**Hw3: 10th April 2023**

**Name: Matrix Magician (Yunwei)**

I did two things for this homework. The one is to do some improvements based on the plain logistic model. The other thing is to learn about Ridge regression and give it a simple test.

First, I tried feature scaling. Logistic regression can be sensitive to the scale of the input features, especially when using regularization. StandardScaler or MinMaxScaler from Scikit-learn is commonly used. However, the error message indicates that the case is a sparse matrix with\_mean parameter set to True. The usual method of subtracting the mean would create a dense matrix, which could be very large and cause memory issues. So I use another scaling technique that supports sparse data, named MaxAbsScaler.

Next, I tried to tune the hyperparameters. In this example, I create a parameter grid that specifies the values of C that I want to try (In terms of the good rule of thumb are 0.001, 0.01, 0.1, 1, 10, 100). Then, I create an instance of LogisticRegression and an instance of GridSearchCV with the logistic regression model and the parameter grid as inputs. I also specify the number of cross-validation folds to use by setting the cv parameter (which is usually 5). Finally, I fit the GridSearchCV object to the scaled training data. The result shows Best parameters are {'C': 0.1, 'penalty': 'l2'}.

Then I tried a feature selection method. Logistic regression can perform better when irrelevant or redundant features are removed from the data. I tried using SelectKBest to select a subset of relevant features. The reference I chose is Chi2 and I set k=12000. The result shown on the test set becomes worse a little.

Moreover, I tried model ensembling by combining multiple logistic regression models, which are training on 2 equal-sized subsets of the data to improve the overall performance of the model. Specifically, I train 2 logit-reg models with the same hyperparameters and make predictions with each model. The final prediction is the average of the individual model predictions. Model ensembling can help reduce the variance of the predictions by averaging the predictions of multiple models. This can help prevent overfitting and improve the generalization performance of the model. Ensembling can also help reduce bias by combining models with different biases. In addition, it can also increase the robustness and stability of predictions. In other words, it can reduce the impact of errors or weaknesses in any individual model. The result shown on the test set did not improve too much.

Finally, I tried ridge regression without tuning parameters. Basically, ridge regression includes a penalty term in the loss function to address multicollinearity, which occurs when the independent variables in a regression model are highly correlated with each other. There is a classifier using Ridge regression in Scikitlearn and text data usually has the problem of multicollinearity. The result shown on the test becomes worse a lot.