Advanced Machine Learning - Optimizers

Wiktor Jakubowski, Zuzanna Kotlińska, Jan Kruszewski

March 2024

Contents

1 Introduction

This project aimed to implement different optimization algorithms and compare their performance on several classification tasks.

1.1 Optimizers

Three optimization algorithms were implemented and tested throughout this project:

- Iterative Reweighted Least Squares (IRLS),
- Stochastic Gradient Descent (SGD),
- Adaptive Moment Estimation (ADAM).

1.2 Data

We have chosen nine different datasets for experiments. All of them are available at OpenML website. These are:

- Small datasets (≤ 10 features):
  - diabetes (37)
  - banknote-authentication (1462)
  - pollen (871)
- Large datasets (> 10 features):
  - puma32H (752)
  - MagicTelescope (1120)
  - vehicle (994)
  - pc3 (1050)
  - waveform-5000 (979)
  - ozone-level-8hr (1487)

The datasets were preprocessed by removing highly correlated features ($>$ 0.8 correlation coefficient) and imputing the missing values using the mean value of the columns in the training dataset. All the datasets had numeric values only so there was no need to perform encoding.

2 Methodology

Each optimizer was used to train logistic regression model. In the training phase, we utilized early stopping rule to stop the model from decaying by detecting deteriorating performance on the training set. In our case, the model's training is stopped when there are five consecutive epochs with worsening loss.

To evaluate the performance of classifiers, we resorted to the Balanced Accuracy metric, defined as:

$$\frac{1}{2}(\frac{TP}{P} + \frac{TN}{N}) \tag{1}$$

In our experiments, optimized logistic regression model by each of the three optimizers. We trained and evaluated them on the aforementioned datasets by conducting cross-validation with 5-folds and setting the same seed. Figures 1, 2 and 3 on the next page portray the results of conducted experiments for SGD, ADAM and IRLS optimizers respectively. In each plot, we have a boxplot depciting balanced accuracy for every one of 10 datasets.

It can be observed that, all in all, Adaptive Moment Estimation optimizer yielded the most stable results across the datasets. Not only was his accuracy on par with the best across three optimizers in nearly every dataset, it was also quite comparable in each fold during cross-validation.

Iterative Reweighted Least Squares optimizer proved to be stable and effective. Of particular note is its result on set 5, where the average accuracy value was more than 20% better than the other two optimizers.

When it comes to Stochastic Gradient Descent, the most varied results can be observed for the different folds of cross-validation. On the other hand results-wise the optimizer seems to be pretty similar to Adam optimizer. Though the algorithm got outperformed by other two on dataset no. 6.

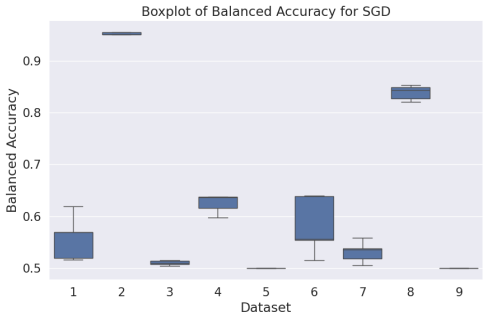

Figure 1: Balanced accuracy score per dataset for SGD optimizer.

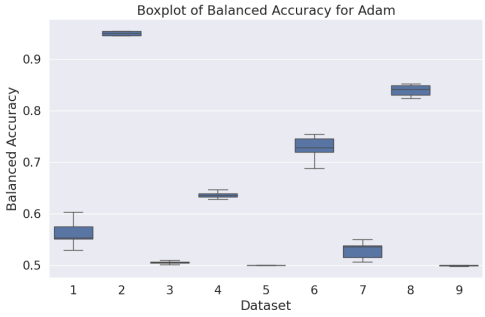

Figure 2: Balanced accuracy score per dataset for Adam optimizer.

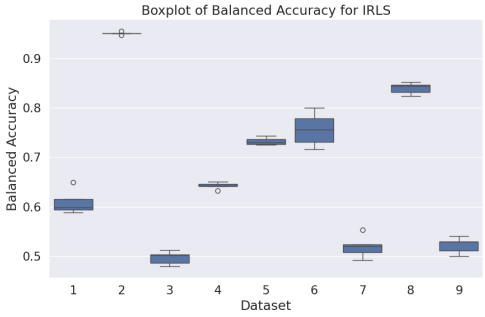

Figure 3: Balanced accuracy score per dataset for IRLS optimizer.

3 Convergence analysis

When inspecting the performance of the logistic regression model, not only is it crucial to assess its accuracy, but also it is vital to see how the loss function reacted to changes in weight values. In the case of the logistic regression model, the loss function is defined as the logarithm of the likelihood function, given as:

$$l(\beta) = \sum_{i=1}^{n} y_i \log\left(p\left(x_i\right)\right) + \left(1 - y_i\right) \log\left(1 - p\left(x_i\right)\right) \tag{2}$$

where $p(x_i)$ is the probability of classifying the input as class 1, and y_i is denoted as the true value of class 1, for i-th observation.

For the algorithm to converge, the loss has to decrease over time, i.e. it does not fluctuate. Figures on the next page show the loss function over time at selected datasets.

In figure 4, we can see that the SGD algorithm converged in almost every instance. There was one case, namely when training on dataset 9, where the loss function fluctuated and hasn't converge.

Figure 5 clearly demonstrates the supremacy of ADAM optimizer in terms of convergence. It converged in every experiment, making it reliable optimizer. However, it is worth noting that for some datasets it takes the most epochs for this optimizer to converge.

When it comes to IRLS, the behaviour of the algorithm is very particular. For each dataset, the algorithm converges after only a few epochs. For the next few hundred epochs, the efficiency of the model does not increase. The behaviour of the algorithm is shown in figure 6.

Figure 4: Log-likelihood per epochs for SGD optimizer.

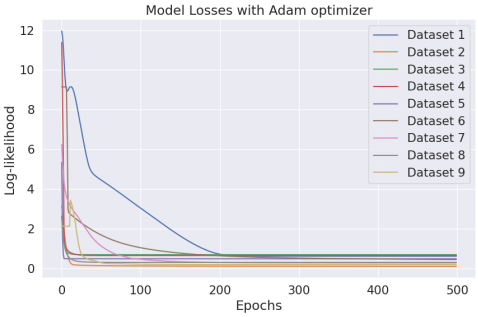

Figure 5: Log-likelihood per epochs for Adam optimizer.

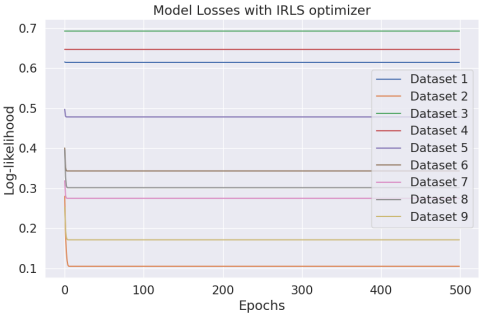

Figure 6: Log-likelihood per epochs for IRLS optimizer.

4 Comparison of classification performance

To create baselines for our models, we compared the performance by training and evaluating several popular statistical and machine learning models, such as:

- Linear Discriminant Analysis
- Quadratic Discriminant Analysis
- Decision Tree
- Random Forest

Figures 7, 8, 9 and 10 show the results on datasets 2, 4, 6, and 8, respectively.

Admittedly, all the baseline models outperformed every optimizer on dataset 2 and Dataset 6. In the former, the differences are significantly smaller than in the latter instance. In the experiments on dataset 4, again the baseline models were better, but only tree-based. IRLS and ADAM performed comparably to the LDA and QDA baselines. The most promising results for optimizers are related to dataset 8. We can see that all of them achieved comparable results to the baseline models. This goes to show that the optimizers are implemented correctly, and their performance can be on the same level as the other prominent classifiers. The reason for the lower accuracy probably lies in the nature of logistic regression itself, not in the limitations of the optimizers.

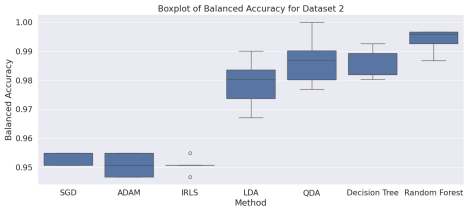

Figure 7: Comparison of classification performance for Dataset 2.

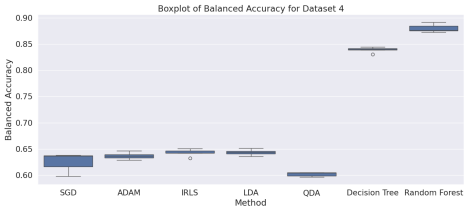

Figure 8: Comparison of classification performance for Dataset 4.

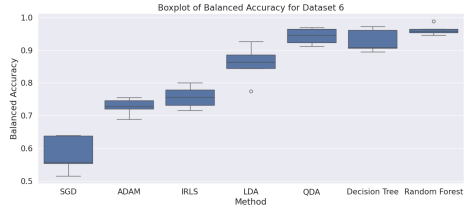

Figure 9: Comparison of classification performance for Dataset 6.

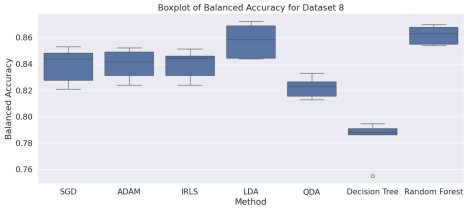

Figure 10: Comparison of classification performance for Dataset 8.

5 Comparison of models with and without interactions

In the case of smaller datasets, where there are not many features present, we should think about feature engineering. For this reason, we conducted the analysis, how artificial features affect the model's performance. We created them by making the combination of two original features, i.e.:

$$\hat{X} = X_i \times X_j \qquad \forall i, j \in [n] \tag{3}$$

where n is the set of features.

Figures 11, 12, 13 show the performance of the models with and without the interactions on three datasets (1, 2, 3).

We can conclude that there is no general answer as to whether or not the created features improve the model's performance. In some cases (such as on dataset 2), we have observed significant improvements), whereas in others - a decline.

Thus, it comes down to the fundamental truth, that it all depends on the data.

Figure 11: Methods performance with and without interactions for Dataset 1.

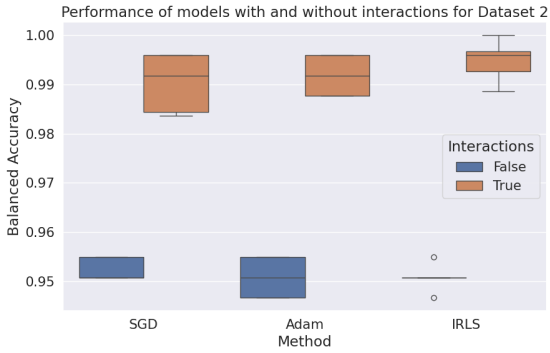

Figure 12: Methods performance with and without interactions for Dataset 2.

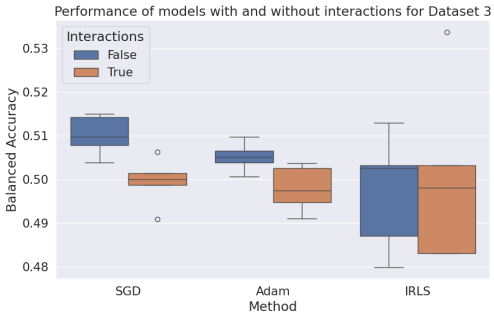

Figure 13: Methods performance with and without interactions for Dataset 3.

6 Conclusion

During the project we conducted a comprehensive analysis of three optimization algorithms for logistic regression models across various datasets. Diverse experiments helped us to understand better behaviour and their characteristics. Adam and IRLS proved stable performance on different CV folds while SGD appeared to perform at the similar level of effectiveness to Adam. None of the optimizers had a bigger problem with convergence and adding features interaction to the dataset has proven to have different effects on the predictive quality of the model.