CS771A: Assignment 3

Technocrats

May 4, 2023

Group members

Aakarsh Mittal	200002
Aditya Jain	200045
Mayank Pushpjeet	200572
Sushmita	201027
Yash Gupta	201143
Shubham Shiv Joshi	200970

Question 1

Methodology

We used the Python library pandas to prepare our data, which provides data structures for efficiently working with large datasets. We trained our linear models by selecting columns **no2op1, no2op2, o3op1, o3op2, OZONE, and NO2.** These columns contain the input features **(no2op1, no2op2, o3op1, o3op2)** and the target variables **(OZONE, NO2)** .

We then trained four different linear models using these columns. The first two models were **Ridge** and **Lasso regression**, linear models with **L2 and L1 regularization**, respectively. These models aim to prevent overfitting by adding a penalty term to the loss function, which encourages the model to have smaller weights.

The third model we used was **Linear regression with ϵ-insensitive loss and L2 regularization (SVR with the linear kernel)**. This model is based on support vector regression and aims to find a hyperplane that maximizes the margin around the training data while allowing for some errors (as controlled by the ϵ parameter).

Finally, we used **Linear regression with Huber loss and L2 regularization**, a robust regression model less sensitive to outliers than traditional least squares regression. The **Huber loss function** combines the squared loss for minor errors and the total loss for significant errors, making it more robust to outliers than just using the squared loss.

By training these different linear models, we aimed to find the model that performs best on our dataset to predict the values of **OZONE and NO2.**

Ridge Regression

We consider linear regression with least squares loss and L2 regularization, also known as ridge regression. The given set of alpha values is:

$$\texttt{alphas} = [0.001, 0.01, 0.1, 1, 10, 100, 1000]$$

After fitting the regression model with these values of alpha, we obtain the following mean absolute errors (MAE) for the target variables Ozone and NO_2:

$$\textbf{Best alpha} = 1000$$
$$\textbf{Ozone MAE} = 5.572957940382797$$
$$\textbf{NO}_2\textbf{ MAE} = 6.6882667808352565$$

However, the minimum MAE values for each target variable are achieved with alpha values of 1 and 10, respectively:

$$\textbf{alpha = 1}$$
$$\textbf{Ozone MAE} = 5.573535834682158$$
$$\textbf{NO}_2\textbf{ MAE} = 6.691136223999061$$

$$\textbf{alpha} = 10$$
$$\textbf{Ozone MAE} = 5.573523735961138$$
$$\textbf{NO}_2\textbf{ MAE} = 6.6910847594827$$

Lasso Regression

We now consider linear regression with least squares loss and L1 regularization, also known as lasso regression. The given set of alpha values is:

$$\texttt{alphas} = [0.01, 0.1, 1, 10]$$

After fitting the regression model with these values of alpha, we obtain the following mean absolute errors (MAE) for the target variables Ozone and NO_2:

$$\textbf{Best alpha value:} = 0.01$$
$$\textbf{Ozone MAE} = 5.573909433251927$$
$$\textbf{NO}_2\textbf{ MAE} = 6.685815765911227$$

However, the minimum MAE values for each target variable are achieved with an alpha value of 1:

$$\textbf{alpha} = 1$$
$$\textbf{Ozone MAE} = 5.623205264715411$$
$$\textbf{NO}_2\textbf{ MAE} = 6.77488735578542$$

SVR with the linear kernel

We have used Linear regression with ϵ -insensitive loss and L2 regularization We have a list of epsilon values and calculate the mean absolute error (MAE) for each. The given epsilon range is:

$$\epsilon = [0.01, 0.1, 1, 10]$$

After fitting the regression model with these values of epsilon, we obtain the following MAE for the target variables Ozone and NO_2:

$$\epsilon = 0.1$$
$$\textbf{Ozone MAE} = 5.558525591985404$$
$$\textbf{NO}_2\textbf{ MAE} = 6.3006275847061$$

However, the minimum MAE values for each target variable are achieved with an epsilon value of 0.01:

$$\textbf{Best } \epsilon = 0.01$$
$$\textbf{Ozone MAE} = 5.558422459331744$$
$$\textbf{NO}_2\textbf{ MAE} = 6.3003386164454955$$

Linear regression with Huber loss and L2 regularization:

We have a list of alpha values and calculate the mean absolute error (MAE) for each alpha value using linear regression with Huber loss and L2 regularization. The given alpha range is:

$$\texttt{alpha} = [0.01, 0.1, 1, 10]$$

After fitting the regression model with these values of alpha, we obtain the following MAE for the target variables Ozone and NO_2:

$$\begin{aligned} \textbf{alpha} &= 1 \\ \textbf{Ozone MAE} &= 5.600128531994584 \\ \textbf{NO}_2\ \textbf{MAE} &= 6.316732837389401 \end{aligned}$$

However, the minimum MAE values for each target variable are achieved with an alpha value of 0.1:

$$\begin{aligned} \textbf{Best alpha} &= 0.1 \\ \textbf{Ozone MAE} &= 5.559949784077467 \\ \textbf{NO}_2\ \textbf{MAE} &= 6.316646974159041 \end{aligned}$$

Conclusion

Using the linear kernel, we perform linear regression with ϵ-insensitive loss and L2 regularization. We get the minimum MAE

$$\begin{aligned} \textbf{Best}\ \epsilon &= 0.01 \\ \textbf{Ozone MAE} &= 5.558422459331744 \\ \textbf{NO}_2\ \textbf{MAE} &= 6.3003386164454955 \end{aligned}$$

Question 2

Methodology

We start by using the **DateTime module** to manipulate the timestamp data to be used as a feature in our model training. Then we use **Pandas** to prepare the data, selecting the features **humidity, temperature, no2op1, no2op2, o3op1, and o3op2** along with the manipulated timestamp.

Next, using the prepared data, we train four different **non-linear models**. The first is the **random forest model**, an ensemble learning method that constructs many decision trees at training time and outputs the average prediction of the individual trees.

The second model is the **gradient boosting regressor**, which builds an additive model in a forward stage-wise fashion, with each new tree minimizing the loss function of the whole model.

The third model is **XGBoost** for regression, an optimized distributed gradient boosting library that uses a more regularized model formalization to control overfitting.

The fourth model is the **K-Nearest Neighbors Regressor** which predicts the output based on the average of the k-nearest training samples.

Finally, we use the **DecisionTreeRegressor** with a **maximum depth** of **5** to fit a single decision tree model to the data.

Hyperparameters

We also tuned the **hyperparameters** of each model to find the optimal values for the best performance.

For **Random Forest**, we varied the number of estimators **(n_estimators)** and the maximum depth of the tree **(max_depth)** .

For **Gradient Boosting Regressor** and **XGBoost** for regression, we changed the learning rate **(learning_rate)**, the number of estimators **(n_estimators)**, the maximum depth of the tree **(max_depth)**, and the minimum number of samples required to split an internal node **(min_samples_split)**.

We varied the number of neighbors **(n_neighbors)** and the type of **distance metric (p)** for the K-Nearest Neighbors Regressor.

Finally, for **DecisionTreeRegressor,** we went to the maximum depth of the tree **(max_depth)**

Random Forest

We use the Random Forest algorithm to perform regression and obtain the following test statistics and mean absolute errors (MAE) for the target variables Ozone and NO_2:

$$\begin{aligned}\textbf{Time} &= 0.110431865200003019\\ \textbf{Ozone MAE} &= 7.371162305144488\\ \textbf{NO}_2\ \textbf{MAE} &= 5.083743621906901\end{aligned}$$

Gradient Boosting Regressor

We use the Gradient Boosting Regressor algorithm to perform regression and obtain the following test statistics and mean absolute errors (MAE) for the target variables Ozone and NO_2:

$$\begin{aligned}\textbf{Time} &= 0.09905723900001248\\ \textbf{Ozone MAE} &= 3.662584102753894\\ \textbf{NO}_2\ \textbf{MAE} &= 2.79597408492873\end{aligned}$$

Decision Tree Regressor(max depth=5)

We use the Decision Tree Regressor algorithm with a maximum depth of 5 to perform regression and obtain the following test statistics and mean absolute errors (MAE) for the target variables Ozone and NO_2:

$$\begin{aligned}\textbf{Time} &= 0.06506391259990778\\ \textbf{Ozone MAE} &= 8.628635729474723\\ \textbf{NO}_2\ \textbf{MAE} &= 5.303585858667564\end{aligned}$$

Conclusion

Using the gradient boosting regressor, we obtained the best results i.e. the minimum MAE

$$\textbf{Time} = 0.09905723900001248$$

$$\begin{aligned}\textbf{Ozone MAE} &= 3.662584102753894\\ \textbf{NO}_2\ \textbf{MAE} &= 2.79597408492873\end{aligned}$$