Machine Learning Project 1

Ren Li, Yifei Song, Hao Zhao
Department of Computer Science, EPFL

I. Data Preprocessing

We use the variables prefixed with DER and PRI as the input data, and exclude the variables which have meaningless values (i.e., -999.0). This gives us the input feature $x \in R^D$, where $D = 8 + 11$. For cross validation, we split the data to a training set and a validation set, whose sample ratio is 9:1. Consequently, the validation is taken in a 10-fold way. The raw data is normalized to have zero-mean and unit-variance (the test data is normalized by the same mean and variance computed from the training data). In our experiments, we also consider the polynomial, the power-series and the sine augmentation for the input feature. Given an input $x = [1, x_1, x_2, ..., x_D]$, the power-series augmentation for degree M is formed by:

$$\phi^M(x) = [1, x_1, ..., x_D, x_1^2, ..., x_D^2, ..., x_1^M, ..., x_D^M], \quad (1)$$

which can be considered as the polynomial augmentation without interactive items (e.g., $x_i x_j$, $x_i^2 x_j x_k$). For the sine augmentation, in addition to $sin_{x_i}, cos_{x_i}, sin_{2x_i}, cos_{2x_i}$, we also consider to incorporate the following items:

$$sin_{x_i} sin_{x_j}, sin_{x_i} cos_{x_j}, cos_{x_i} sin_{x_j}, cos_{x_i} cos_{x_j}, \quad (2)$$

where sin_{x_i} and cos_{x_i} denote the sine and cosine value of x_i. In the following sections, we will use $y \in R^{N\times 1}$, $X \in R^{N\times D}$, w, N and σ to represent the ground truth, the input, the model parameters, the number of samples, and the sigmoid function respectively.

II. Method

A. Least Square - (Stochastic) Gradient Descent

The loss function of the least square is $L(w) = \frac{1}{2N}(y - Xw)^T(y - Xw)$. We first employ the gradient descent (GD) and the stochastic gradient descent (SGD) method to find the optimal parameter w. The update rule of w is

$$w^{t+1} := w^t - \gamma g, \quad (3)$$

where g is $-\frac{1}{N}X^T(y - Xw)$ and $-x_n(y_n - x_n^T w)$ for GD and SGD, respectively. γ denotes the step size.

In order to enhance the prediction performance, we simply utilize the power-series augmentation. The model performance is evaluated on the validation sets of cross validation. Table I shows the validation accuracy of GD and SGD with different augmentation degrees after 150 iterations (note for SGD, one iteration means N steps of Eq. 3).

M	1	5	10
GD	0.733	0.752	0.620
SGD	0.683	0.677	0.665

Table I
The validation results with different degrees of augmentation.

We can observe that the best degree M for GD and SGD is 5 and 1, which means only a proper augmentation degree can be helpful to avoid overfitting and underfitting. The best result of GD (0.752) outperforms that of SGD (0.683) by about 0.07 in accuracy. Since SGD uses a single sample to compute the gradient, which can be noisy and random for the weight update, its performance is consequently inferior to GD, and may not be able to find the global minimum.

B. Least Square - Normal Equations

Next, we implement the least square method by solving normal equations $X^T X w = X^T y$. Following Section II-A, we use the power-series augmentation (this may cause $X^T X$ non-invertible, so we apply pseudo inverse for these cases). Table II summarizes the results on the validation sets. One can note that while increasing the augmentation degree increases the accuracy, too large degree actually will not help for the improvement (see $M = 20$ vs $M = 30$). Finally, we evaluate the performance on the test set for $M = 20$, and get 0.789 and 0.676 for accuracy and F1 score respectively, which is comparable to the result of ridge regression (0.788/0.674, the last column of Table III).

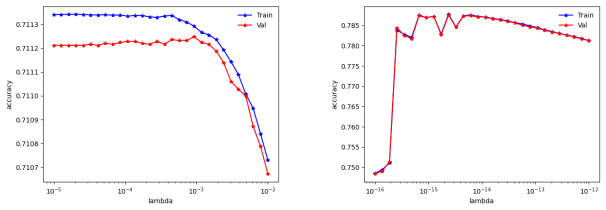
M	1	5	10	20	30
Accuracy	0.711	0.756	0.774	0.788	0.788

Table II
The validation results with different degrees of augmentation.

C. Ridge Regression

In ridge regression, the model parameters are computed by $w = (X^T X + 2N\lambda I)^{-1} X^T y$, where λ is the weight for the regularization loss. In order to determine the value of λ, we run cross validation and calculate the validation accuracy for each λ in $[\lambda_{min}, \lambda_{max}]$. We choose the λ^* with the highest accuracy as the best one, and use it to give the final parameter w^*. Fig. 1(a) shows the results under different λ values. We also consider to augment the given features as previous sections to further improve the prediction results. Fig. 1(b) shows the results when degree $M = 15$, and we

can notice the improvement in the best validation accuracy is over 0.07 when compared to $M = 1$. It is also indicated in Fig. 1 that a too large or too small value for λ will harm the model performance. Table III reports the accuracy and F1 score on the test set. As expected, increasing M will lead to the higher accuracy and F1 score on the test set. **We obtain the best test result, 0.813/0.717 for the accuracy/F1 score, by ridge regression with the mixture of different augmentation strategies (Poly3+Sine+Power, elaborated in Section II-D).**



(a) $M = 1$ (b) $M = 15$

Figure 1. The validation results with different regularization weight.

M	1	3	7	10	15
Accuracy	0.712	0.738	0.767	0.775	**0.788**
F1 score	0.512	0.583	0.636	0.650	**0.674**

Table III
THE TEST RESULTS WITH DIFFERENT DEGREES OF AUGMENTATION.

D. (Regularized) Logistic Regression

Finally, We apply the logistic regression method which uses Eq. 4 to represent the probability of predefined categories for the binary classification problem.

$$p(y_n = 1|x_n, w) = \sigma(x_n^T w),\ p(y_n = 0|x_n, w) = 1-\sigma(x_n^T w), \tag{4}$$

where $y_n = 1$ denotes the Higgs boson class, and $y_n = 0$ denotes the background type. This approach combines the sigmoid function that can take any real-valued number predicted by the linear function and map it into an S-shaped curve between 0 and 1, which gives an intuitive interpretability to this classification method.

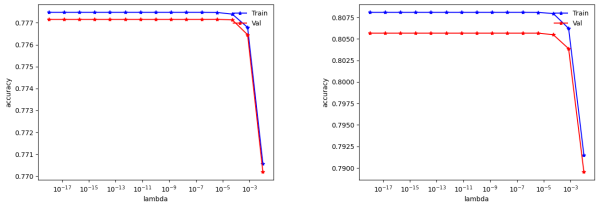
The loss function of the logistic regression can be regarded as the cross entropy, which is given by:

$$L(w) = -\frac{1}{N}\sum_{n=1}^{N} y_n ln[\sigma(x_n^T w)] + (1 - y_n) ln[1 - \sigma(x_n^T w)]. \tag{5}$$

We can derive the gradient expression of Eq. 5 as $\nabla L(w) = \frac{1}{N} X^T[\sigma(Xw) - y]$. For the regularized logistic regression, since its loss function is $L_R(w) = L(w) + \lambda||w||^2$ (λ is the weight for the regularization item), its gradient is expressed as $\nabla L_R(w) = \nabla L(w) + 2\lambda w$ accordingly. Gradient descent algorithm is used to find the global optimal weights for both the regularized and the un-regularized logistic regression. Following Section II-C, the value of λ is determined by the results of cross validation.

Here we consider the polynomial/sine/power-series augmentation, because we want to introduce more nonlinear components with different properties into the feature to boost the model performance. Table IV(a) shows the test results of the logistic regression with different combination of feature augmentation, where Poly2 and Poly3 represent the polynomial augmentation with degree 2 and 3. Because Poly3 introduces richer interactive items, it performs better in test set with 0.794/0.654 for the accuracy/F1 score than Poly2 with 0.777/0.654. The model's classification ability benefits from the mixed augmentation strategies, which can be proven by the last 3 columns in Table IV(a). Combining Sine augmentation method with Poly3, the accuracy metric rises from 0.794 to 0.799, and F1 score also improves nearly 0.01. When power-series augmentation is included into the strategy, the accuracy and F1 score are higher, increased to 0.809/0.711. These inter-group comparison results show that richer feature expression is helpful to enhance the model performance.

The results of the regularized logistic regression in Table IV(b) are almost the same as those in Table IV(a). Due to the limited computation resource, we have to stop GD after a specific number of iterations while the training loss is still reducing. So this gives us the flat curves for $\lambda < 10^{-5}$ in Fig. 2 where we show the validation results under different values of λ for the augmentation of Poly2 and Poly3+Sine+Power. If given enough iterations, we believe the validation accuracy will go down along with the decrease of λ's value.



(a) Poly2 (b) Poly3+Sine+Power

Figure 2. The validation results with different regularization weight.

(a) Logistic Regression

Aug. Type	Poly2	Poly3	Poly3+Sine	Poly3+Sine+Power
Accuracy	0.777	0.794	0.799	0.809
F1 score	0.654	0.685	0.694	0.711

(b) Regularized Logistic Regression

Aug. Type	Poly2	Poly3	Poly3+Sine	Poly3+Sine+Power
Accuracy	0.777	0.794	0.798	0.809
F1 score	0.654	0.685	0.692	0.711

Table IV
THE TEST RESULTS WITH DIFFERENT AUGMENTATION STRATEGIES.