Truc Tran Trung

Evaluation on 'qm7' data for regression using machine learning and deep learning models

August 12, 2024

1. **Introduction:**

   - In this project, we focus on applying advanced Machine Learning techniques to analyze the qm7 dataset, with the primary objective of developing robust regression models. The qm7 dataset consists of various features that we aim to leverage to predict a continuous target variable. Regression analysis is critical in numerous fields, including finance, economics, and engineering, where predicting continuous outcomes based on input variables is essential.
   - To achieve our objective, we explore and implement four different regression models: XGBoost, Support Vector Machines (SVM), Linear Regression, and Graph Neural Networks (GNNs). Each of these models brings unique strengths to the table, and our goal is to compare their performance and determine the most effective approach for the dataset at hand.

2. **Data Overview:**

   - This dataset is a subset of GDB-13 (a database of nearly 1 billion stable and synthetically accessible organic molecules) composed of all molecules of up to 23 atoms (including 7 heavy atoms C, N, O, and S), totalling 7165 molecules. We provide the Coulomb matrix representation of these molecules and their atomization energies computed similarly to the FHI-AIMS implementation of the Perdew-Burke-Ernzerhof hybrid functional (PBE0). This dataset features a large variety of molecular structures such as double and triple bonds, cycles, carboxy, cyanide, amide, alcohol and epoxy.
   - **Features:** The dataset is composed of three multidimensional arrays X (7165 x 23 x 23), T (7165) and P (5 x 1433) representing the inputs (Coulomb matrices), the labels (atomization energies) and the splits for cross-validation, respectively. The dataset also contain two additional multidimensional arrays Z (7165) and R (7165 x 3) representing the atomic charge and the cartesian coordinate of each atom in the molecules.
   - **Target Variable:** The goal is to predict a specific target variable that is crucial for the analysis (Feature T).

3. **Data Preprocessing:**

   - **Loading Data:** The dataset was loaded from a '.mat' file using the 'scipy.io.loadmat' function, which provided access to key matrices such as feature matrix 'X', Cartesian coordinates 'R', atomic numbers 'Z', atomization energies 'T' (label), and other properties 'P'.

- **Reshaping Data:** The matrices ‘R‘ and ‘X‘ were reshaped to flatten them into 2D matrices, with the first dimension representing the number of samples and the second dimension containing the features, making them suitable for input into Machine Learning models.
- **One-Hot Encoding:** The atomic numbers in the ‘Z‘ matrix were one-hot encoded to convert the categorical atomic number data into a binary matrix format. This encoding was crucial for ensuring that the atomic types were represented as distinct features in the model.
- **Concat:** Combine all features into one Feature using hstack.
- **Convert:** convert Features into dataframe and save into CSV file.

	0	1	2	3	4	5	6	7	8	9	...	981	982	983	984	985	986	987	988	989	Label
0	36.858105	2.907633	2.907612	2.907564	2.905349	0.000000	0.000000	0.000000	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5578.0	-417.96
1	36.858105	12.599944	2.902000	1.473118	1.473101	2.901973	2.901886	1.473102	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2504.0	-712.42
2	36.858105	14.261827	1.503703	2.924997	2.924732	1.503680	0.000000	0.000000	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4851.0	-564.21
3	36.858105	15.871878	2.979434	1.401225	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3130.0	-404.88
4	73.516693	17.885317	10.561490	4.355064	2.062530	2.069614	1.581991	1.590780	1.261482	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6670.0	-808.87

5 rows × 991 columns

Figure 1: Dataframe in table format.

4. **Exploratory Data Analysis (EDA):**

- **Distribution of Features:** Histograms were plotted to visualize the distribution of numerical features.

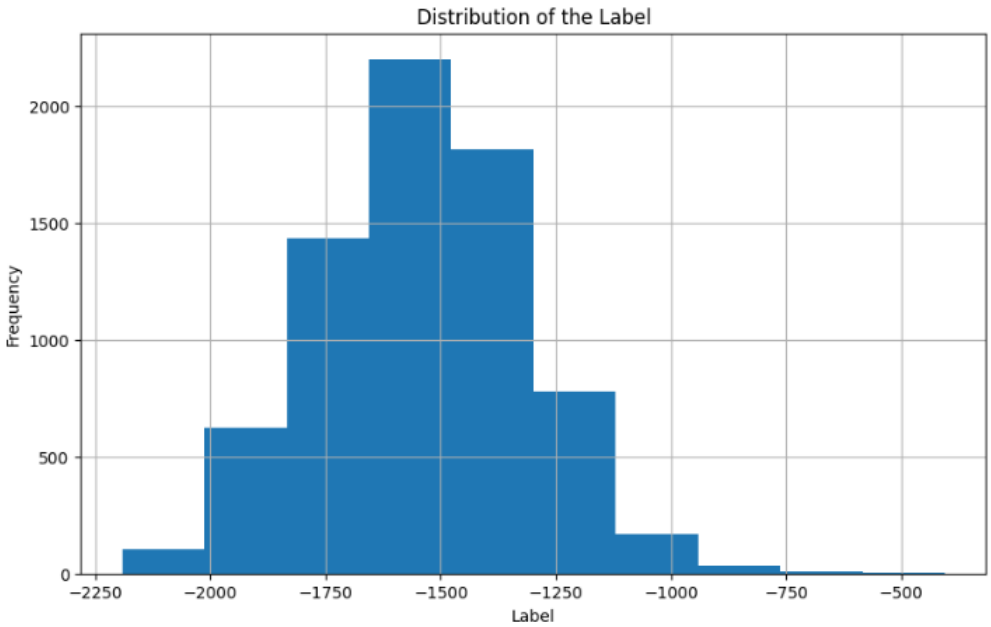

Figure 2: Distribution of Labels.

5. **Model Selection and Training:**

- **Linear Regression:** Used as a baseline regression model to predict the continuous target variable, providing a simple and interpretable approach.

- **Support Vector Machine (SVM):** Implemented for regression tasks (SVR), using both Poly and RBF kernels to find the optimal hyperplane that minimizes error.
- **Kernel Ridge Regression:** Combines Ridge Regression with the kernel trick to capture non-linear relationships between the features and the target variable (Using RBF, Poly, Sigmoid kernels).
- **XGBoost:** An advanced gradient boosting algorithm used to enhance model performance by sequentially building decision trees to minimize prediction error.
- **Graph Neural Networks (GNN):** Applied to leverage potential graph-structured data, capturing complex relationships and dependencies in the dataset.
- **K-Fold Cross-Validation:** Employed to evaluate the performance of each model, ensuring robustness by splitting the dataset into k subsets, training on k-1 subsets, and validating on the remaining subset, iteratively.

6. **Model Evaluation:**

- **Mean Squared Error (MSE):** The MSE was calculated for each model to measure the average squared difference between the observed actual outcomes and the predictions. Both cross-validation (CV) and test set MSEs were computed to evaluate model performance.
- **R-Squared ($\mathbf{R^2}$ Score):** The R^2 score was used to assess the proportion of variance in the dependent variable that is predictable from the independent variables. R^2 scores were calculated for both cross-validation and test sets to understand the explanatory power of each model.
- **Standard Deviation of MSE and $\mathbf{R^2}$:** The standard deviation of MSE and R^2 scores across cross-validation folds was computed to assess the consistency and reliability of the models. Models with lower standard deviation values are considered more stable.
- **Model Comparison:** The models were compared based on their MSE and R^2 scores, both for cross-validation and test datasets. This comparison helps identify the most accurate and generalizable model among those tested.

7. **Results:**

- **Best Performing Model:** Among the models evaluated, the Kernel Ridge model with a polynomial kernel (KernelRidge_poly) demonstrated the best overall performance. It achieved the lowest Mean Squared Error (MSE) on both cross-validation (CV) and test sets, with values of 2.86×10^4 and 1.78×10^4 respectively. Additionally, the R^2 score for this model was 0.466 on the CV set and 0.648 on the test set, indicating a good fit and generalizability. This makes KernelRidge_poly the most effective model in predicting the target variable in this analysis.
- **Comparison of Model Performance:** The XGBoost model also performed exceptionally well, particularly in the cross-validation stage, where it achieved an R^2 score of 0.996 and an MSE of 2.13×10^2, demonstrating its strong ability to capture the underlying patterns in the data. However, test set results for XGBoost are not available, which limits the ability to fully assess its generalization capability.

  Other models such as the Support Vector Regressor with a radial basis function kernel (SVR_rbf) and the polynomial kernel (SVR_poly) showed reasonable performance, with R^2 scores close to zero on the test set, indicating marginal predictive power. The Kernel Ridge model with an RBF kernel (KernelRidge_rbf), however, performed poorly, with a highly negative R^2 score of -7.527 in CV and -6.757 on the test set, signifying a poor fit and overfitting to the training data.

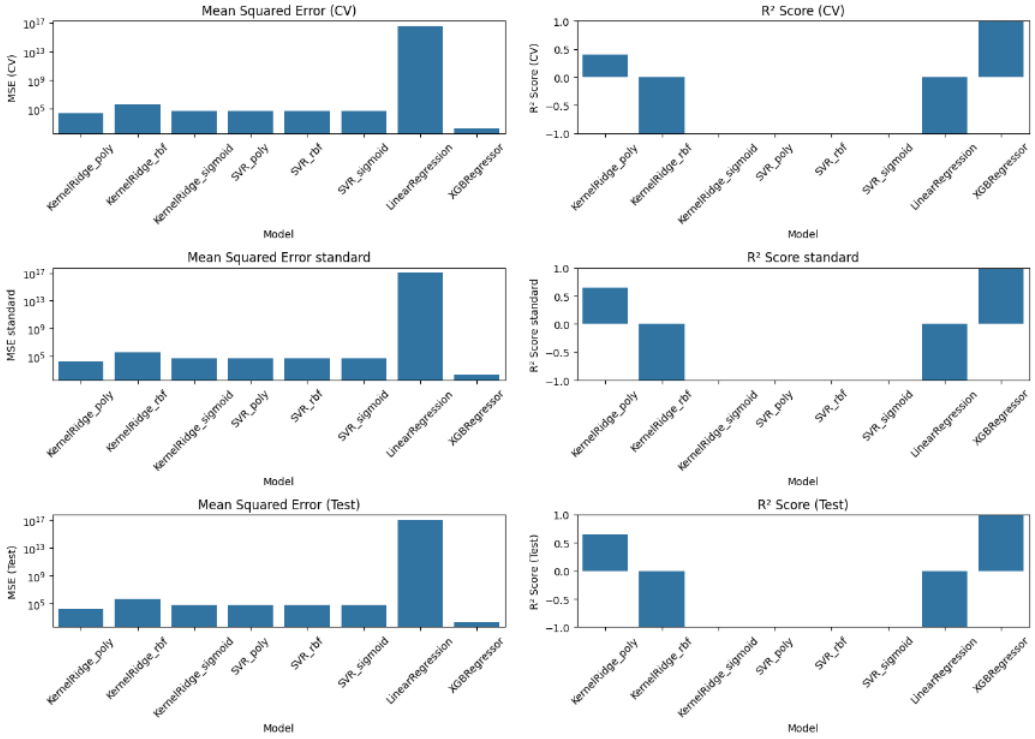

Figure 3: Ressult of the models.

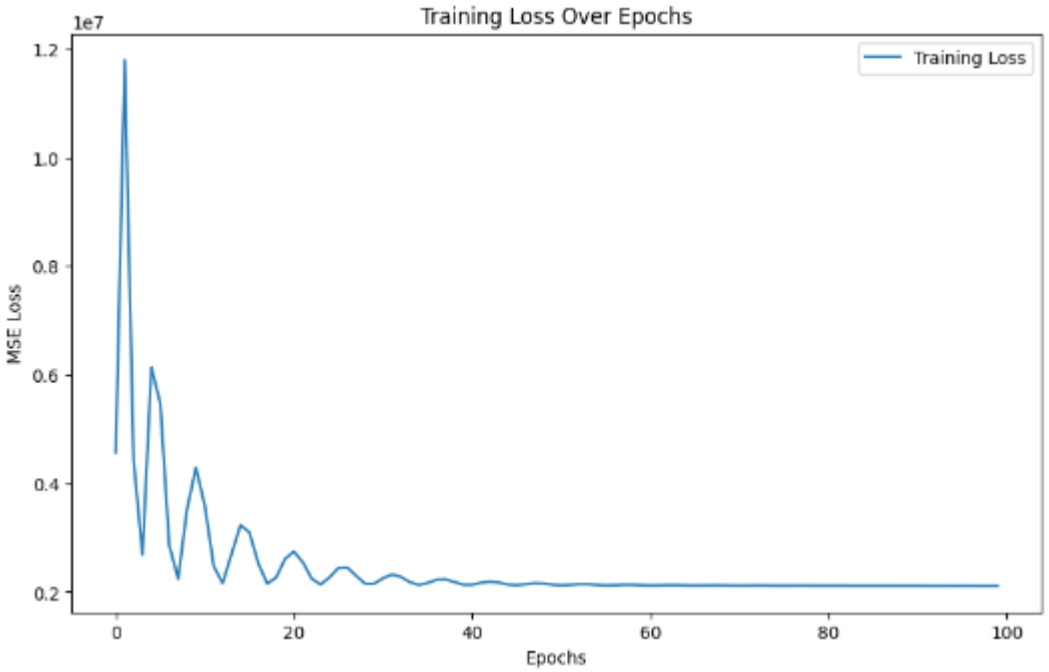

Figure 4: GNN Loss.

- **Model Stability and Consistency:** Beyond raw performance metrics, the stability and consistency of each model were evaluated by examining the standard deviation of the MSE and R^2 scores across the cross-validation folds. The Kernel Ridge model with a polynomial kernel (KernelRidge_poly) not only delivered strong performance but also maintained a relatively low standard deviation in its metrics, with an MSE standard deviation of 1.88×10^4 and an R^2 standard deviation of 0.440. This suggests that the model is reliable and consistent across different subsets of the data, making it a strong candidate for deployment in a production setting.

  On the other hand, models like the Kernel Ridge with an RBF kernel and Linear Regression exhibited high variability in their performance, with significantly large standard deviations in their R^2 scores (1.68 and 1.77×10^{12} respectively). This high variance indicates that these models are unstable and may produce inconsistent predictions, making them less suitable for robust applications.

- **Insights from Feature Importance and Model Interpretability:** Although this analysis primarily focused on the quantitative performance of different models, further exploration into the models' interpretability could yield additional insights. For instance, feature importance analysis using models like XGBoost or examining the coefficients in linear models could help identify which features are most influential in predicting the target variable. Such insights could guide future feature engineering efforts or inform domain-specific decisions based on the model's predictions.

- **Considerations for Future Work:** The results of this analysis suggest several avenues for future research and model improvement. First, optimizing hyperparameters for the best-performing models, particularly KernelRidge_poly and XGBoost, could further enhance their predictive power. Second, incorporating feature selection or dimensionality reduction techniques may help improve the performance of models like SVR and Linear Regression by reducing overfitting. Finally, exploring ensemble methods that combine predictions from multiple models could lead to more robust and accurate predictions, particularly in complex datasets like the one used in this study.

8. **Conclusion and Future Work:**

- The project successfully demonstrated the effectiveness of multiple Machine Learning models on the ‘mb7‘ dataset. The analysis highlighted the importance of using appropriate feature extraction and feature selection techniques to improve model accuracy and generalization.

- Future work could involve exploring deeper models, such as deep learning architectures, which may better capture complex patterns in the data. Additionally, employing advanced techniques such as hyperparameter tuning, more sophisticated feature engineering, and automated feature selection could further enhance model performance. These steps could help in identifying the most relevant features, reducing overfitting, and ultimately improving the robustness of the predictions.