# Advanced Machine Learning, Project 1 - Report

Julia Kaznowska, Jan Smoleń, Filip Szympliński

April 2024

# 1 Methodology

## 1.1 Datasets

In this project the following datasets were used:

- Small datasets: Magic gamma telescope, Palmer penguins and Banana
- Large datasets: Ionosphere, Airline passenger satisfaction, Connectionist bench sonar mines vs rocks, PCOS, Breast cancer Wisconsin diagnostic and Covertype.

The prepossessing of datasets contained:

- Removing highly correlated variables (Pearson correlation coefficient greater than $0.8$)
- Dropping null values (due to the fact that they constituted to less than $5\%$ of all the data)
- Encoding categorical variables into binary using one-hot encoding
- Encoding categorical variables using ordinal encoding when natural hierarchy was present
- Scaling the variables (using StandardScaler from sklearn)

In case of the penguins dataset species *Adelie* and *Chinstrap* were joined into one because there was not much difference between them. In the cover type dataset multiple cover types were joined into two new classes to make this dataset suitable for a binary classification problem.

## 1.2 Stopping rules

In this project, two methods of stopping were implemented. Their purpose is to stop training process when it's no longer needed. Two criteria were used: no improvement in loss on validation dataset and convergence analysis.

### 1.2.1 Early stopping

This method splits the training data into two subsets to evaluate the model. First - `train` - is used strictly for the training process. The second one, called `val`, which is $20\%$ of the original training data, is used for validation of the model. Its purpose is to prevent over-fitting. The idea is that if during `n_iter_no_change` iterations (default $5$) there is no improvement over the best loss on `val` data which is grater than `epsilon_change` (default $10^{-2}$), the training is stopped. After that, the best model (in terms of loss on `val` data) is returned.

### 1.2.2 Convergence detection

This method of stopping the training process does not split the training data. The idea standing behind this is as follows: the value of the loss on the training data is constantly monitored. If there is a convergence, the training is stopped, otherwise it's not. The exception occurs when during 500 iteration there is no convergence. In that situation, the training is aborted and the model used in the last iteration is returned. The convergence in this implementation is the situation when the difference between the smallest and the largest training loss within the last 10 iterations is smaller than `epsilon_change` (default $10^{-2}$). Otherwise, a train loss does not converge. Naturally, this definition can be applied when there are at least 10 iterations of the training loop.

## 2 Convergence analysis

Convergance analysis was a task, where we had to check how the value of the log-likelihood function depends on the number of iterations for all the optimizers.

For this task we prepared the datasets as the samples of size 10,000 drawn from original datasets (if the dataset had more observations than that). `SGD` and `IWLS` had the `learning rate` of 1e-3. `Adam` had the following parameters, checked upon in literature: `learning rate` of 1e-2, `beta 1` of 0.9 and `beta 2` of 0.999.

It turns out that most of the algorithms prepared by us did not converge (didn't end before 500 iterations). The only ones that did are presented in table 1:

| | SGD | IWLS | Adam |
|---|---|---|---|
| banana | 500 | **15** | 500 |
| magic_telescope | 500 | **14** | **353** |
| penguins | 500 | 500 | 500 |
| sonar | 500 | 500 | 500 |
| ionosphere | 500 | 500 | **170** |
| breast_cancer | 500 | 500 | **354** |
| airline | 500 | **14** | 500 |
| covertype | 500 | 500 | 500 |
| pcos | 500 | 500 | 500 |

Table 1: Number of iterations before finishing the algorithm

For `SGD` the algorithms have never converged. For both `Adam` and `IWLS` the argorithms converged 3 times. `IWLS` had much lower number of iterations needed to converge than `Adam`.

When looking at the figure 1 one can see several interesting things. First one of them is the sudden drop of log-likelihood function for `IWLS` in `penguins` dataset. Another one is that `IWLS` for `ionosphere` and `covertype` had two sudden jumps in log-likelihood function. In the rest of the plots the function seems to be coming to the convergence in a soft, gentle way.

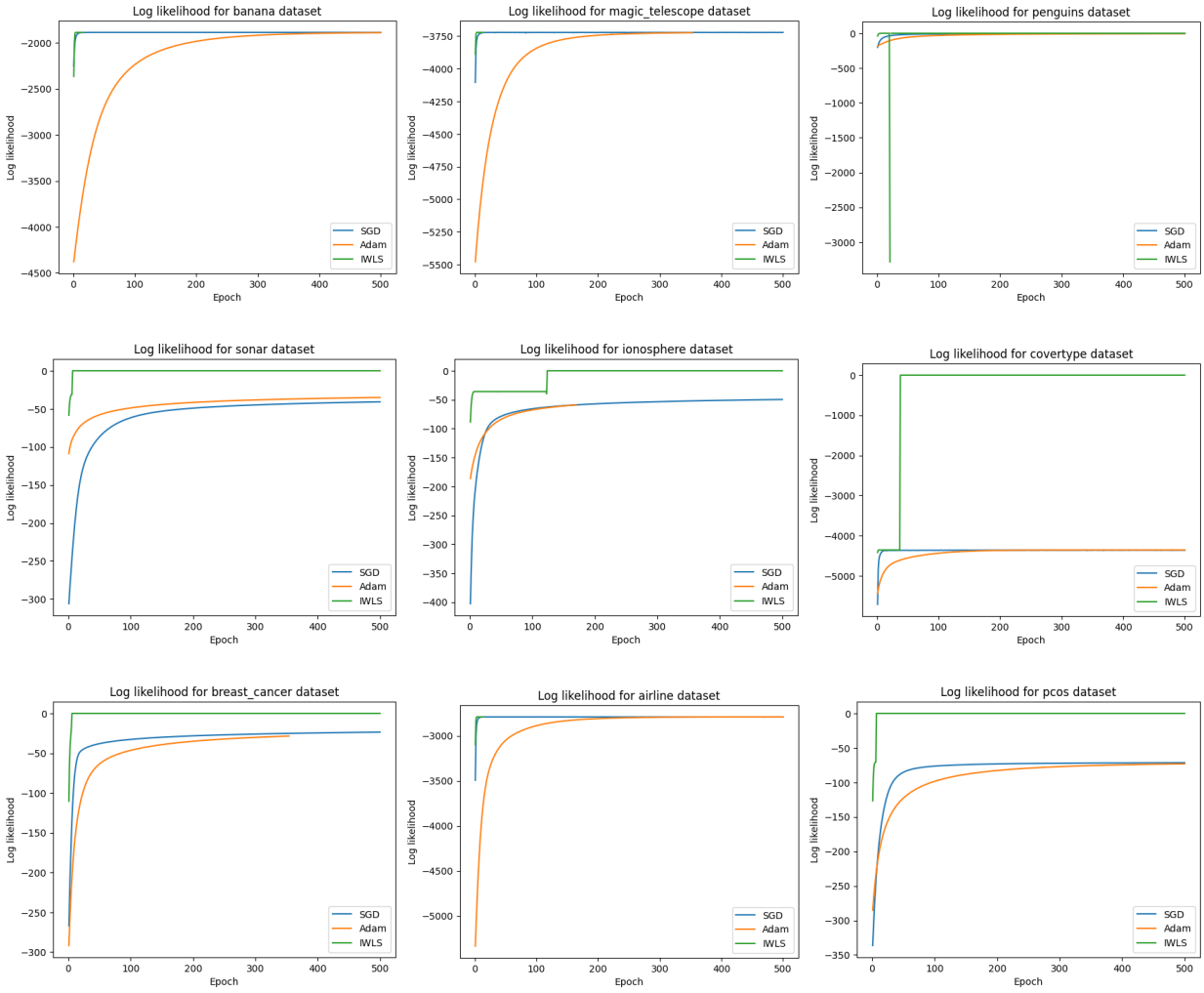


Figure 1: Log likelihood of algorithms per dataset

# 3 Comparison of classification performance

In order to inspect the performance of implemented optimizers, we compared it with 4 widely used methods: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), decision tree and random forest. All models were applied using existing implementations from scikit-learn with default parameters.

Experiments were conducted as follows: for each dataset-model combination, 10 train-test splits (test size = 0.2) were performed. Sample of size 1000 was drawn from each dataset. Balanced accuracy was calculated on test set and max number of iterations was set to 500. For implemented optimizers, both stopping rules were enabled, with parameters `n_iter_no_change` = 50 and `epsilon_change` =$10^{-5}$. Full results are shown in table 2 and on fig 2. We also inspected mean number of iterations before stopping in table 3. The variance of the results can be seen on the boxplots in figure 2.

As we can see, our implementations performed comparatively with widely used solutions. sometimes even outperforming them. IWLS and Adam achieved same mean scores of 0.83, with SGD achieving slightly lower 0.78. Best method in most cases turned out to be random forest (mean 0.88). Thanks to using boxplots in fig 2, we see that variation of results van vary greatly based on individual dataset and model. Also number of iterations needed to stop training was directly dependend on model as well as dataset.

Table 2: Mean balanced accuracy of examined methods on 9 inspected datasets

| | SGD | IWLS | Adam | LDA | QDA | RandomForest | DecisionTree | mean |
|---|---|---|---|---|---|---|---|---|
| banana | 0.77 | 0.85 | 0.86 | 0.86 | 0.91 | **0.94** | 0.88 | 0.87 |
| magic_telescope | 0.72 | 0.74 | 0.71 | 0.71 | 0.73 | **0.76** | 0.73 | 0.73 |
| penguins | 0.89 | 0.94 | 0.99 | **1.00** | 0.82 | **1.00** | **1.00** | 0.95 |
| sonar | 0.71 | 0.72 | 0.76 | 0.75 | 0.72 | **0.80** | 0.74 | 0.74 |
| ionosphere | 0.84 | 0.82 | 0.85 | 0.83 | 0.88 | **0.93** | 0.87 | 0.86 |
| breast_cancer | 0.93 | 0.95 | **0.96** | 0.95 | 0.96 | 0.94 | 0.93 | 0.95 |
| airline | 0.78 | 0.88 | 0.83 | 0.86 | 0.84 | **0.90** | 0.89 | 0.85 |
| covertype | 0.61 | 0.69 | 0.68 | 0.69 | 0.58 | **0.74** | 0.64 | 0.66 |
| pcos | 0.77 | 0.85 | 0.82 | **0.89** | 0.79 | 0.88 | 0.81 | 0.83 |
| mean | 0.78 | 0.83 | 0.83 | 0.84 | 0.80 | **0.88** | 0.83 | |

Table 3: Mean number of iterations

| | SGD | IWLS | Adam | mean |
|---|---|---|---|---|
| banana | 50.40 | 14.00 | 410.00 | 158.13 |
| magic_telescope | 113.10 | 14.00 | 500.00 | 209.03 |
| penguins | 56.80 | 51.80 | 330.50 | 146.37 |
| sonar | 132.70 | 51.70 | 460.00 | 214.80 |
| ionosphere | 243.10 | 55.00 | 500.00 | 266.03 |
| breast_cancer | 63.80 | 52.50 | 316.20 | 144.17 |
| airline | 57.40 | 14.00 | 239.00 | 103.47 |
| covertype | 64.00 | 54.00 | 323.90 | 147.30 |
| pcos | 51.10 | 30.40 | 174.70 | 85.40 |
| mean | 92.49 | 37.49 | 361.59 | |

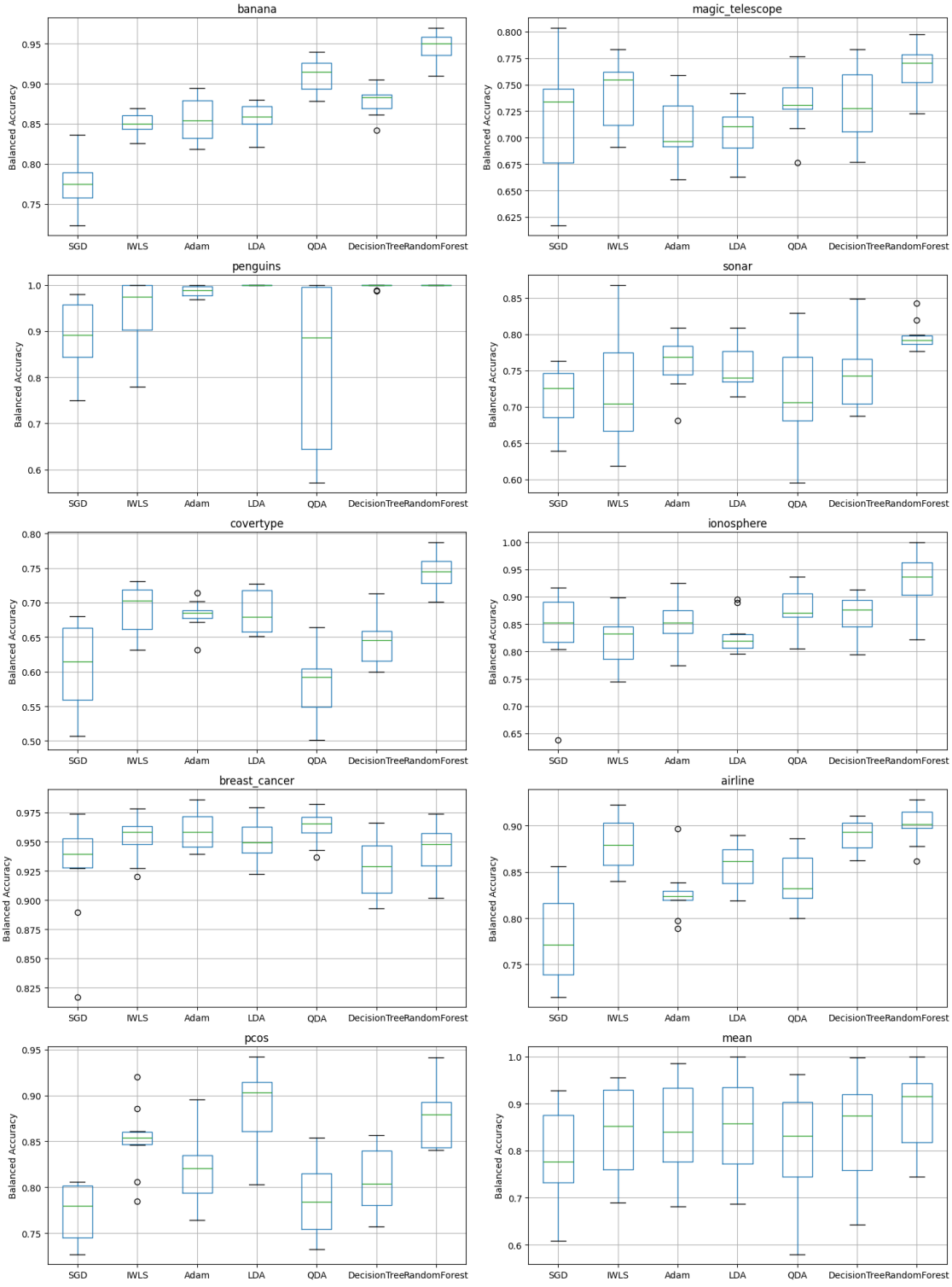


Figure 2: Boxplots of balanced accuracy achieved by examined methods on 9 inspected datasets and aggregated means

# 4 Comparison of classification performance of models with and without interactions

In the end, we inspected the impact of inclusion of interactions (defined as product of each pair of variables) on implemented optimizers' accuracy and number of iterations. We limited ourselves to 3 datasets with smaller number of variables. The experiments were conducted in the identical manner to chapter 3. The results are shown in tables 4 and 5.

The addition of interactions seems to have the biggest impact on SGD optimizer. It signficantly improves it accuracy, as well extends the learning process before reaching stop conditions, which seems intuitive. For the other optimizers the addition of interaction doesn't have big impact on length training. However, for IWLS, it makes the accuracy much lower. It seems to be the caused by low scored achieved on penguins dataset. It is rather particular dataset, as some models are able to achieve perfect accuracy on it (see 2 - on 3, we can see that IWLS with interactions achieved very varied results, with some as low as 0.2 - it indicates some problems with the model in this particular case.

## 4.1 Ending conclusions

During our work, we were able to draw some conclusions about constructing logistic regression optimizers. What stood out is the impact individual details can have on training process - for example, weight initialization and feature scaling were crucial to enable some models to train at all. Conducted experiments prove that our implemented models can perform as well as widely used solutions. However, their speed was considerably worse - but it is to be expected, as code optimalization was not the purpose of this task.

Table 4: Mean accuracy of models with and without interactions

| | SGD | SGD_INT | IWLS | IWLS_INT | Adam | Adam_INT | mean |
|---|---|---|---|---|---|---|---|
| banana | 0.77 | 0.90 | 0.85 | **0.93** | 0.86 | **0.93** | 0.87 |
| magic_telescope | 0.72 | 0.72 | 0.74 | **0.76** | 0.71 | 0.70 | 0.72 |
| penguins | 0.89 | 0.92 | 0.94 | 0.60 | **0.99** | **0.99** | 0.89 |
| mean | 0.79 | 0.85 | 0.85 | 0.76 | 0.85 | **0.88** | |

Table 5: Mean number of iterations of models with and without interactions

| | SGD | SGD_INT | IWLS | IWLS_INT | Adam | Adam_INT | mean |
|---|---|---|---|---|---|---|---|
| banana | 50.40 | 67.60 | 14.00 | 16.30 | 410.00 | 454.60 | 168.82 |
| magic_telescope | 113.10 | 87.10 | 14.00 | 18.90 | 500.00 | 500.00 | 205.52 |
| penguins | 56.80 | 158.40 | 51.80 | 50.40 | 330.50 | 282.50 | 155.07 |
| mean | 73.43 | 104.37 | 26.60 | 28.53 | 413.50 | 412.37 | |

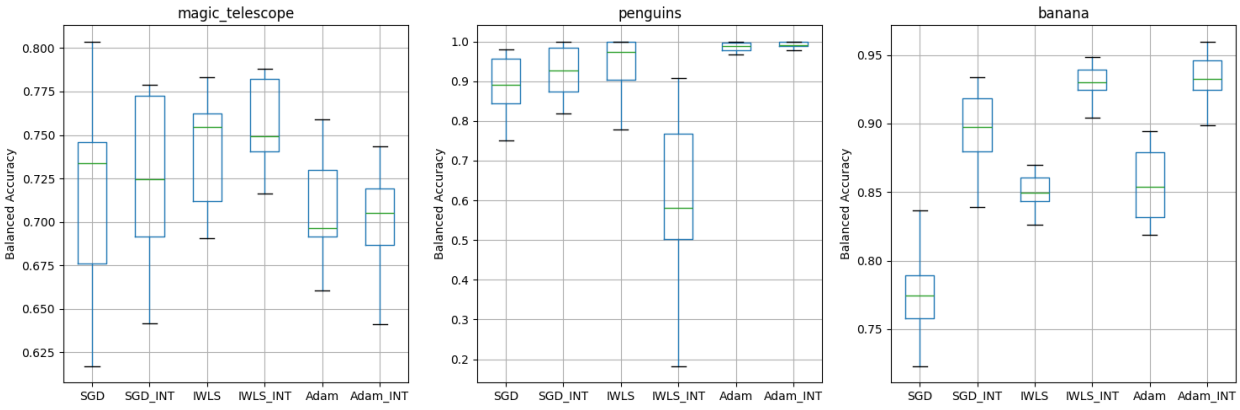


Figure 3: Boxplots of balanced accuracy achieved by implented methods with and without interactions on 3 small datasets