Assignment 3 Option 2
Quantity To Produce Quality

A.D. van der Merwe
Department of Computer Science
University of Stellenbosch
24923273
24923273@sun.ac.za

***Abstract*—**

I. Introduction

II. Background

This section presents background information on the gradient descent optimisation algorithm, logistic regression model, and ensemble learning. Additionally, background information on bootstrap aggregating, basis functions, and performance metrics used in this report.

A. Gradient Descent

The gradient descent algorithm was first introduced by Augustin-Louis Cauchy in 1847 [3]. Cauchy introduced gradient descent to solve optimisation systems of simultaneous equations through iterative optimisation to find the minimum of a function. Cauchy also introduced the step size parameter, now commonly referred to as the learning rate, to control how large the steps are for each iteration as the algorithm updates model parameters to reach an optimal solution.

The generic learning algorithm of gradient descent is represented by Algorithm 1.

Algorithm 1 Gradient Descent Learning Algorithm

1: Preprocess the training set D_T as necessary
2: Initialise parameter vector, $\mathbf{w}(t)$, $t = 0$
3: Initialise the learning rate η
4: **while** stopping condition not satisfied **do**
5: **for** each $i = 1, ..., n_T$ **do**
6: Calculate error signal, $\delta(t)$
7: Calculate a search direction, $\mathbf{q}(t) = f\left(\mathbf{w}(t), \delta(t)\right)$
8: Update parameter vector: $\mathbf{w}(t+1) = \mathbf{w}(t) + \eta\mathbf{q}(t)$
9: **end for**
10: $t = t + 1$
11: Compute prediction error
12: **end while**
13: Return $\mathbf{w}(t-1)$ as solution

B. Logistic Regression

The logistic regression model was first introduced by David Cox in 1958 as a method to perform binary classification [4]. Cox specifically designed the logistic regression model to model the probability of a binary outcome as a function of descriptive features.

To construct a logistic regression model that makes use of gradient descent as an optimisation algorithm, a threshold function that is continuous, and therefore differentiable is needed. This function is known as the logistic function and is represented by the mathematical equation below.

$$Logistic(z) = \frac{1}{1 + e^{-z}} \tag{1}$$

where z is a numeric value.

Before the logistic regression model is constructed the binary target features are mapped to 0 or 1. The logistic regression model is then constructed by use of the equation that follows.

$$\mathbb{M}_{\mathbf{w}}(\mathbf{d}_i) = \frac{1}{1 + e^{-\mathbf{w}\cdot\mathbf{d}_i}} \tag{2}$$

where $\mathbf{d}_i$ is a vector of the i-th descriptive features, with the bias term represented by $\mathbf{d}_0$ and equal to one, $\mathbf{w}$ is a vector of weights, where $\mathbf{w}_0$ represents the weight of the bias term, and the weights that remain corresponds to their respective descriptive features in $\mathbf{d}_i$. The term $\mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)$ represents the predicted output for the i-th instance of the logistic regression model. The output of the logistic regression model can be interpreted as probabilities of the occurrence of a target instance that belongs to a specific class. The probability the i-th target instance that belongs to class one is given by the equation below.

$$P(y_i = 1|\mathbf{d}_i) = \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i) \tag{3}$$

where y_i is the true label for the i-th observation. Similarly, the probability of the i-th target instance that belongs to class zero is given by the equation below.

$$P(y_i = 0|\mathbf{d}_i) = 1 - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i) \tag{4}$$

To classify the i-th target instance, $\mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)$ is compared to a threshold of 0.5. The equation used to classify the i-th target feature that belongs to either class zero or class one is given as follows.

$$\hat{y}_i = \begin{cases} 0 & \text{if } \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i) < 0.5 \\ 1 & \text{if } \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i) \geq 0.5 \end{cases} \tag{5}$$

where $\hat{y}_i$ is the predicted class of the i-th binary target variable.

Gradient descent is used as the optimisation algorithm to find the optimal decision boundary for a logistic regression model. The optimal decision boundary is defined as the set of weights that minimise the sum of squared error (SSE) based on the training set. The mathematical represntation of the SSE is as follows.

$$L_2\left(\mathbb{M}_{\mathbf{w}}, \mathcal{D}\right) = \frac{1}{2}\sum_{i=1}^{n}\left(y_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)\right)^2 \tag{6}$$

where $\mathcal{D}$ is the training dataset and n is the number of instances in the training dataset and L_2 is the SSE of the training dataset.

The equation used to represent the error signal used in the gradient descent optimisation algorithm to update the weights of the logistic regression model is as follows.

$$\delta(\mathcal{D}, w_j) = \sum_{i=1}^{n}\left((y_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))\mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)(1 - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))d_{j,i}\right) \tag{7}$$

where w_j is the j-th weight of the logistic regression model.

The equation used to update the weights of the logistic regression model by use of the gradient descent optimisation algorithm is as follows.

$$w_j = w_j + \eta\sum_{i=1}^{n}\left((y_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))\mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)(1 - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))d_{j,i}\right) \tag{8}$$

where η is the learning rate.

The logistic regression model is quite robust to noise and outliers in the dataset. However, the logistic regression can not handle missing values and sensitive to imbalanced classes. Additionally, the logistic regression model requires categorical features to be encoded into numerical representation by either the ordinal encoded or the one hot encoded technique and the data needs to be scaled or normalised. The logistic regression also assumes a linear relationship between the descriptive features, where the actual relationship between descriptive features might be non-linear.

C. Basis Functions

Basis functions are non-linear elements which transforms the linear inputs to the logistic regression into non-linear represntations, while the model itself remains linear in terms of the weights [5]. The addition of basis functions allows logistic regression model to capture relationships between descriptive features which are non-linear.

The data is transformed by use of a series of basis functions, which enables the logistic regression model to effectively manage non-linear relationships between descriptive features. A logistic regression model that makes use of basis functions is represented by the equation below.

$$\mathbb{M}_{\mathbf{w}}(\mathbf{d}_i) = \frac{1}{1 + e^{-\sum_{j=0}^{b} w_j\phi_j(\mathbf{d}_i)}} \tag{9}$$

where ϕ_0 to ϕ_b are a series of b basis functions that each transform the i-th input vector $\mathbf{d}_i$ in a different way. Usually b is larger than n, which means that there are more basis functions than there are descriptive features.

There are several disadvantages when basis functions are used in a logistic regression model to capture non-linear relationships between descriptive features. Firstly, some prior knowledge of these non-linear relationships is required to select appropriate basis functions. Secondly, an increased number of basis functions results in larger gradient descent search spaces, which can lead to longer convergence times and complicate the optimisation process.

D. Ensemble Learning

Ensemble learning combines several individual models to to obtain better generalisation performance and predict a new instance based on multiple models opposed to a single model [7].

Each model in an ensemble is trained on the same dataset and yields slightly different results due to variations in training data, model configurations or model architectures. Each model performs differently on certain data patterns. By aggregating these diverse models, the ensemble will balance individual errors and achieve better overall performance than any single model alone. This diversity helps the ensemble to cover the limitations of each model, which results in more accurate and robust predictions. An ensemble also helps to decrease the variance in the model predictions.

Approaches used to create these diverse models are

- Train the the same type of model on different subsets of the observation of the training data.
- Train the same type of model which uses different features of the training data.
- Use different types of models, that results in heterogeneous ensembles and cancels out the inductive bias of each model.
- Use different training or optimisation algorithms.
- Use different control parameters
- Use different model architectures.

An ensemble that contains only one type of model is called a homogeneous ensemble.

E. Bootstrap Aggregating

Bootstrap aggregating, also know as bagging, was first introduced by Leo Breiman, in 1996, as a method used to generate a diverse set of models to produce an aggregated model [2]. This diverse set of models is formed by training different models on a subset of the original data.

F. Performance Metrics

Performance metrics are essential tools when the effectiveness of classification models are evaluated. Performance metrics provide a quantitative measure of how reliable and accurate a prediction model performs classification on a dataset. Key metrics include accuracy, precision, recall, and F1-score, each offering unique insights into different aspects of model performance [1].

a) Accuracy: Accuracy is a common method used to evaluate the performance of classification models. The accuracy of a predictive classification model is determined by the proportion of correctly predicted labels against the total number of predictions. The calculation of the accuracy of a predictive model is as follows:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{10}$$

Accuracy is a popular choice of performance measure mainly beacause it is fairly easy to understand and compute. Accuracy generally perform well on well balanced datasets. On imbalanced datasets, accuracy can produce values that are misleading.

b) Precision and Recall: Precision is the proportion of true positive (TP) predictions against all of the TP and false positive (FP). The equation to calculate the precision of a classification model is as follows:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{11}$$

Recall is the proportion of TP predictions against all of the TP and false negative (FN). The equation to calculate the recall of a classification model is as follows:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{12}$$

c) F1-score: When a binary classification dataset has imbalanced classes, the accuracy of a model can present a high score that does not represent good performance, as the majority group could be overclassified. Therefore, when an imbalanced binary classification dataset is used, it is better to use multiple performance metrics.

The binary F1-score, also known as the Dice similarity coefficient, is the harmonic mean of precision and recall, that provides a balance between the precision and recall [6]. The equation used to calculate the binary F1-score is as follows.

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{13}$$

The binary F1-score proves especially useful when model performance is assessed on imbalanced binary classification datasets.

III. Implementation

IV. Empirical Procedure

A. Performance Metrics

B. Data Preprocessing

C. Experimental Setup

D. Control Parameters

E. Statistical Significance and Analysis

V. Research Results

VI. Conclusion

TABLE I: Linear Logistic Regression Control Parameters

Dataset	Control Parameters		
	eta	*epochs*	*patience*
Breast Cancer	0.00239	10678	6
Diabetes Dataset	0.01121	18690	9
Banana Quality	0.00023	4338	5
Water Quality	0.00601	25484	9
Spiral Dataset	0.06708	17335	7

TABLE II: Non-Linear Logistic Regression Control Parameters

Dataset	Control Parameters			
	eta	*epochs*	*patience*	*% poly*
Breast Cancer	0.00011	7290	10	40
Diabetes Dataset				
Banana Quality				
Water Quality				
Spiral Dataset				

References

[1] J. Braet, M. Cristina, Hinojosa-Lee, and J. Springael. "Evaluating performance metrics in emotion lexicon distillation: a focus on F1 scores." (2024).

[2] L. Breiman "Bagging predictors." In: Machine learning (1996).

[3] A. Cauchy "Méthode générale pour la résolution des systemes d'équations simultanées." In: Comp. Rend. Sci. Paris (1847).

[4] D. R. Cox "The regression analysis of binary sequences." In: Journal of the Royal Statistical Society Series B: Statistical Methodology (1958).

[5] J. D. Kelleher, B. Mac Namee, and A. D'arcy "Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies." In: MIT press (2020).

[6] B. J. Erickson, and K. Felipe "Magician's corner: 9. Performance metrics for machine learning models." In: Radiology: Artificial Intelligence 3(2021).

[7] M. A. Ganaie, M., Hu, A. K. Malik, M. Tanveer and P. N. Suganthan "Ensemble deep learning: A review." In: Engineering Applications of Artificial Intelligence (2022).