that K-means might miss.

6. Conclusion

The project provided valuable learning experiences, allowing the application of both new and revised concepts in data preprocessing, modelling, evaluation, and clustering. The analysis revealed significant patterns in the toxicity data, highlighting the distribution of highly, less, and medium toxic data points across different clusters. The insights gained can be used to improve future analyses and model development.

References

- ProjectReport_GroupC_21BCSB78_AdityaRanjanSahu.html