AMATH 352: HOMEWORK 7

JOLIE TRAN

ptran32@uw.edu

1. INTRODUCTION AND OVERVIEW OF THE PROBLEM

This report classifies the political parties of the politicians based on their 1984 congressional voting records by training a linear classification model and investigates the quality of that model with the choice of different numbers of features using mean squared error.

2. THEORETICAL BACKGROUND AND DESCRIPTION OF ALGORITHM

2.1. **Linear classification model:** A linear classifier achieves this by making a classification decision based on the value of a linear combination of the feature vectors $\underline{x}_j's$ [1]. We need to find an affine function $f: \mathbb{R}^n \to \{-1, +1\}$ so that $f(\underline{x}_j) = \underline{y}_j$. We can write $f(\underline{x}_j) = \alpha_0 + \alpha_1 x_{j1} + \alpha_2 x_{j2} + \cdots + \alpha_N x_{jN}$, where $\underline{\alpha} \in \mathbb{R}^{N+1}$, $\underline{x}_j \in \mathbb{R}^N$, $j \in \{1, 2, \dots, M\}$. This function then can be rewritten as system of equation:

$$\begin{bmatrix} \alpha_0 & \alpha_1 x_{11} & \alpha_2 x_{12} & \cdots & \alpha_N x_{1N} \\ \alpha_0 & \alpha_1 x_{21} & \alpha_2 x_{22} & \cdots & \alpha_N x_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_0 & \alpha_1 x_{M1} & \alpha_2 x_{M2} & \cdots & \alpha_N x_{MN} \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1N} \\ 1 & x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix}$$

In this report, we use our training dataset to train the model and use the threshold function $y_{predict}(\underline{x}) = \text{sign}(f(\underline{x}))$ to map the linear function $f(x)$ to the values $+1$ and -1 (note that y_j's have the value $\{-1, +1\}$ while $f(\underline{x})$ can give us any values).

2.2. **Mean squared error (MSE):** To validate or test the quality of our model, we can compute the error between the predicted and actual values. One such method is to use MSE, where we can compute the mean of squared error. In this report, we want to check the error between $\underline{y}_{predicted}$ of our linear classification model with the computed coefficients $\underline{\alpha}$.

$$MSE(\underline{\alpha}) = \frac{1}{k} \sum_{j=1}^{k} \left\| \text{sign}\left(f(\underline{x}_i)\right) - \underline{\text{y}}_i \right\|_2^2$$

3. COMPUTATIONAL RESULTS

Given a dataset which contains 4 arrays $X_{train} \in \mathbb{R}^{300\times16}$, $Y_{train} \in \mathbb{R}^{300}$, $X_{test} \in \mathbb{R}^{135\times16}$, $Y_{test} \in \mathbb{R}^{135}$, where $x_{train_j}, x_{test_j} = \{-1, 0, +1\}$ and $y_{train_j}, y_{test_j} = \{-1, +1\}$. We can construct a linear classification model with different values of features $N = 16, N = 2,\ N = 3$, and $N = 4$. After training the linear function $f(\underline{x}) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_N x_N$ and taking $y_{predict}(\underline{x}) = \text{sign}(f(\underline{x}))$ as our threshold function, we can validate the model by computing the error between the predicted values and the actual values of both the training and test set. In this report, we use the MSE method.

Looking at Table 1, we see that the more features that we consider into training our classifier model, the more accurate the model is. However, one thing to note from this table is that we don't need all 16 features for our classifier model. In fact, we only need 4 features (or the first 4 columns) from our training dataset X_{train} and we can get the same level of accuracy as when we have all 16 features in our model.

One more thing that we notice from Table 1 is that this linear classification model works well on the test dataset, and even better than on the training dataset.

	$N = 16$	$N = 2$	$N = 3$	$N = 4$
MSE of the training data	0.133	1.24	0.373	0.133
MSE of the test data	0.120	0.573	0.360	0.120

Table 1: Mean Squared Error of the Training and Test Data on Different Number of Features

4. SUMMARY AND CONCLUSIONS

We can construct a linear classification model on the 1984 congressional voting records and see that this classifier works well for our dataset. This model works even better with the test data as we get the smaller mean squared error between the actual and the predicted values compared to the training data. Additionally, this model, as we expected, performs poorly on a small number of features $N = 2$, and to our surprise, performs as its best capacity with the feature value $N = 4$. It is important to choose the right number of features so that we train the linear classification model to perform effectively (small error) and efficiently (cost the least). Overall, we are satisfy with the small error bound when we choose the right N so we can say this is a good classification model for our dataset.

REFERENCES

[1] Yuan, G.X., C.H. Ho, C.J. Lin, 2012: Recent Advances of Large-Scale Linear Classification, *IEEE*, **100**, 9, http://dmkd.cs.vt.edu/TUTORIAL/Bigdata/Papers/IEEE12.pdf