Logistic Regression

Jingxu (Jack) Hu
jingxu.hu@mail.mcgill.ca

Alex Goulet
alex.goulet@mail.mcgill.ca

Karanvir Sidhu
karanvir.sidhu@mail.mcgill.ca

Abstract

This paper assesses the performance of a Logistic Regression binary classifier on two distinct data sets. The two data sets separately consist of data related to financial insolvency and cancer research. The classifier was trained using Gradient Descent and final models were chosen using k-fold cross validation. The primary model parameters involved are learning rates, decay rates, feature orders, stopping criteria, data set preprocessing techniques, and feature selection. The results suggest that for both data sets, model accuracy tended to increase as the polynomial model orders were increased from 1 to 2 or 3 and then decrease afterwards. Additionally, the accuracy tended to increase when the magnitude of the stopping error was decreased, the learning rate was decreased, their decay rate was decreased, and when some of the features were selectively removed. Lastly, preprocessing the data from both sets using standardization and normalization yielded higher final accuracy on the test set across all models. The Hepatitis final model yielded an accuracy of 93.10% on the testing set using a 3^{rd} order model, while the Bankruptcy final model achieved an accuracy of 80.65% on the test set using a 2^{nd} order model with the removal of redundant features.

1 Introduction

1.1 Overview

In the realm of machine learning, there exits a vast number of methodologies for predicting data of various types. Often times, there are particular rules one must adhere to in order to apply a certain algorithm to a specific type of data from a certain field of research. For instance, when to use Support Vector Machines vs Bag of Quantized Words for an image detection/classification problem. It is common to analyze prediction/classification problems with various classifier models. The most appropriate model/classifier chosen would depend largely upon the nature of the data and whether or not the problem falls under regression or classification. Formally, regression in machine learning represents situations where models predict true output values given sample data[1], with the true outputs being either 0 or 1 in the case of binary classification. The purpose of this paper is to examine the performance a Logistic Regression binary classifier on two distinct data sets, namely Hepatitis and Bankruptcy, with continuous and discrete features. Furthermore, if need be, various means of improving the performance of the Logistic Regression classifier will be explored in order to maximize accuracy.

1.2 Logistic Regression

Logistic Regression is a classification algorithm derived from Linear Regression. In Logistic Regression, one attempts to directly model the decision boundary between two binary class labels by estimating $P(y_i = 1 \,|\, x_i)$ and therefore $P(y_i = 0 \,|\, x_i)$ through the use of the Sigmoid function σ [1]. The output of the σ function is interpreted as a probability ranging between 0 and 1 while its

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

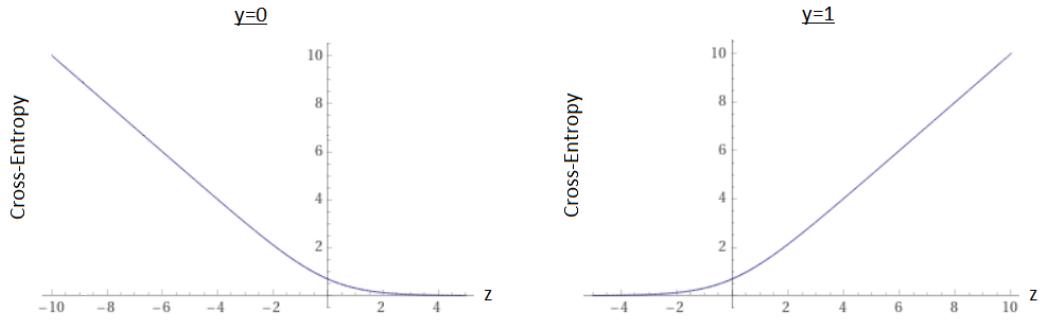

Figure 1: Cross Entropy Function

argument corresponds to the log-odds ratio of $P(y = 1 \mid x)$ over $P(y = 0 \mid x)$ and is modeled as a weighted linear combination of the input data feature values.

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

$$P(y_i = 1 \mid \boldsymbol{x_i}) = \sigma(\boldsymbol{w^T}\, \boldsymbol{x_i}) \tag{2}$$

$$P(y = 0 \mid \boldsymbol{x_i}) = 1 - \sigma(\boldsymbol{w^T}\, \boldsymbol{x_i}) \tag{3}$$

$$Cross - entropy = -\sum_{i=1}^{n} y_i \ln(\sigma(\boldsymbol{w^T}\, \boldsymbol{x_i}) + (1 - y_i) \ln(1 - \sigma(\boldsymbol{w^T}\, \boldsymbol{x_i}) \tag{4}$$

To train the binary classifier, Log-Likelihood of the logistic function can be maximized by minimizing the associated Cross-Entropy Loss on the training set through an optimization algorithm such as Gradient Descent, The weight vector is gradually optimized in such a way as to decrease the error of the binary classifier by choosing an appropriate search direction and learning rate. From Figure 1, the cross-entropy for a certain class label ($y = 1$ or $y = 0$) is a convex function and so any optimization algorithm should in theory converge to a global minimum in the absence of local minima. Furthermore, as the function being optimized is convex, more powerful optimization algorithms such as the Conjugate Gradient can be used. This is because the cross-entropy function behaves somewhat like a quadratic function.

Finally, during classification, the trained model directly estimates the class label probabilities on the test set and assigns the corresponding class label accordingly. The final reported test accuracy is then evaluated as a ratio of class labels that were predicted accurately over the total number of test samples.

1.3 Data Sets and Findings

In order to evaluate the performance of Logistical Regression, the algorithm was applied to two distinct high dimensional data sets. Throughout all the experiments, many elements affected the performance metrics of the classifier to different extents, such as data preprocessing techniques, model (feature) orders, learning rates, decay rates (rates at which the learning rates decrease with the number of iterations), stopping criteria, and feature selection. Firstly, data preprocessing allows for one to standardize across different features assuming their distribution is somewhat Gaussian and normalize features such that their range will all be between 0 and 1. After many simulations, results suggest that data preprocessing increases the final model accuracy on the test set. Secondly, a polynomial model of order 3 produced the highest model accuracy on the test set for the Hepatitis data set, while order 2 did the same for the Bankruptcy data set. Additionally, careful selection of the learning rates, decay rates, and stopping errors allowed for the final model accuracy on the test set to increase for both data sets. Lastly, removing certain features in the bankruptcy data set increased the final model accuracy on the test set, while keeping all features in the Hepatitis data set produced the highest final model accuracy on the test set.

2 Data Sets

2.1 Overview

The logistic regression algorithm is used to solve two binary classification problems, namely hepatitis and bankruptcy. The former aims to differentiate between patients with a terminal case of hepatitis and survivors, whereas the latter attempts to predict bankruptcy status based on a variety of economic attributes. The original hepatitis data set contains 142 samples of 19 features each, whereas the original bankruptcy data set contains 453 samples of 64 features each. Furthermore, in the hepatitis data set, the class label 1 is assigned to survivors and the class label 0 is assigned to terminal cases of hepatitis, with a proportion of 116 ones to 26 zeroes. In the bankruptcy data set, the class label 1 is assigned to bankruptcies and the class label 0 is assigned to non-bankruptcies, with a proportion of 203 ones to 250 zeroes.

2.2 Feature analysis

The hepatitis data set is composed of continuous features, such as "bilirubin", discrete features, such as age, and binary (discrete) features, such as sex. The bankruptcy data set, however, contains only continuous features. Moreover, in the hepatitis data set, among the continuous and discrete (non-binary) features, the distributions of the features are all somewhat Gaussian, with the exception of the feature corresponding to "protime". As for the bankruptcy data set, most of the features are somewhat Gaussian as well. However, some of them have such little variance that it can be difficult to observe that they follow a Gaussian distribution to some extent. Finally, new features were added to both data sets in the form of integer powers of the original features, thereby creating higher order models that can potentially provide more accurate predictions.

3 Results

This section is further divided into three subsections, namely Hyper-parameter Tuning, Model Selection and the removal of features. In this report, the number of iterations until convergence is reported to compare the convergence speed rather than the computation time. This is because the computation time for a machine is dependent on its memory as well as its processor, whereas the number of iterations provide a more general outlook on the convergence speed of the algorithms. It should also be noted that the implementation of the Logistical Regression used for the following results is fully vectorized to decrease computation time and that standardization was implemented for both data sets.

3.1 Hyper-parameter Tuning

This experiment was catered towards improving the convergence speed of the gradient descent using the learning rate, denoted by α. Since α corresponds to the step size taken in the direction of the gradient of cross-entropy function, it directly affects the convergence speed and therefore the accuracy of the gradient descent. The effect of the learning rate on the number of iterations can be clearly seen in Figure 2, in which the number of iterations increases very rapidly with the increase in learning rate for both data sets. Even when using a very small value for the learning rate, such as $\alpha = 0.001$, the number of iterations for the Bankruptcy and Hepatitis data sets were found to be 16828 and 17190, respectively. Then, the next step was to experiment with the time-decaying learning rate found in the equation below:

$$\alpha_k = \frac{\alpha_0}{\gamma\, k + 1} \tag{5}$$

where k is the iteration number, γ is the rate of decay, and α_0 is the initial learning rate. The hyper-parameters γ and α_0 were optimised by using (5) and using the fact that the cross-entropy function is convex (see Figure 1). Therefore, a large value for the initial learning rate and the decay was used, so that even if the optimization algorithm was oscillating due to the learning rate being too large, it would eventually converge due to the effect of the decay rate. The optimal values for initial learning rate and decay were found to be equal to 0.45 and 0.1, respectively.

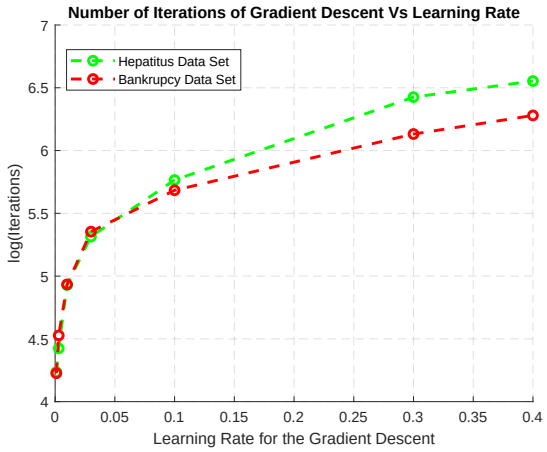

Figure 2: Log of number of iterations vs the Learning rate.

3.2 Model-Selection

In this section, the model selection procedure that is described was employed for selecting the best fitting model for the features provided in both data sets. As described in [2], the K-fold cross validation was used to select the model for both data sets. The K-fold cross validation was employed to both data sets for different model orders. The results obtained after the 10-fold cross validation for models of order 1, 2, 3 and 4 for both the Bankruptcy and Hepatitis data sets are shown in Table 3.2.

Table 1: Table depicting the average percent accuracies for the 10-fold cross validation

	Hepatitis Data	Bankruptcy Data
Model Order 1	73.33%	77.66%
Model Order 2	74.84%	78.88 %
Model Order 3	75.90%	71.94%
Model Order 4	73.18%	73.911%

Table 3.2 shows that for the Bankruptcy data set, a model of order 2 yields the maximum average 10-fold accuracy of 78.88%. In the case of the Hepatitis data set, a model of order 3 exhibits the highest average 10-fold accuracy. It should also be noted that the models of order 2 and 3 are unlikely to suffer from over-fitting as these models have relatively low variance. Furthermore, using the model of order 2 for the Bankruptcy data set resulted in a final accuracy of 79.57% when attempting to predict the class labels of the test set. As for the Hepatitis data set, a final accuracy of 93.103% was achieved on the test set. The final test accuracy results and number of iterations taken by Gradient Descent to converge during the final training process for both data sets are summarised in Table 2.

Table 2: Table summarising the final test results.

	Hepatitis Data	Bankruptcy Data
Model Order Chosen	3	2
Test Set Accuracy	93.103%	79.57%
Iterations	3685	4049

3.3 Feature-Selection

From the data visualizations graphs that can be found in Appendix A, features which do not appear to contribute to the classification were identified mainly based on the concept of Signal-to-Noise Ratio (SNR). For instance, in the case of the Bankruptcy data set, nine features which data contributing

to only one level (i.e. histogram bin) were found. The histograms for those features can be seen in Figure 8 of Appendix B. As shown in that figure, the features $20, 21, 24, 37, 43, 44, 47, 59$, and 69 have a very high number of feature values that correspond to only one class label. The number of minority samples are almost negligible compared to the majority samples. As such, those features were removed from the Bankruptcy data. However, with the removal of above-mentioned features, the the final accuracy on the test set increased from 79.57% to 80.64% and a faster convergence was observed as the number of features were reduced.

Similarly, for the Hepatitis data set, the concept of SNR was once again used and the feature named Protein was removed. While the final accuracy on the test set did decrease from 93.10% to 89.65%, a faster convergence speed was also observed. The results for the feature removal process are summarised in Table 3.

Table 3: Table depicting the accuracies and iterations after removing the features

	Hepatitis Data	Bankruptcy Data
Model Order used	3	2
Accuracy After Removing Features	89.65%	80.64%
Iterations took to Converge	2611	3956

4 Discussion and Conclusion

Firstly, increasing the learning rate directly increases the convergence speed, but at the risk of the Gradient Descent algorithm oscillating due to the learning rate being too large. Furthermore, since a faster convergence speed allows for a smaller stopping error and that a smaller stopping error directly leads to a higher accuracy, increasing the learning rate indirectly increases the accuracy of any model. Secondly, to be able to use one learning rate across multiple simulations, such as for each fold of the k-cross validation, a high learning rate coupled with a nonzero decay rate, such as 0.45 and 0.1 respectively, were found to be optimal. This is because the high learning rate ensures that all simulations will converge within a reasonable amount of time, while the decay rate ensures that the Gradient Descent algorithm eventually stops oscillating and starts converging in case the learning rate is initially too large. Finally, the final model accuracies on the test sets for the Bankruptcy and Hepatitis data sets were evaluated at 80.64% and 93.103%, respectively. It should be noted that the above-mentioned accuracies are obtained using the same learning rate.

4.1 Future Work

Several directions for further work based on the results of this project are described below:

- **Adaptive Learning rate**: The learning rate tuning method presented in this paper is based on heuristic methods. A more rigorous learning rate algorithm can be implemented as described in [3], where the learning rate can be computed using a second gradient descent algorithm. Moreover, other optimization methods can be used to compute the learning rate as a function of relative error.
- **Feature Models**: In this paper, model order ranging from from 1 to 4 were implemented. A variety of models common in logistic regression classification such as logarithmic models, hyperbolic models,etc. should implemented and tested using the existing code.
- **Feature Removal**: As can be seen in Section 3.2, the convergence speed was increased but at the cost of loss of final accuracy on the test set. More optimized, automated methods for removing features such as the LASSO method [5], methods based on correlation matrix [4], etc. should be implemented.
- **Conjugate Gradient**: Due to the cross-entropy function being convex and behaving somewhat like a quadratic, the Conjugate Gradient (CG) algorithm was implemented experimentally as the fit_CG function along with Gradient Descent fit function. However, due to the nature of this paper, it was not fully explored. Nonetheless, in the many tests that were done using the CG algorithm, results suggest that convergence speed was much higher and that the learning rates could be initialized at a higher value than with Gradient Descent without the algorithm oscillating.

5 Statement of Contributions

The project tasks were split equally amongst the three team members. Jack Hu was in charge of K-Fold Cross-Validation, data pipeline, overall software architecture, and performance analysis. Alex Goulet was in charge of data preprocessing and optimization. Karanvir Sidhu was in charge of simulations and model selection.

6 Appendix

6.1 Appendix A - Data Visualization

The below graphs represent the general form of feature distributions for both the Hepatitis and Bankruptcy data sets. The most common distribution found in both data sets is the Gaussian Distribution as seen in Figure 3 and 5. The other two graphs, Figure 4 and 6, represent random distributions found within the two data sets. Lastly, normalization was applied to both data sets in order to bring feature values between $0 - 1$

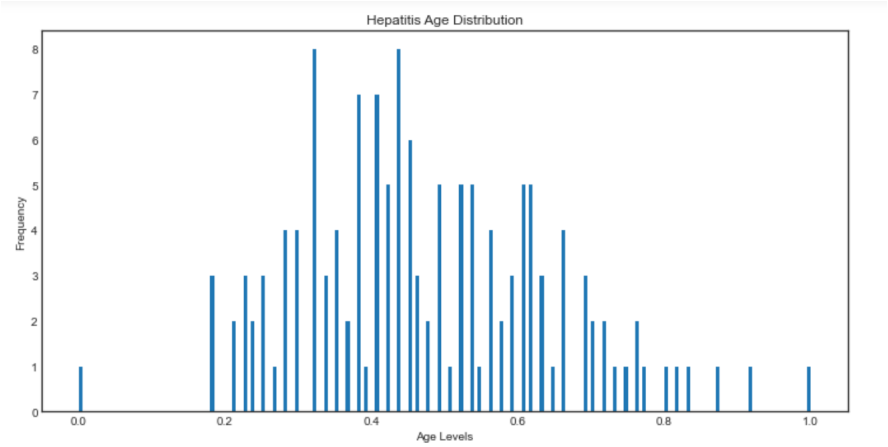

Figure 3: Hepatitis Gaussian Feature Visualization

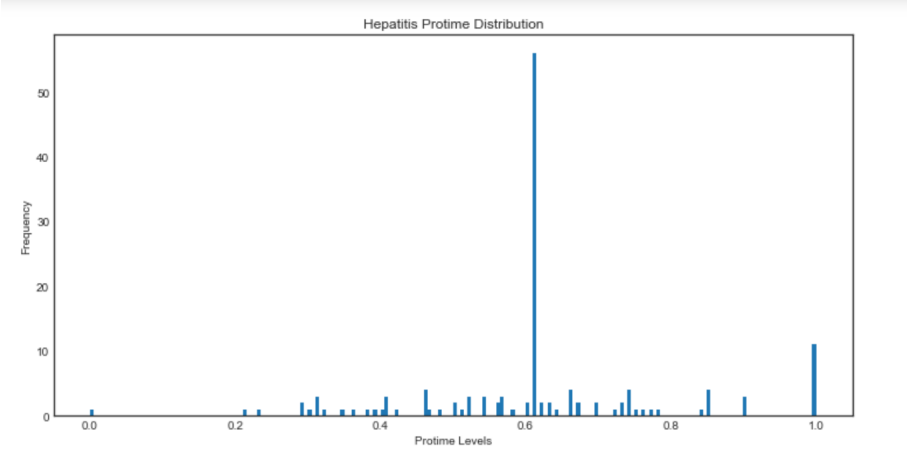

Figure 4: Hepatitis Random Feature Visualization

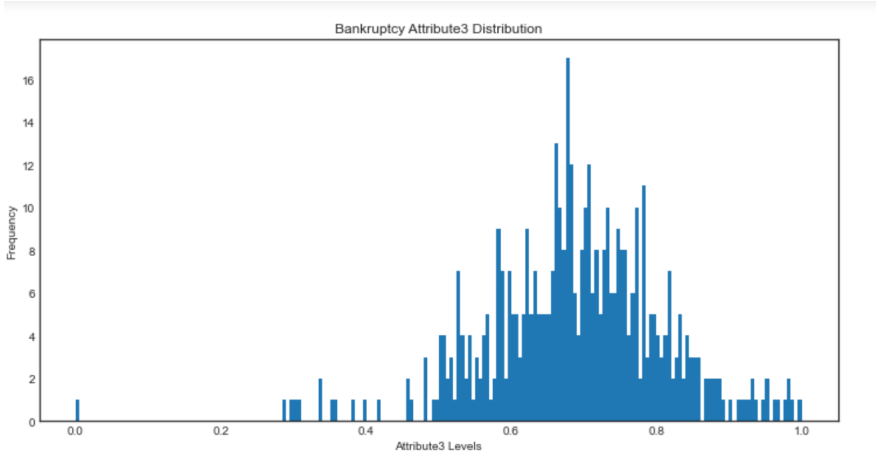

Figure 5: Bankruptcy Gaussian Feature Visualization

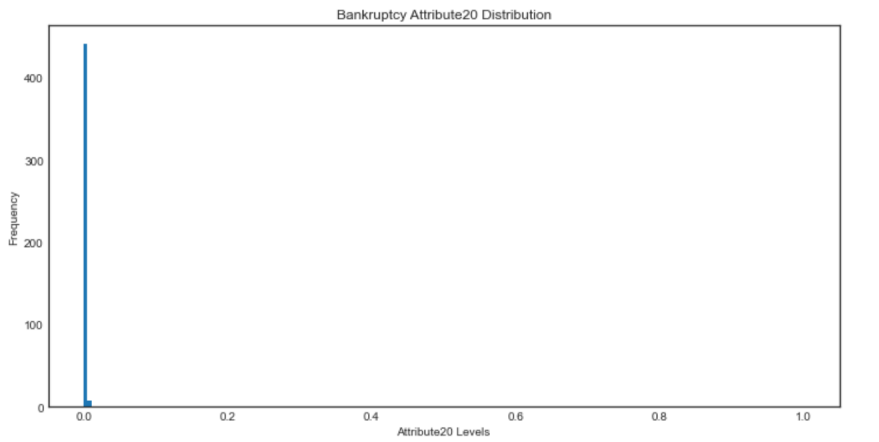

Figure 6: Bankruptcy Random Feature Visualization

6.2 Appendix B - Removed Features

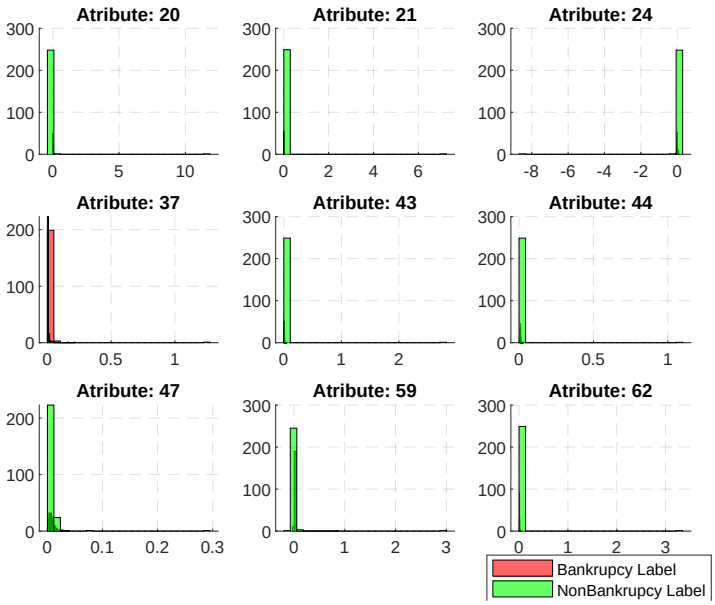

Figure 7: The histograms of the removed features for the Bankruptcy data set

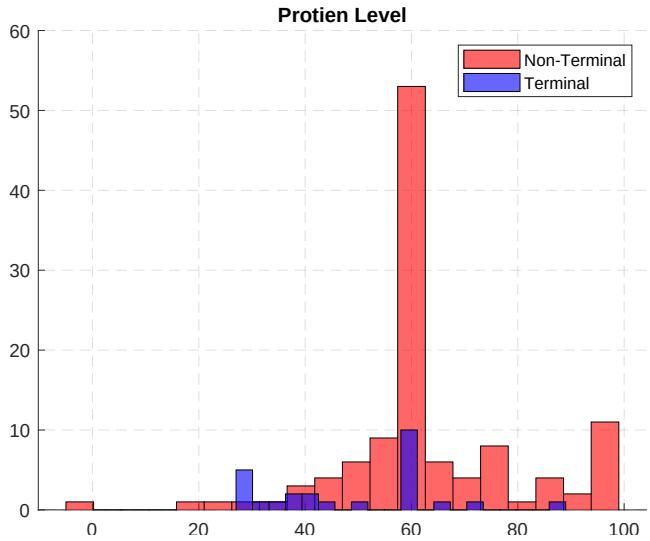

Figure 8: The histogram of the removed feature from the Hepatitis data set

References

1. X. Zou, Y. Hu, Z. Tian and K. Shen, "Logistic Regression Model Optimization and Case Analysis," 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 2019, pp. 135-139, doi: 10.1109/ICCSNT47585.2019.8962457.
2. N. Armanfard, “ECSE 551 - Machine Learning for Engineers Lecture 4 and 5 - Linear Regression,” in Machine Learning Lecture Slides.
3. Ravaut M, Gorti S. Gradient descent revisited via an adaptive online learning rate. arXiv preprint arXiv:1801.09136. 2018 Jan 27.
4.Toloşi L, Lengauer T. Classification with correlated features: unreliability of feature ranking and solutions. Bioinformatics. 2011 Jul 15;27(14):1986-94.
5.Meier L, Van De Geer S, Bühlmann P. The group lasso for logistic regression. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2008 Feb;70(1):53-71.